Министерство науки и высшего образования Российской Федерации
Томский государственный университет
систем управления и радиоэлектроники (ТУСУР)

XVIII Международная научно-практическая конференция

ЭЛЕКТРОННЫЕ СРЕДСТВА И СИСТЕМЫ УПРАВЛЕНИЯ

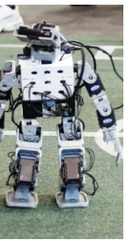
Материалы докладов

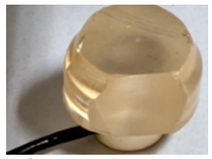
г. Томск

16–18 ноября 2022 г.

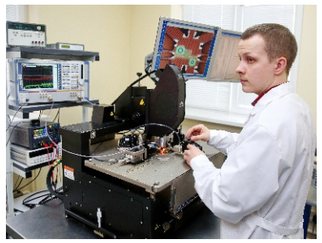
В двух частях

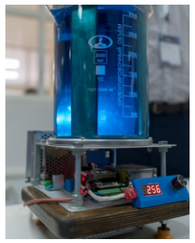
ЧАСТЬ 2

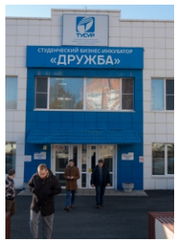




Выращенный кристалл ниобата лития со срезами по осям симметрии (пьезоэлектрический преобразователь установлен снизу)







Министерство науки и высшего образования Российской Федерации
Томский государственный университет систем управления и радиоэлектроники (ТУСУР)

ЭЛЕКТРОННЫЕ СРЕДСТВА И СИСТЕМЫ УПРАВЛЕНИЯ

XVIII Международная научно-практическая конференция

16–18 ноября 2022 г.

Материалы докладов

В двух частях

Часть 2

В-Спектр
Томск – 2022

УДК 621.37/39 + 681.3
ББК (Ж/О) 32.84.85.965
Э 45

Э 45 **Электронные средства и системы управления**: материалы докладов XVIII Международной научно-практической конференции (16–18 ноября 2022 г.): в 2 ч. – Ч. 2. – Томск: В-Спектр, 2022. – 266 с.

Electronic Devices and Control Systems: XVIII International Scientific-practical Conference (November 16–18, 2022). – Part 2. – Tomsk: V-Spektr, 2022. – 266 p.

ISBN 978-5-91191-479-0
ISBN 978-5-91191-480-6 (Ч. 1)
ISBN 978-5-91191-481-3 (Ч. 2)

Часть 1 – 2–13 секции.
Часть 2 – 14–22 секции.

Сборник содержит материалы докладов, представленных на XVIII Международной научно-практической конференции «Электронные средства и системы управления» (Томск, 16–18 ноября 2022 г.), по следующим направлениям: радиотехнические и телекоммуникационные системы; наноэлектроника СВЧ, интеллектуальные системы проектирования, автоматизация проектирования электронных устройств и систем; космические радиоэлектронные устройства; нанотехнологии в электронике; антенны и микроволновые устройства; приборы и методы контроля; нелинейная оптика; интеллектуальная силовая электроника и преобразовательная техника; плазменная электроника; органическая и неорганическая полупроводниковая светотехника; электромагнитная совместимость; автоматизация и оптимизация систем управления и обработки информации; информационная безопасность; информационные технологии в управлении и принятии решений; информационные технологии в обучении; видеоинформационные технологии и цифровое телевидение; цифровая связь и вейвлет-фрактальная обработка сигналов и изображений; вычислительный интеллект и машинное обучение; робототехника и мехатроника; проектирование и разработка веб-систем и мобильных приложений.

Для студентов, преподавателей и специалистов, интересующихся проблемами систем управления.

УДК 621.37/39 + 681.3
ББК (Ж/О) 32.84.85.965

Спонсор конференции:

ООО «50ОМ ТЕХ.»

ISBN 978-5-91191-479-0
ISBN 978-5-91191-481-3 (Ч. 2)

© ТУСУР, 2022

СПОНСОР КОНФЕРЕНЦИИ – ООО «50ОМ ТЕХНОЛОДЖИЗ»

50ohm Technologies

ООО «50ом Технолоджиз»
info@50ohm.tech
https://50ohm.tech/ru
634045, г. Томск

Тел.: +7-923-408-0408
Факс: 50ohmTechRus
company/50ohm-technologies-llc

Компания 50ohm Technologies разрабатывает программное обеспечение для автоматизации измерений, построения моделей компонентов и проектирования ВЧ- и СВЧ-радиоэлектронных устройств.

50ohm Technologies предлагает решения задач автоматизации рабочих процессов с учётом индивидуальных особенностей предприятия. Миссия компании – разрабатывать удобные, умные, интеллектуальные инструменты, которые помогают инженерам в области СВЧ-электроники быстро решать возникающие задачи.

Компания разрабатывает программные решения по направлениям:

- автоматизация измерений устройств электроники и радиоэлектроники;
- базы данных результатов измерений и их автоматическая обработка;
- автоматизация проектирования СВЧ-устройств;
- построение моделей электронных компонентов;
- подготовка научно-технической документации.

Компания обладает компетенциями в использовании методов искусственного интеллекта и экспертных систем. Внедрение данных технологий на предприятие позволяет перейти на качественно новый уровень и автоматизировать наиболее рутинные этапы бизнес-процессов.

50ohm Technologies предлагает услуги по разработке систем автоматизации измерений, реализуемых на основе оборудования заказчика. Использование готовых сценариев измерений конкретных компонентов и устройств в значительной степени сократит время тестирования и повысит эффективность измерений. 50ohm Technologies производит разработку решений автоматизированной генерации технической документации по типовым шаблонам – от оформления графиков до формирования готовых документов.

Компания обладает значительным опытом построения моделей пассивных и активных СВЧ-компонентов. Создание программного обеспечения на основе общепринятых и авторских методик в значительной степени упрощает и автоматизирует процесс построения моделей, уменьшая временные и финансовые затраты предприятия на данном этапе.

Наиболее сложным этапом в процессе проектирования СВЧ-устройства является получение схемотехнического и топологического решений. За годы научной работы коллектив получил успешный опыт разработки и использования программных модулей САПР, основанных на методах искусственного интеллекта. Такие программы позволяют получить целый набор решений, из которых разработчик может выбрать наиболее подходящее для дальнейшей реализации. Также компания занимается автоматизацией проектных операций в популярных коммерческих САПР СВЧ-устройств и интеграцией между ними.

Директор ООО «50ом Технолоджиз» – ***Калентьев Алексей Анатольевич***

Секция 14

АВТОМАТИЗАЦИЯ И ОПТИМИЗАЦИЯ СИСТЕМ УПРАВЛЕНИЯ И ОБРАБОТКИ ИНФОРМАЦИИ

Председатели секции – ***Кориков Анатолий Михайлович****, профессор каф. АСУ, д.т.н.;*
Катаев Михаил Юрьевич*, профессор каф. АСУ, д.т.н.*

УДК 621

А.Б. Кураков, А.Б. Гомбоин

Нейросетевой подход в задаче определения центра тяжести человека

Нейронные сети с каждым годом становятся все популярнее и популярнее. Многие задачи компьютерного зрения уже сегодня используют нейронные сети. Старые алгоритмы хоть и надежны, но требуют большего времени на обработку и не всегда дают точные результаты. Рассматривается применение двух нейронных сетей в задаче определения центра тяжести человека. С помощью полученных данных центра тяжести можно получить множество показателей походки человека, что позволяет определять возможные болезни человека. Такой подход будет интегрирован в общую информационную систему обработки изображений с целью выделения паттернов движения человека.

Ключевые слова: изображение, методы обработки, центр тяжести, походка человека, нейронные сети.

Образ жизни людей кардинально изменился за последние десятилетия, увеличились и продолжительность жизни, и количество времени, проводимое без активного участия опорно-двигательного аппарата. Все это непосредственно влияет на наш организм, количество людей, страдающим от болезней опорно-двигательного аппарата, с каждым годом растет.

Уже сейчас Всемирная организация здравоохранения бьет тревогу, что миру не хватает современных подходов в вопросах лечения и реабилитации пациентов с болезнями опорно-двигательного аппарата.

Ходьба является одним из процессов деятельности опорно-двигательного аппарата. В среднем по миру человек проходит 2700–5000 шагов в день, если же человек страдает плоскостопием, то он ежедневно разрушает свой же организм, и вовремя подобранное лечение позволит в разы продлить корректную работу опорно-двигательного аппарата.

Исследование ходьбы человека неразрывно связано с фиксацией характеристик тела во время передвижения. Анализ полученных данных позволяет определять различные виды и характеристики ходьбы, влияние движения различных мышц и суставов друг на друга.

Одной из важнейших характеристик походки человека является мера положения и перемещения человека в пространстве и времени. Таким образом, можно говорить, что, зная изменения центра тяжести человека, можно определять множество параметров походки, таких как симметрия шага, длина шага, скорость одного шага и т.д.

Для получения необходимых параметров походки человека необходимо знать центр тяжести человека, поэтому необходимо найти человека на изображении. Для этого строится специальная сцена с простым статичным фоном. Существуют различные подходы для нахождения человека на изображении, основанные на разнице изображений, применении алгоритмов вычитания фона и фонового моделирования гауссовской смеси, применении нейронных сетей [1].

Далее рассмотрим именно применение нейронных сетей в задаче определения центра тяжести человека.

Нахождение человека на изображении

Для начала требуется внести определение нейронной сети. Нейронная сеть (или искусственная нейронная сеть) представляет собой математическую модель, а также её программное или аппаратное воплощение, построенное по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма [2].

YOLO (you only look once) – архитектура нейронных сетей, предназначенная для обнаружения объектов на изображении. Главной особенностью YOLO является их уникальный подход к решению задачи обнаружения объектов на изображении.

Один из способов решения задачи детекции заключается в разбиении изображения на квадратные области, затем классификация этих областей на наличие объекта и классификация самого объекта. Таким образом, изображение просматривается дважды (один раз для определения областей, где есть объект, второй – для классификации этого объекта). Этот способ работает долго и требует больших затрат вычислительных мощностей.

YOLO же использует другой принцип. Исходное изображение сжимается таким образом, чтобы получить квадратную матрицу размером 13×13, в каждой клетке которой записана информация о наличии объекта и классе этого объекта на соответствующей части картинки. Таким образом, YOLO просматривает картинку один раз, что существенно увеличивает скорость обработки.

YOLOv5 – обновленная пятая версия YOLO, созданная с использованием фреймворка PyTorch. YOLOv5 интегрирована в одноименный модуль для Python3, который можно установить с pypi. Данный модуль предоставляет отличную структуру как для обучения модели, так и для тестирования полученной модели с возможностью построения графиков различных показателей [3].

Для тестирования YOLO были использованы 10 заранее заготовленных видеозаписей, удовлетворяющих следующим условиям: статический фон, человек проходит из одного края в другой 1–3 раза, длительность прохода в одну сторону 5–7 с, т.е. 5–8 шагов. Результат тестирования приведен на рис. 1.

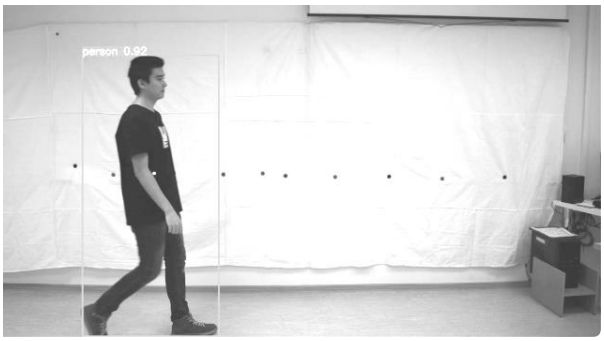


Рис. 1. Применение YOLOv5 на видеозаписи идущего человека

Нахождение центра тяжести человека

Существует множество различных способов определения центра тяжести человека. В данной работе будет рассматриваться способ определения ключевых точек человека.

После того как с помощью YOLOv5 был обнаружен человек, необходимо вырезать получившийся прямоугольник с человеком, для того чтобы определять ключевые точки более точно и быстрее.

Всего используется 17 ключевых точек:

- нос;
- левый глаз;
- правый глаз;
- левое ухо;
- правое ухо;
- левое плечо;
- правое плечо;
- левый локоть;
- правый локоть;
- левое запястье;
- правое запястье;
- левое бедро;
- правое бедро;
- левое колено;
- правое колено;
- левая лодыжка;
- правая лодыжка.

Для определения описанных выше ключевых точек используется отдельная нейронная сеть. Название данной нейросети MoveNet. Она использует тепловые карты для точной локализации ключевых точек человека. Это восходящая модель оценки, которая означает, что она сначала обнаруживает суставы человека. Подход, который используется, позволяет невероятно быстро определять ключевые точки любого человека [4].

Исходя из описанного выше, центра тяжести можно найти по следующим формулам:

$$x_c = \frac{\sum_{i=0}^{17} x_i}{17}, \quad (1)$$

$$y_c = \frac{\sum_{i=0}^{17} y_i}{17}. \quad (2)$$

Результат применения данного алгоритма приведен на рис. 2.

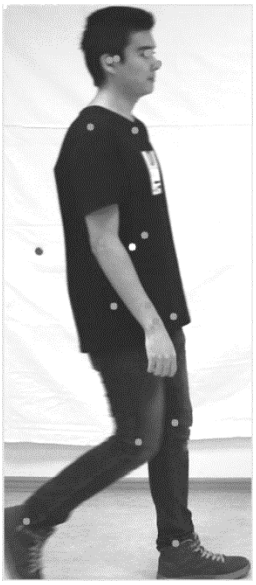


Рис. 2. Найденные 17 ключевых точек и центр тяжести

Обработка результатов

Так как обрабатывается не одно изображение, а целая видеозапись, это позволяет получить полную картину изменений центра тяжести человека. Однако не стоит забывать, что хоть описанный выше алгоритм нахождения центра тяжести является быстрым и достаточно точным, все равно даже в нем есть место погрешностям и неточностям, таким образом, требуется проводить дополнительное сглаживание полученной кривой. Для сглаживания применяется фильтр Свацки–Голея [5]. На рис. 3 приведен график без сглаживания, на рис. 4 – сглаженный график.

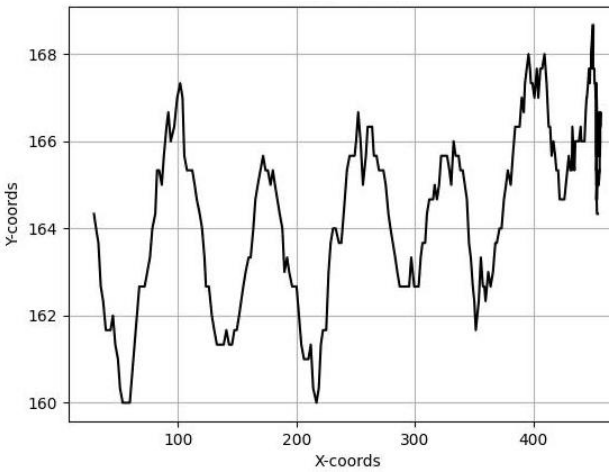


Рис. 3. Необработанные данные центра тяжести

Если сопоставить локальные минимумы и максимумы, можно заметить, что максимумы соответствуют, когда у человека две ноги сведены вместе. Локальные минимумы равны ситуации, когда ноги у человека максимально разведены.

Заключение

В данной работе было рассмотрено применение нейронных сетей в задаче определения центра тяже-

сти человека. Рассмотренный подход с использование двух нейронных сетей показал свою быстроту и эффективность, что позволяет применять данный подход в дальнейшей разработке информационной системы обработки изображений с целью выделения паттернов движения человека.

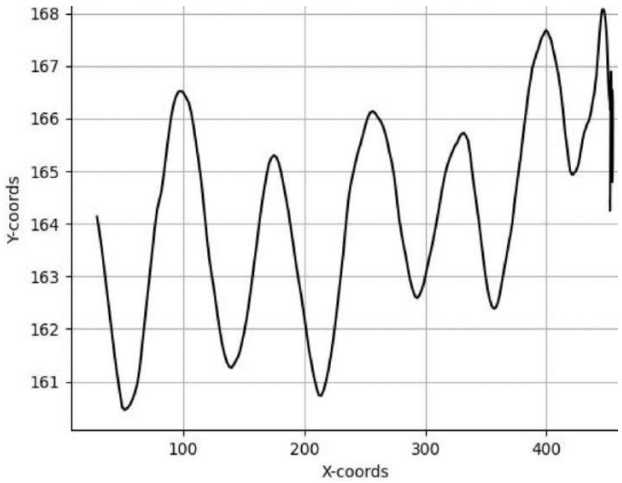


Рис. 4. Обработанные данные центра тяжести

Приведенный подход позволяет определять центр тяжести не только в ситуации, когда используется исключительно статический фон, но и в ситуации, когда фон может динамически изменяться, что делает данный подход в разы практичнее.

Литература

1. Катаев М.Ю. Выделение паттернов движения на основе центра масс фигуры человека с использованием видеоизображений / М.Ю. Катаев и др. // Доклады ТУСУР. – 2019. – Т. 22, № 4. – С. 56–61.

2. Использование YOLOv5 для задачи детекции [Электронный ресурс]. – Режим доступа: https://vc.ru/new-techaudit/326571-ispolzovanie-yolov5-dlya-zadachi-detekcii (дата обращения: 10.09.2022).

3. YOLOv5 Documentation [Электронный ресурс]. – Режим доступа: https://docs.ultralytics.com (дата обращения: 11.09.2022).

4. How to Use Background Subtraction Methods [Электронный ресурс]. – Режим доступа: https://docs.opencv.org/3.4/d1/dc5/tutorial_background_subtraction.html (дата обращения: 10.09.2020).

5. Human pose estimation with MoveNet [Электронный ресурс]. – Режим доступа: https://www.kaggle.com/code/ibrahimserouis99/human-pose-estimation-with-movenet (дата обращения: 11.09.2020).

6. Шапиро Л., Компьютерное зрение / Л. Шапиро, Дж. Стокман; пер. с англ. – 2-е изд. – М.: БИНОМ, 2013. – 752 с.

Кураков Андрей Борисович
Ассистент каф. автоматизированных систем управления (АСУ) ТУСУРа
Эл. почта: sniperaft@bk.ru

Гомбоин Александр Булатович
Студент каф. АСУ ТУСУРа
Эл. почта: a.gomboin@mail.ru

УДК 004.932

Р.К. Карпов, М.Ю. Катаев

Методика оценивания цветового качества кирпичной продукции

Рассмотрены методы анализа кирпичной продукции, представлен один из вариантов использования методов компьютерного зрения в задаче оценки качества производственной продукции – керамических кирпичей. Приведена методика цветового отличия для этой оценки. За основу использовались реальные изображения производственного процесса. Представленная методика для решения задачи оценки качества кирпичной продукции позволяет ускорить и автоматизировать процесс.

Ключевые слова: кирпич, строительное производство, оценка, цвет, брак, техническое зрение.

Кирпич – это искусственный камень правильной формы, используемый в качестве строительного материала [1]. Правильная форма, или форма без дефектов, – залог успешного строительства стен и других конструкций. Цвет кирпича является основным параметром, определяющим качество произведенной продукции, так как говорит о наличии или отсутствии различных дефектов, например, наличие явно выраженных цветовых отличий, как белые полосы на красном кирпиче. Также точное восприятие цвета является важным фактором при обработке и анализе для выявления остальных дефектов (трещины, сколы).

На предприятиях и заводах по производству строительной продукции процесс определения брака до сих пор является наименее автоматизированным. Несмотря на то, что современные подходы автоматизации процессов с помощью компьютерного зрения достигли значительных результатов, на предприятиях используется ручной физический труд. Это напрямую влияет на качество выполняемой работы, т.к. зависит от степени квалификации инженеров-специалистов. Также при ручном определении брака для анализа используется лишь небольшая часть от всей произведенной партии продукции, из-за чего брак может быть пропущен. Повышение качества процесса определения брака методами компьютерного зрения является важной и актуальной задачей производства.

Эффективность анализа качества кирпичной продукции заключается в используемых для этого методах. Различают разрушающие (основаны на прямом воздействии с изделием) и неразрушающие (позволяют проводить испытания над физическими явлениями, положенными в основу объектов анализа) методы анализа качества поверхностей. На производственном предприятии [2] для анализа кирпичной продукции инженеры-специалисты используют несколько кирпичей из партии для выявления средней оценки, что значительно снижает точность анализа всей партии в целом.

При выполнении контроля качества поверхности различного типа на предприятиях руководствуются нормативными документами (ГОСТ, СНиП, СП, РД) [3], где описаны разнообразные методики контроля, однако в существенно меньшей степени методики, основанные на методах компьютерного зрения. В этих документах определены нормы дефектности, которые связаны с конкретной технологией измерений. Поэтому оценка цвета кирпичной продукции методами компьютерного зрения позволит производить точный контроль качества произведенной продукции, что поможет при выявлении и решении как производственных проблем, так и экономических.

Автоматизированная система компьютерного зрения может решать задачи по оценке и анализу качества произведенной продукции. Основные показатели, которыми руководствуются инженеры при анализе, указаны в соответствующих нормативных документах ГОСТ. Для автоматизированной системы предлагается такое решение, при котором над транспортной лентой с блоком кирпичей будут находиться цифровые камеры [4], которые фиксируют и передают изображения в систему для обработки и анализа (рис. 1).

Рис. 1. Схема системы над транспортной лентой

Результаты работы

Методы обнаружения и анализа изображений выделяют свойства кирпича на изображении, по которым можно определять наличие дефектов в производстве. После обработки изображений собираются все полученные данные, выполняется анализ с помощью различных методов, например, на основе сравнения полученных характеристик продукции с эталонными характеристиками, и формируется протокол.

Изображение – это представление восприятия в цифровом виде. Лучи света, состоящие из цветового спектра, отражаясь от объекта, несут его изображение (тот цвет, который мы видим, – это и есть отраженный цвет), которое попадает в камеру (объектив) и фокусируется на матрице. Матрица (или аппарат-

ная составляющая камеры) преобразовывает поток света в поток электронов (электрический ток), который впоследствии проходит некоторые операции преобразования и превращается в битовую последовательность, которая уже и преобразовывается в изображение.

Изображение, полученное с помощью цифровых камер, подвержено внешним факторам, из-за чего цвет, который мы видим на изображении, может существенно отличаться от настоящего [5]. Поэтому для анализа является важной задача по восстановлению модели цвета кирпичной продукции.

Основными влияющими на цвет факторами являются матрица камеры (ее аппаратная функция), сам объект изображения (спектр отражения) и свет (излучение). Исходя из этого, значение пикселя в соответствующих координатах можно узнать по формуле (1):

$$DN(x, y) = \alpha + \beta \times IN(x, y), \quad (1)$$

где IN – цвет пикселя в соответствующих координатах, который находится по формуле (2):

$$DN(x, y) = \int (A(\lambda) \times I(\lambda) \times \cos\varphi \times \rho(\lambda, x, y)) d\lambda, \quad (2)$$

где $A(\lambda)$ – аппаратная функция цифровой камеры в RGB-каналах; $I(\lambda)$ – освещение (свет); $\cos\varphi$ – угол падения света; $\rho(\lambda, x, y)$ – отражающая способность объекта, которая и определяет цвет.

Аппаратная функция камеры представляется в виде спектра излучения трех RGB-каналов, а свет – в виде спектра электромагнитного излучения. Во время производства имеется несколько источников света – естественный (солнце) и искусственный (лампочка над транспортной лентой). На рис. 2 представлены спектры излучения для разных ламп.

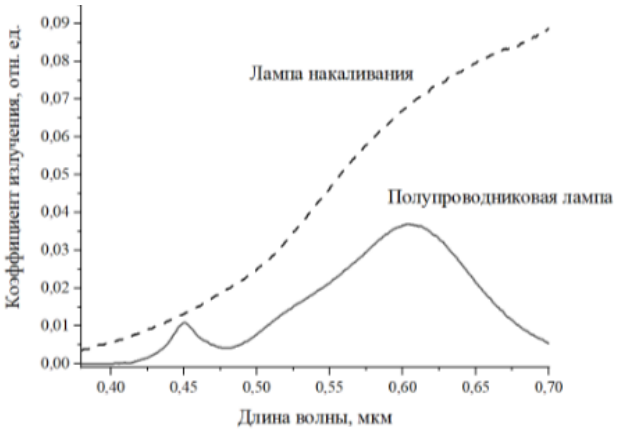


Рис. 2. Спектры излучения разных ламп (лампы накаливания и полупроводниковой лампы)

На практике во время производства спектр естественного источника света измерить невозможно из-за его динамичного состояния и отсутствия необходимых для этого приборов, поэтому мы можем использовать только спектр излучения искусственного источника света.

Как было сказано ранее, цифровая камера, как и мы, воспроизводит объект на изображении за счет того, что отраженные лучи от объекта, несущие цвет, попадают на сам объектив и формируют само изображение (которое мы и видим). Соответственно, если объекты поглощают и отражают часть спектра излучения, то они имеют свои отражающие функции, которые могут быть представлены в виде спектров излучения. На рис. 3 и 4 представлены спектры красного и коричневого кирпича.

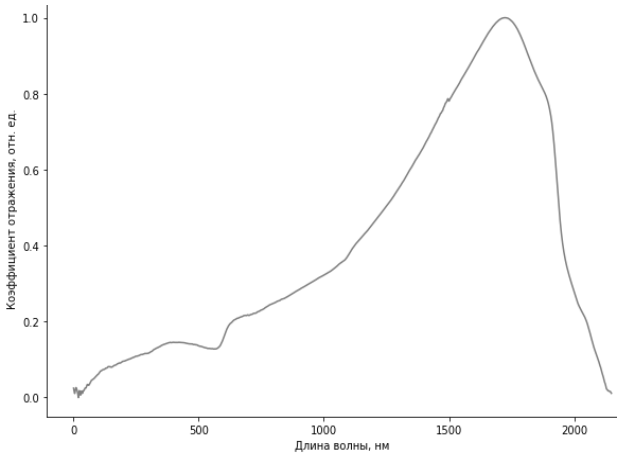


Рис. 3. Спектральная функция красного кирпича

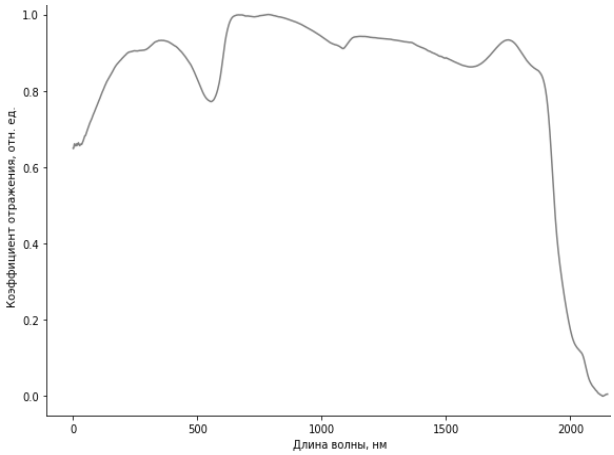


Рис. 4. Спектральная функция коричневого кирпича

Исходя из вышесказанного, чтобы составить модель цвета кирпича для анализа, необходимо исключить все возможные внешние факторы, влияющие на цвет, а именно воздействие освещения и аппаратную составляющую цифровой камеры. Для этого необходимо решить уравнение (3), при котором из изображения с объектом исключается изображение со светом:

$$P = \frac{\int (A(\lambda) \times I(\lambda) \times \cos\varphi \times \rho(\lambda, x, y)) d\lambda}{\int (A(\lambda) \times I(\lambda) \times \cos\varphi) d\lambda}. \quad (3)$$

Для реализации данного метода и составления модели цвета кирпича необходимо иметь:

1. Изображение с объектом (кирпичом) – это изображение, на котором четко виден объект, т.е. кирпич. Само изображение должно пройти первичную предобработку для исключения фона с изображения по маске, т.е. на изображении должен быть только кирпич и одноцветный фон (рис. 5).

2. Изображение со светом – это изображение, сделанное в процессе фотографирования источника света, т.е. излучения, которое направлено в объектив цифровой камеры. Необходимо, чтобы свет заполнял всё изображение. Цветовой тон такого изображения будет приближен к белому (рис. 6).

Рис. 5. Изображение с объектом без фона (красный кирпич)

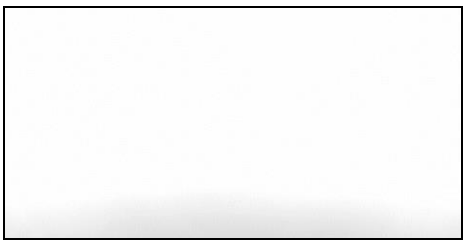


Рис. 6. Изображение со светом

Важно, чтобы аппаратные настройки цифровой камеры для обоих изображений были неизменными. В зависимости от аппаратных составляющих цифровой камеры, расстояния съемки и типа освещения (ламп) изображение со светом может отличаться. Так, например, изображение может быть более серым или приближенным к желтым или коричневым оттенкам.

Имея соответствующие изображения, необходимо получить значения для анализа и построения модели, для этого нужно построить их гистограммы. Из гистограммы кирпича для каждого канала цвета (синий, зеленый, красный) требуется определить значение пикселя с максимальным значением (высота функции). На рис. 7 представлена гистограмма кирпича, значения которого:

– синий: значение пикселя – 81, число пикселей – 84580;

– зеленый: значение пикселя – 144, число пикселей – 69874;

– красный: значение пикселя – 253, число пикселей – 294084.

Также на рис. 8 представлена гистограмма изображения со светом, который использовался для текущего объекта.

Конечным этапом моделирования является вычисление коэффициентов для каждого RGB-канала по описанной ранее формуле (3) и полученным значениям с соответствующих гистограмм.

Полученные коэффициенты, точки можно использовать для дальнейшего анализа. Совместив полученные коэффициенты модели кирпича и соответствующего спектра, можно найти отклонение по спектральной функции кирпича (рис. 9):

– отклонение по синему каналу цвета: 2,85%;

– отклонение по зеленому каналу цвета: 17,24%;

– отклонение по красному каналу цвета: 0,13%.

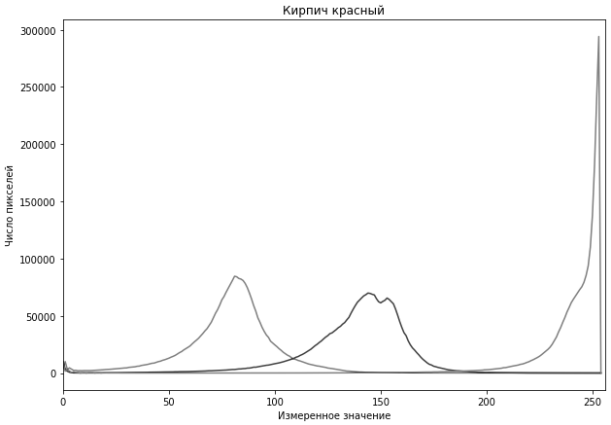


Рис. 7. Гистограмма изображения с кирпичом

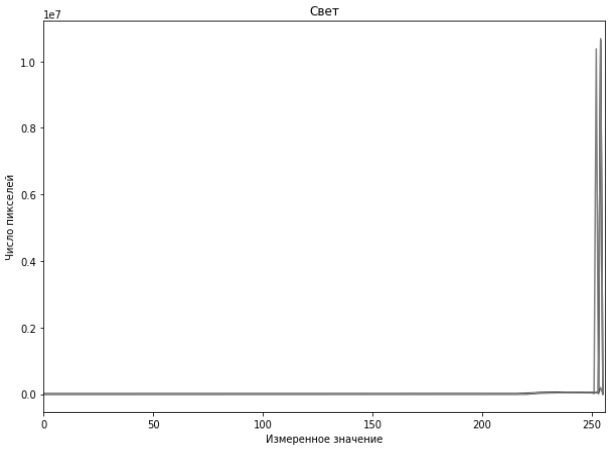


Рис. 8. Гистограмма изображения со светом (для текущего объекта)

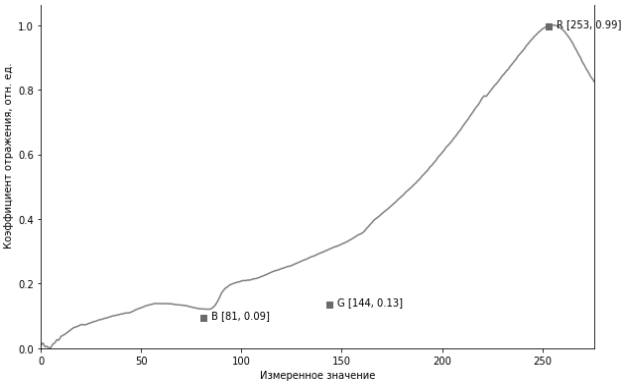


Рис. 9. Спектральная функция и полученные значения

Большое отклонение по зеленому каналу цвета обосновывается тем, что гистограмма кирпича по зеленому цвету имеет 2 примерно равные вершины, а не одну (одна вершина имела бы большее значение и отклонение было бы меньше), что может говорить о наличии дефекта цвета для данного кирпича.

Заключение

По результатам проведенной работы можно сделать вывод, что данная методика является легкореализуемой и достаточной для расчёта значений, необходимых для анализа и оценки цветового качества кирпичной продукции. Рассмотрены реальные значения гистограмм изображений кирпича и света, на основании которых были проведены расчёты в соответствии с методикой.

Литература

1. Стратегия развития строительной отрасли и жилищно-коммунального хозяйства Российской Федерации до 2030 года с прогнозом на период до 2035 года. – М.: Минстрой РФ, 2019. – 82 с.

2. Керамический кирпич и его место в строительстве современных зданий / А.И. Ананьев, О.И. Лобов // Проектирование и строительство в Сибири. – 2013. – 241 с.

3. ГОСТ 530–2012. Кирпич и камень керамические. Общие технические условия. – М.: ВНИИСТРОМ «Научный центр керамики», 2013. – 49 с.

4. Обработка и анализ изображений в задачах машинного зрения / Ю.В. Визильтер, С.Ю. Желтов, А.В. Бондаренко. – М.: Физматкнига, 2010. – 672 с.

5. Понс Ж. Компьютерное зрение // Современный подход. – М.: ИД Вильямс, 2004. – 465 с.

Карпов Роман Константинович
Студент каф. АСУ ТУСУРа
Эл. почта: doitlaunot@gmail.com

Катаев Михаил Юрьевич
Д-р техн. наук, профессор каф. АСУ ТУСУРа
Эл. почта: kmy@asu.tusur.ru

УДК 681.523.4

В.Т. Тран, А.М. Кориков

Управление с использованием нейронной сети RBF трехосным карданным подвесом на беспилотных летательных аппаратах

Выполнено моделирование и управление трехосным карданным подвесом (ТКП) на беспилотных летательных аппаратах (БПЛА). Синтезирован адаптивный скользящий режим управления с использованием нейронной сети RBF для модели ТКП. Для моделирования работы системы управления ТКП использовано программное обеспечение MATLAB SIMULINK. Выполнено моделирование работы системы отслеживания местоположения ТКП и движения ТКП при воздействии неизвестных внешних сил. Дана оценка работоспособности системы управления ТКП, находящегося под влиянием случайных помех.

Ключевые слова: моделирование, карданный подвес, беспилотный летательный аппарат, синтез, адаптивное скользящее управление, нейронная сеть, моделирование, SIMULINK.

Трехосный карданный подвес (ТКП) с камерой для отслеживания цели широко используется во многих областях: промышленность, сельское хозяйство, транспорт и т.п. [1]. ТКП часто устанавливается на автомобилях, кораблях, самолетах и других транспортных средствах. Имея простую и гибкую структуру, ТКП в основном применяется для установки такого оборудования, как камеры, оптические системы, радары и т.п. Одним из наиболее популярных применений ТКП является его использование для подвески камер на беспилотных устройствах: беспилотных летательных аппаратах (БПЛА), беспилотных автомобилях и т.д.

Однако увеличение количества осей управления усложняет моделирование и расчет ТКП.

Чтобы преодолеть отмеченные сложности, в нашей статье предлагается использовать адаптивный скользящий регулятор (АСР) с искусственной нейронной сетью RBF для идентификации и аппроксимации неопределенных параметров воздушной среды, что улучшит качество работы системы.

Кинетическая модель трехосного карданного подвеса

ТКП состоит из следующих основных частей: рамка движения канала рыскания (РДКР), рамка движения канала крена (РДКК), рамка движения канала тангажа (РДКТ). Отметим, что РДКР стабилизирует камеру по каналу рыскания, РДКК – по каналу крена, а РДКТ – по каналу тангажа (рис. 1).

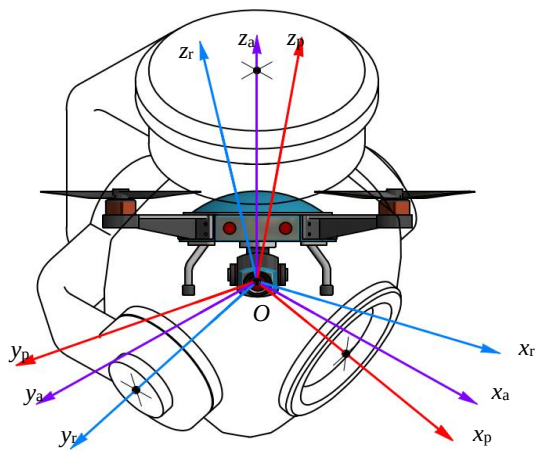


Рис. 1. Трехосный подвес с камерой на БПЛА

Система координат $Ox_a y_a z_a$ прикреплена к РДКР; система координат $Ox_r y_r z_r$ прикреплена к РДКК; система координат $Ox_p y_p z_p$ прикреплена к РДКТ.

РДКР может свободно вращаться вокруг оси Oz_a, но не может вращаться вокруг осей Ox_a и Oy_a; РДКК может свободно вращаться вокруг оси Ox_r, но не может вращаться вокруг осей Oy_r и Oz_r; РДКТ может свободно вращаться вокруг оси Oy_p, но не может вращаться вокруг осей Ox_p и Oz_p [1].

При описании математической модели ТКП используется второй закон Ньютона. Далее при записи уравнений движения для ТКП используются обозначения, введенные в [2]. Итак, модель движения для ТКП описывается следующей системой уравнений:

$$\begin{cases} I_{RSxx}\ddot{\beta} = T_{Rx} - \left(\ddot{\alpha}\cos\beta + \dot{\alpha}\dot{\beta}\sin\beta\right) I_{RSxz} + \\ + \frac{1}{2}\left(I_{RSyy} - I_{RSzz}\right)\dot{\alpha}^2 \sin 2\beta - T_{Px}\cos\gamma + T_{Pz}\sin\gamma + T_{Dx}; \\ I_{yy}\ddot{\gamma} = T_{Py} - I_{yy}\ddot{\alpha}\sin\beta + \left(I_{zz} - I_{xx}\right)\left(\dot{\alpha}\dot{\beta}\left(1 - 2\sin^2\gamma\right)\right) \times \\ \times\cos\beta - \frac{1}{2}\left(\dot{\alpha}^2\cos^2\beta - \dot{\beta}^2\right)\sin 2\gamma\Big) + T_{Dy}; \\ I_{ASzz}\ddot{\alpha} = T_{Az} - (T_{Ry} - T_{Py})\sin\beta - \\ -\left(T_{Rz} + \left(T_{Px}\sin\gamma + T_{Pz}\cos\gamma\right)\right)\cos\beta + T_{Dz}, \end{cases} \quad (1)$$

где α – угол поворота системы координат $Ox_a y_a z_a$ относительно системы координат $Ox_b y_b z_b$ вокруг оси Oz_a (угол поворота РДКР вокруг оси Oz_a); β – угол поворота системы координат $Ox_r y_r z_r$ относительно системы координат $Ox_a y_a z_a$ вокруг оси Ox_r (угол поворота РДКК вокруг оси Ox_r); γ – угол поворота системы координат $Ox_p y_p z_p$ относительно системы координат $Ox_r y_r z_r$ вокруг оси Oy_p (угол поворота РДКТ вокруг оси Oy_p). Система координат $Ox_b y_b z_b$ – система координат, устанавливаемая на устройство БПЛА; $\mathbf{T}_A = [T_{Ax}\ T_{Ay}\ T_{Az}]^T$ – внешний момент силы, действующий на РДКР; $\mathbf{T}_R = [T_{Rx}\ T_{Ry}\ T_{Rz}]^T$ – внешний момент силы, дей-

ствующий на РДКК; $\mathbf{T}_P = [T_{Px}\ T_{Py}\ T_{Pz}]^T$ – внешний момент силы, действующий на РДКТ; $\mathbf{J}_P$ – момент инерции РДКТ:

$$\mathbf{J}_P = \begin{bmatrix} I_{Pxx} & 0 & 0 \\ 0 & I_{Pyy} & 0 \\ 0 & 0 & I_{Pzz} \end{bmatrix};$$

$\mathbf{J}_{RS}$ – сумма моментов инерции РДКТ и РДКК:

$$\mathbf{J}_{RS} = \begin{bmatrix} I_{RSxx} & 0 & I_{RSxz} \\ 0 & I_{RSyy} & 0 \\ I_{RSxz} & 0 & I_{RSzz} \end{bmatrix},$$

где

$$I_{RSxx} = I_{Rxx} + I_{Pzz} + (I_{Pxx} - I_{Pzz})\cos^2\gamma;$$
$$I_{RSxz} = (I_{Pzz} - I_{Pxx})\sin\gamma\cos\gamma;$$
$$I_{RSyy} = I_{Ryy} + I_{\mathrm{Pyy}};$$
$$I_{RSzz} = I_{Rzz} + I_{Pxx} + (I_{Pzz} - I_{Pxx})\cos^2\gamma;$$

$\mathbf{J}_R$ – момент инерции РДКК:

$$\mathbf{J}_R = \begin{bmatrix} I_{Rxx} & 0 & 0 \\ 0 & I_{Ryy} & 0 \\ 0 & 0 & I_{Rzz} \end{bmatrix};$$

$\mathbf{J}_{AS}$ – сумма моментов инерции РДКТ, РДКК и РДКР:

$$\mathbf{J}_{AS} = \begin{bmatrix} I_{ASxx} & I_{ASxy} & I_{ASxz} \\ I_{ASxy} & I_{ASyy} & I_{ASyz} \\ I_{ASxz} & I_{ASyz} & I_{ASzz} \end{bmatrix},$$

где

$$I_{ASxx} = I_{Axx} + I_{RSxx};\ \ I_{ASxy} = -I_{RSxz}\sin\beta;$$
$$I_{ASxz} = I_{RSxz}\cos\beta;$$
$$I_{ASyy} = I_{Ayy} + I_{RSyy}\cos^2\beta + I_{RSzz}\sin^2\beta;$$
$$I_{ASyz} = I_{RSyy}\cos\beta\sin\beta\ - I_{RSzz}\cos\beta\sin\beta;$$
$$I_{ASzz} = I_{Azz} + I_{RSzz}\cos^2\beta + I_{RSyy}\sin^2\beta.$$

$\mathbf{J}_A$ – момент инерции РДКР:

$$\mathbf{J}_A = \begin{bmatrix} I_{Axx} & 0 & 0 \\ 0 & I_{Ayy} & 0 \\ 0 & 0 & I_{Azz} \end{bmatrix}.$$

Синтез адаптивного скользящего режима управления с использованием нейронной сети RBF для ТКП

Из системы уравнений (1) зададим переменные состояния:

$x_1 = \beta,\ \dot{x}_1 = x_2,\ x_3 = \gamma,\ \dot{x}_3 = x_4,\ x_5 = \alpha,\ \dot{x}_5 = x_6.$

Пусть T_{Dx}, T_{Dy}, T_{Dz} – неизвестные факторы возмущения, влияющие на ТКП (влияние гравитации на рабочий процесс ТКП, влияние центробежной силы инерции рамок и т.п.), тогда система уравнений (1) перепишется следующим образом:

$$\begin{cases} \dot{x}_1 = x_2; \\ \dot{x}_2 = \dfrac{1}{I_{RSxx}}\Big[T_{Rx} - \big(\dot{x}_6\cos x_1 + x_1 x_6 \sin x_1\big) I_{RSxz} + \\ + \dfrac{1}{2}\big(I_{RSyy} - I_{RSzz}\big) x_6^2 \sin 2x_1 - T_{Px}\cos x_3 + T_{Pz}\sin x_3 + T_{Dx}\Big]; \\ \dot{x}_3 = x_4; \\ \dot{\gamma} = \dfrac{1}{I_{yy}}\Big[T_{Py} - I_{yy}\dot{x}_6\sin x_1 + \big(I_{zz} - I_{xx}\big)\big(x_1 x_6 \times \\ \times(1 - 2\sin^2 x_3)\cos\beta - \dfrac{1}{2}\Big(x_6^2\cos^2 x_1 - x_2^2\Big)\sin 2x_3\Big) + T_{Dy}\Big]; \\ \dot{x}_5 = x_6; \\ \dot{x}_6 = \dfrac{1}{I_{ASzz}}\Big[T_{Az} - \big(T_{Ry} - T_{Py}\big)\sin x_1 - \\ -\big(T_{Rz} + \big(T_{Px}\sin x_3 + T_{Pz}\cos x_3\big)\big)\cos x_1 + T_{Dz}\Big]. \end{cases} \quad (2)$$

Задача состоит в определении таких управляющих сигналов T_{Rx}, T_{Py}, T_{Az} чтобы выходной сигнал β, γ, α соответствовал начальному набору сигналов x_{1d}, x_{3d}, x_{5d} ($x_{1d} = \beta_d$, $x_{3d} = \gamma_d$, $x_{5d} = \alpha_d$).

Выберем поверхность скольжения в виде

$$\begin{cases} S_1 = x_2 - \dot{x}_{1d} + k_1(x_1 - x_{1d}), \\ S_2 = x_4 - \dot{x}_{3d} + k_2(x_3 - x_{3d}), \\ S_3 = x_6 - \dot{x}_{5d} + k_3(x_5 - x_{5d}), \end{cases} \quad (3)$$

где k_1, k_2, k_3 – константы, выбранные таким образом, что характеристический многочлен уравнения $S = 0$ удовлетворяет критерию устойчивости Гурвица [3].

Из системы уравнений (6) имеем

$$\begin{cases} \dot{S}_1 = \dot{x}_2 - \ddot{x}_{1d} + k_1(\dot{x}_1 - \dot{x}_{1d}), \\ \dot{S}_2 = \dot{x}_4 - \ddot{x}_{3d} + k_2(\dot{x}_3 - \dot{x}_{3d}), \\ \dot{S}_3 = \dot{x}_6 - \ddot{x}_{5d} + k_3(\dot{x}_5 - \dot{x}_{5d}). \end{cases} \quad (4)$$

Ошибка системы $e = x - x_d$ приближается к нулю, если $S\dot{S}$ <0 или

$$\dot{S} = -K\,\mathrm{sign}(S). \quad (5)$$

Комбинируя системы уравнений (2)–(5), мы получаем следующую систему уравнений для определения управляющих сигналов:

$$\begin{cases} T_{Rx} = -f_1(x) - f_1'(x) + I_{RSxx}\big(\ddot{x}_{1d} - k_1\dot{e}_1\big) - I_{RSxx}K_1\,\mathrm{sign}(S_1), \\ T_{Py} = -f_3(x) - f_3'(x) + I_{Pyy}\big(\ddot{x}_{3d} - k_2\dot{e}_2\big) - I_{Pyy}K_2\,\mathrm{sign}(S_2), \\ T_{Az} = -f_5(x) - f_5'(x) + I_{ASzz}\big(\ddot{x}_{5d} - k_3\dot{e}_3\big) - I_{ASzz}K_3\,\mathrm{sign}(S_3), \end{cases} \quad (6)$$

где $f_1(x) = \big(\dot{x}_6\cos x_1 + x_1 x_6\sin x_1\big) I_{RSxz} - \dfrac{1}{2}\Big(I_{RSyy} -$

$-I_{RSzz}\big)x_6^2\sin 2x_1 + T_{Px}\cos x_3 - T_{Pz}\sin x_3;$

$f_2(x) = I_{yy}\dot{x}_6\sin x_1 + \big(I_{zz} - I_{xx}\big)\big(x_1 x_6(1 - 2\sin^2 x_3)\cos\beta +$

$+\dfrac{1}{2}\Big(x_6^2\cos^2 x_1 - x_2^2\Big)\sin 2x_3\Big);$

$$f_3(x) = \big(T_{Ry} - T_{Py}\big)\sin x_1 - T_{Rz}\cos x_1 -$$
$$-\big(T_{Px}\sin x_3 + T_{Pz}\cos x_3\big)\cos x_1;$$
$$f_1'(x) = T_{Dx};\quad f_2'(x) = T_{Dy};\quad f_3'(x) = T_{Dz};$$

$e_1 = x_1 - x_{1d}$; $e_2 = x_3 - x_{3d}$; $e_3 = x_5 - x_{5d}$.

В (6) функции $f_1'(x), f_2'(x), f_3'(x)$ содержат неизвестные параметры: влияние гравитации на рабочий процесс ТКП, центробежной силы инерции рамок и т.п.

Для аппроксимации функций $f_1'(x), f_2'(x), f_3'(x)$ в скользящем законе управления используется нейронная сеть RBF. Адаптивный закон определяется на основе теории устойчивости Ляпунова [3].

В нашем исследовании используются 3 нейронные сети RBF для аппроксимации функции $f_1'(x), f_2'(x), f_3'(x)$. Каждая нейронная сеть RBF содержит 2 нейрона входного слоя, 5 нейронов скрытого слоя и 1 нейрон выходного слоя. Входы нейронных сетей – это сигналы x_1, x_2, x_3, x_4, x_5, x_6. Выходные данные нейронных сетей RBF – это функции $\hat{f}_1(x), \hat{f}_2(x), \hat{f}_3(x)$, которые передаются в контроллер АСР для расчета управляющего сигнала в соответствии с системой уравнений (6). Радиальная базисная функция (гауссоида) RBF-сети в скрытых узлах, обозначенная далее через h_j, является нелинейной функцией активации (ФА) и определяется формулой

$$h_j = \exp\left(-\frac{\|\mathbf{x} - c_j\|^2}{2b_j^2}\right).$$

Выходной сигнал RBF-сети определяется по формуле

$$f = \mathbf{W}^{*T}\mathbf{h}(\mathbf{x}) + \varepsilon, \qquad (7)$$

где $\mathbf{x}$ – входной сигнал нейронной сети RBF, $\mathbf{h}(\mathbf{x})$ – вектор ФА, $\mathbf{W}^*$ – вектор идеальных значений весов сети, погрешность сети $\varepsilon \le \varepsilon_N$.

В нейронной сети RBF следующие параметры: вектор входных сигналов $\mathbf{x} = [x_1, x_2]^T$, вектор ФА в узлах скрытого слоя $\mathbf{h} = [h_1, h_2, h_3, h_4, h_5]^T$, вектор радиусов сети $\mathbf{b} = [b_1, b_2, b_3, b_4, b_5]^T$, вектор весов сети $\mathbf{W} = [w_1, w_2, w_3, w_4, w_5]^T$, матрица центров ФА сети

$$\mathbf{c} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} \\ c_{21} & c_{22} & c_{23} & c_{24} & c_{25} \end{bmatrix}.$$

Выход нейронной сети RBF при аппроксимации функций определяется по формуле

$$\hat{f}(x) = \hat{\mathbf{W}}^T\mathbf{h}(\mathbf{x}), \qquad (8)$$

где $\hat{\mathbf{W}}$ – матрица оценочных весов нейронной сети.

После аппроксимации функций $f_1'(x), f_2'(x), f_3'(x)$ нейронной сетью RBF по формуле (8) управляющие сигналы формируются в соответствии со следующей системой уравнений:

$$\begin{cases} T_{Rx} = -f_1(x) - \hat{f}_1(x) + I_{RSxx}\left(\ddot{x}_{1d} - k_1\dot{e}_1\right) - I_{RSxx}K_1\,\mathrm{sign}(S_1), \\ T_{Py} = -f_2(x) - \hat{f}_2(x) + I_{Pyy}\left(\ddot{x}_{3d} - k_2\dot{e}_2\right) - I_{Pyy}K_2\,\mathrm{sign}(S_2), \\ T_{Az} = -f_3(x) - \hat{f}_3(x) + I_{ASzz}\left(\ddot{x}_{5d} - k_3\dot{e}_3\right) - I_{ASzz}K_3\,\mathrm{sign}(S_3), \end{cases} \qquad (9)$$

где

$$\begin{cases} \hat{f}_1(x) = \hat{\mathbf{W}}_1^T\mathbf{h}(x_1, x_2), \\ \hat{f}_2(x) = \hat{\mathbf{W}}_2^T\mathbf{h}(x_3, x_4), \\ \hat{f}_3(x) = \hat{\mathbf{W}}_3^T\mathbf{h}(x_5, x_6), \end{cases} \qquad (10)$$

$\hat{\mathbf{W}}_1; \hat{\mathbf{W}}_2; \hat{\mathbf{W}}_3$ определяется по формулам

$$\begin{cases} \dot{\hat{\mathbf{W}}}_1 = \lambda S_1\mathbf{h}(x_1, x_2), \\ \dot{\hat{\mathbf{W}}}_2 = \lambda S_2\mathbf{h}(x_3, x_4), , \\ \dot{\hat{\mathbf{W}}}_3 = \lambda S_3\mathbf{h}(x_5, x_6), \end{cases} \qquad (11)$$

где λ – коэффициент адаптации нейронной сети.

Моделирование и оценка результатов

Моделирование в программной среде MATLAB SIMULINK [4] выполнено на основе уравнений (2)–(6), (9)–(11). Значения параметров модели приведены в таблице.

Значения параметров модели

Символ	Значение	Единица
I_{Pxx}	0,00072	кгм^2
I_{Pyy}	0,00064	кгм^2
I_{Pzz}	0,00081	кгм^2
I_{Rxx}	0,00016	кгм^2
I_{Ryy}	0,00013	кгм^2
I_{Rzz}	0,00027	кгм^2
I_{Axx}	0,00100	кгм^2
I_{Ayy}	0,00120	кгм^2
I_{Azz}	0,00032	кгм^2
k_1, k_2, k_3	3	–
K_1, K_2, K_3	20	–

Параметры нейронной сети RBF: матрица радиусов сети $b = 5$, коэффициент адаптации $\lambda = 23$, матрица центров ФА сети:

$$\boldsymbol{c} = \begin{bmatrix} -1 & -0{,}5 & 0 & 0{,}5 & 1 \\ -1 & -0{,}5 & 0 & 0{,}5 & 1 \end{bmatrix}.$$

Результаты моделирования работы ТКП при изменении его положения представлены на рис. 2–10.

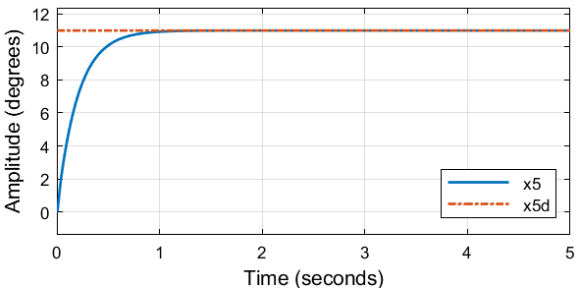


Рис. 2. Реакция ТКП на изменение угла α
α_d = 11° при использовании АСР

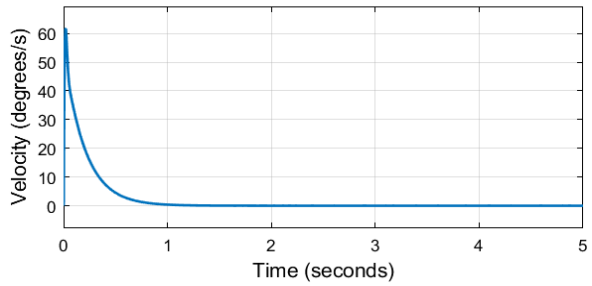


Рис. 3. Реакция ТКП на угловую скорость $\dot{\alpha}$ при использовании АСР

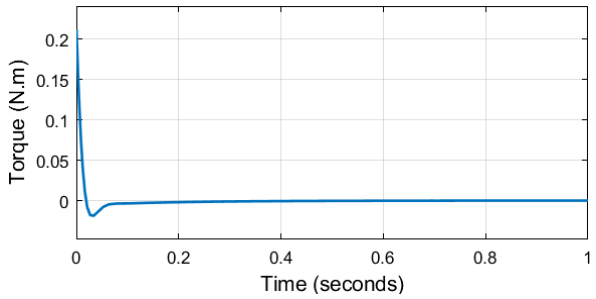


Рис. 4. Реакция ТКП на управляющий сигнал T_{Az} при использовании АСР

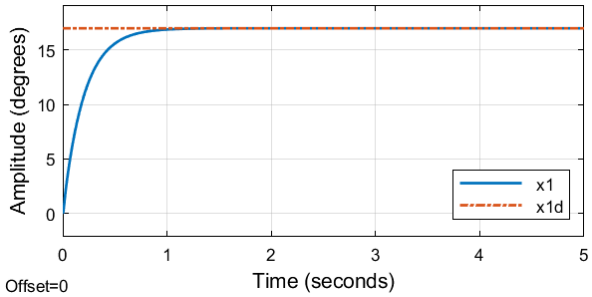


Рис. 5. Реакция ТКП на изменение угла β β_d = 17° при использовании АСР

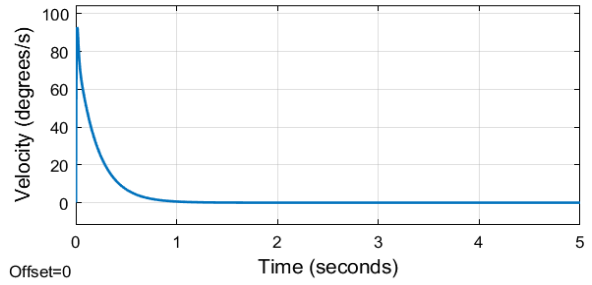


Рис. 6. Реакция ТКП на угловую скорость $\dot{\beta}$ при использовании АСР

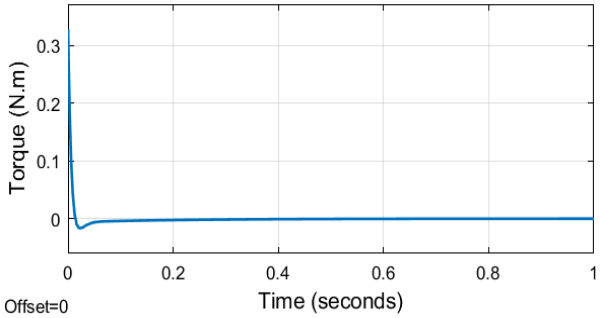


Рис. 7. Реакция ТКП на управляющий сигнал T_{Rx} при использовании АСР

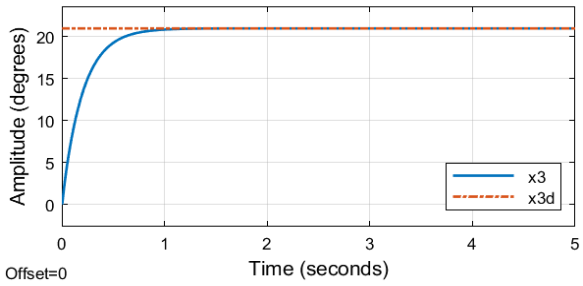


Рис. 8. Реакция ТКП на изменение угла γ γ_d = 21° при использовании АСР

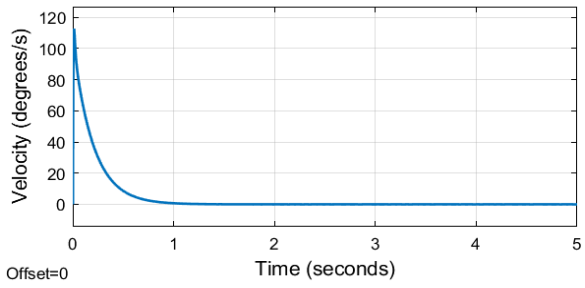


Рис. 9. Реакция ТКП на угловую скорость $\dot{\gamma}$ при использовании АСР

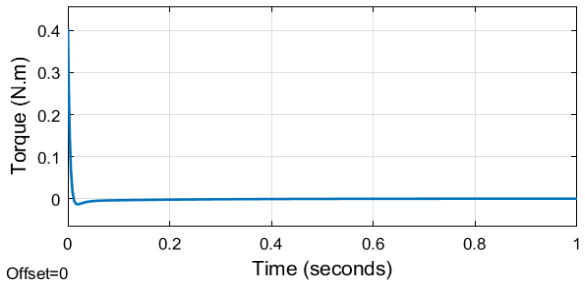


Рис. 10. Реакция ТКП на управляющий сигнал T_{Py} при использовании АСР

Из рис. 2–10 следует, что система с адаптивным скользящим регулятором имеет устойчивое (стабильное) рабочее состояние, время стабилизации системы находится в пределах 1,5– 2 с, время перехода в стабильное состояние незначительное, ошибка настройки (ошибка между α, β, γ и $\alpha_d, \beta_d, \gamma_d$) мала (незначима), система устойчива и стабильно работает в рабочем режиме.

Результаты моделирования работы ТКП при длительном воздействии на систему неизвестных внешних моментов постоянной величины представлены на рис. 11, 12.

На рис. 11, 12 показано, что с входными сигналами системы $\beta_d = 17°$, $\gamma_d = 21°$ прикладывается внешний момент 0,01 $N.m$ в направлении Ox_r и Oy_p. Из рис. 11, 12 следует, что классический скользящий регулятор (КСР) не распознает действующий внешний момент, изменяющий параметры системы. Объект управления (ТКП) под действием внешней силы изменил своё положение: координаты ТКП изменились и ТКП стабилизировался на новом положении. Однако адаптивный скользящий регулятор, использующий нейронную сеть RBF, идентифицировал фактор изменчивости (внешний

момент) и соответствующим образом скорректировал управляющий сигнал. Система управления ТКП была скорректирована, и БПЛА стабилизируется в исходном положении. Этот факт имеет важное значение для управления ТКП в условиях длительного воздействия ветра и силы тяжести, вызванных отклонением центра тяжести ТКП.

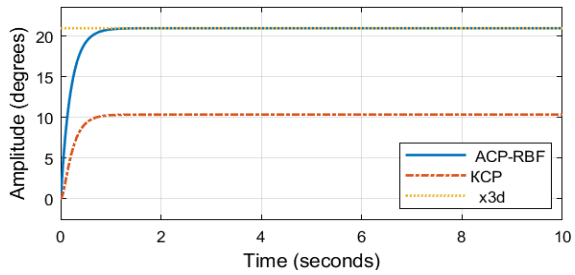


Рис. 11. Реакция ТКП на возмущение по каналу тангажа при использовании АСР

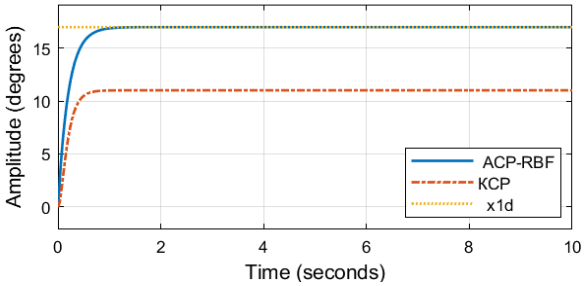


Рис. 12. Реакция ТКП на возмущение по каналу крена при использовании АСР

Для проверки работы ТКП в условиях помех проведем проверку работы ТКП на РДКР при положении $\alpha_d = 11°$. В систему подается неопределенный случайный сигнал. Результаты моделирования ТКП показаны на рис. 13.

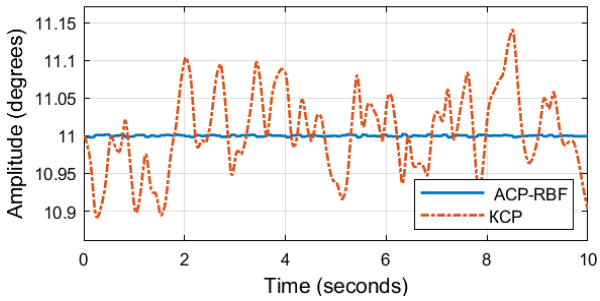


Рис. 13. Изменение положения ТКП по каналу рыскания при воздействии на него случайной помехи

Из рис. 13 видно, что при адаптивном скользящем регуляторе ТКП достаточно эффективно работает при наличии возмущений. При использовании классического скользящего регулятора на ТКП сильно влияет шум.

Заключение

В работе построена математическая модель ТКП. Синтезирован адаптивный скользящий режим управления с использованием нейронной сети RBF. Результаты моделирования ТКП в программной среде MATLAB SIMULINK показывают, что система работает достаточно стабильно в условиях непредсказуемого изменения параметров системы, таких как гравитация, вызывающая отклонение центра тяжести в процессе движения, центробежная сила инерции, возникающая при вращении. Кроме того, система достаточно стабильно работает и при наличии случайных помех.

Литература

1. Тран В.Т. Моделирование трехосного карданного подвеса на беспилотных летательных аппаратах // Сборник избранных статей научной сессии ТУСУР: междунар науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР–2022» 18–20 мая 2022 г., г. Томск: в 3 ч. – Ч. 2. – Томск: В-Спектр, 2022. – С. 30–32.

2. Тран В.Т. Выбор регулятора, работающего в скользящем режиме, для автоматизированной транспортной системы / В.Т. Тран, А.М. Кориков, Т.Т. Нгуен // Доклады ТУСУР. – 2021. – Т. 24, № 4. – С. 79–84.

3. Кориков А.М. Основы теории управления: учеб. пособие. – 2-е изд. – Томск: Изд-во НТЛ, 2002. – 392 с.

4. Дорф Р. Современные системы управления / Р. Дорф, Р. Бишоп. – М.: Лаборатория базовых знаний, 2004. – 832 с.

Тран Ван Трук
Аспирант каф. АСУ ТУСУРа
Эл. почта: att82glass@gmail.com

Кориков Анатолий Михайлович
Д-р техн. наук, профессор каф. АСУ ТУСУРа
Эл. почта: korikov@asu.tusur.ru

УДК 621

А.Б. Гомбоин, А.Б. Кураков

Серверная часть веб-системы оценки походки человека по скелетной модели на языке Dart

Благодаря анализу походки человека можно узнать о многих особенностях как физического, так и эмоционального состояния человека. На сегодняшний день существует различное множество методов изучения походки человека, как, например, скелетная модель, исследование походки с помощью центра тяжести и др. Для получения данных о походке человека используются различные технологии: датчики захвата движения, лазеры и видеофиксация движения человека. Первые технологии часто применяются в клинических и фундаментальных исследованиях, но они достаточно дорогостоящие и требуют специализированных знаний и опыта. Данная веб-система использует видеофиксацию для анализа походки человека, что поможет избежать проблем с дороговизной и отсутствием в специальных знаниях [1]. Для анализа походки используется нейронная сеть с открытым исходным кодом и уже предобученными данными.

Для того чтобы пользователи могли без проблем и сложностей взаимодействовать с данной нейронной сетью и получать уже готовую скелетную модель, необходима серверная часть.

Ключевые слова: веб-сервер, скелетная модель, сервер, Dart, API, база данных, БД, библиотека, Shelf.

Для написания серверной части использовался язык программирования Dart.

Данный язык относительно молодой, впервые он появился в 2007 г., но стабильная версия выпущена в 2017 г. компанией Google. Dart – строго типизированный и C-подобный язык. Он используется для разработки веб-приложений, мобильных приложений, серверного программного обеспечения, десктопных приложений и некоторых устройств IoT.

Dart является объектно-ориентированным языком, основан на классах и поддерживает наследование миксинов. Необходимо учесть несколько моментов перед работай с данным языком: все является объектом, каждый объект является экземпляром класса, все объекты наследуются от класса Object; присутствует проверка статических типов при их использовании при помощи специальных инструментов языка; Dart анализирует весь код перед запуском.

Для взаимодействия сервера с уже готовой клиентской частью, которая была написана с помощью языка Dart и фреймворка Flutter, необходимо реализовать API [2, 3].

API – небольшие фрагменты кода, которые позволяют цифровым устройствам, программным приложениям и серверам данных взаимодействовать друг с другом и использовать данные функции друг друга через документированный интерфейс. API-интерфейсы находятся между приложением и веб-сервером, выступая в качестве промежуточного уровня, который обрабатывает передачу данных между системами. Пример работы: клиентское приложение инициирует вызов, также называется запрос API для получения информации; после получения действительного запроса API выполняет вызов внешней программы или веб-сервера; сервер отправляет ответ API с запрошенной информацией; API передает данные первоначальному запрашиваемому приложению [4–6].

Цель – разработка API, которая удовлетворяла бы требованиям для работы с веб-системой.

Для достижения поставленной цели были выделены следующие задачи:

1. Поиск необходимых библиотек для написания API.
2. Создать базу данных
3. Описать необходимые методы API.

Поиск необходимых библиотек

Для создания API на Dart была найдена и использована библиотека Shelf. Она упрощает создание и компоновку веб-серверов и частей веб-серверов.

Данная библиотека также имеет большое количество дополнений, которые расширяют возможности библиотеки.

Одно из дополнений – shelf-router, которое позволяет легко и удобно настроить маршрутизацию API и предоставляет необходимые методы REST, например такие, как GET, POST, DELETE и т.д.

Также было использовано ещё одно дополнение, которое необходимо для работы с файлами видео, – shelf-static. Данное дополнение дает возможность обращаться к файлу на сервере и тем самым генерировать ссылку для дальнейшей работы с ним.

Для более удобного использования методов API использовалось дополнение shelf-cors-headers, которое заранее прописывает заголовки CORS, без которых данные не передаются через браузер пользователю к клиентской части.

Для более лучшей безопасности данных пользователей было решено использовать JWT (JSON Web Token). Данная технология позволяет генерировать на сервере токен, который содержит относящуюся к пользователю информацию в формате JSON. При вызове конкретного метода будет отправляться токен пользователя, чтобы API могло идентифицировать пользователя и выполнить конкретное действие. Для реализации на языке Dart была использована библиотека dart-jsonwebtoken, которая выполняет все поставленные перед ней задачи, а именно кодирует и декодирует данные пользователя с помощью секретного ключа, тем самым создавая токен.

Создание базы данных

Веб-система будет предоставляться большому количеству людей, для этого необходимо позаботиться о том, как и где будет храниться вся информация каждого пользователя. Поэтому было решено внедрить базу данных в веб-сервер.

База данных – программное обеспечение, которое позволяет хранить информацию и в структурированном виде обрабатывать её.

В ходе разработки API было решено использовать СУБД PostgreSQL. Данная СУБД была выбранна, так как имеет открытый исходный код, большое количество примеров с пояснениями и различную документацию на русском языке. Также данная СУБД позволяет гибко управлять базами данных. Благодаря ей можно создавать, улучшать, изменять или удалять записи, отправлять транзакцию [7, 8].

В ходе разработки БД были проанализированы данные, которые необходимо хранить. Были выделены сущности и созданы таблицы, часть из которых представлена на рис. 1.

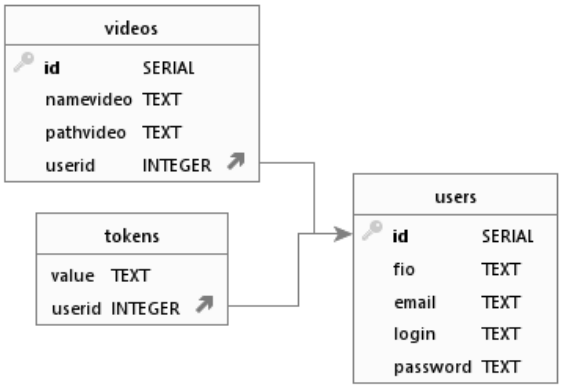


Рис. 1. Схема части БД

Методы API

1. Для регистрации нового пользователя. Данный метод подключается к базе данных и проверяет, правильно ли были переданы данные от клиента, далее происходит проверка на то, что идентичных данных нет в базе данных, если все данные успешно записываются в БД и клиенту отправляется ответ с кодом 200 и уникальным токеном.

2. Для авторизации пользователя. В данном методе для начала проверяются данные на пустоту и далее на правильность ввода логина и пароля пользователем, если данные были введены верно, пользователю отправляется ответ с кодом 200 и уникальный токен. Данные для проверки берутся из базы данных.

3. Для загрузки видео на сервер. Первоначально метод подключается к базе данных, далее проверяет токен пользователя, после успешной проверки начинается передача видеофайла на веб-сервер с помощью формата multipart/form-data. Данный формат позволяет принять на сервер различные файлы, в том числе и видеофайлы, для этого была использована библиотека shelf-multipart. Переданное имя файла от пользователя проверяется с данными в БД на дубликат и, если название уникально для данного пользователя, видео сохраняется на сервере и в БД записывается название файла и путь к нему.

4. Для выгрузки видео на сторону клиента. Данный метод подключается к базе данных и проверяет токен пользователя. После успешного подключения к БД и проверки токена используется условный оператор, который проверяет переданный аргумент в ссылке, что пользователь хочет получить названия всех его загруженных видео или просмотреть одно из них. Если пользователь хочет получить все названия видеофайлов, то происходит запрос в базу данных на получение имен всех имеющихся видео у пользователя и отправляется клиенту. Если просмотреть конкретное видео, то пользователю отправляется путь к файлу, который был получен из запроса к базе данных.

5. Для удаления видеофайла. Первые этапы идентичны остальным методам: проверка токена и подключение к БД. Далее по id, полученному от пользователя, идёт обращение к базе данных с запросом на удаление видео. После успешного удаления пользователю отправляется ответ с кодом 200.

6. Для построения скелетной модели и метод для отправки полученного видео пользователю. Первый метод имеет аналогичное начало, как и другие методы. Далее метод обрабатывает полученное от пользователя название видео, по которому будет строиться скелетная модель. Выбранное видео отправляется на вход нейронной сети, которая будет строить скелетную модель и сохранит получившееся видео со скелетной моделью на сервере и отправит пользователю ответ с кодом 200 о выполненной работе. Второй метод открывает получившееся видео со скелетной моделью и покадрово отправляет пользователю.

Заключение

В результате данной работы был разработан API-интерфейс на языке программирования Dart и семейства библиотек Shelf. К данной API подключена нейронная сеть для построения скелетной модели. Реализована база данных для хранения информации пользователей с помощью СУБД Postgresql.

Поставленная перед нами задача была полностью выполнена. Полученные результаты будут использоваться в дальнейшей разработке данного программного обеспечения.

Литература

1. Методика построения фронтальной скелетной модели / М.Ю. Катаев, Н.Г. Катаева, А.П. Коробко, Т.М. Шаймарданов [Электронный ресурс]. – Режим доступа: https://journal.tusur.ru/storage/97952/23-Катаев-Катаева-Коробко-Шаймарданов.pdf?1516335483, свободный (дата обращения: 14.09.2022).

2. Учебник по симулятору с flutter [Электронный ресурс]. – Ружим доступа: https://books.google.ru/books?hl=ru&lr=&id=extDwAAQBAJ&oi=fnd&pg=PR21&dq=flutter+mobile+app&ots=YVMte3ejFi&sig=qsGRYs1F4YHKGq1g6PnqST8Um8&redir_esc=y#v=onepage&q=flutter%20mobile%20app&f=false, свободный (дата обращения: 14.09.2022).

3. Bracha G. The Dart programming lanluage // Addison-Wesley Professional. – 2015. – 92 c.

4. Manuaba I.B.P., Rudiastini E. API REST web service and backend system of lecture's assessment information system on politeknik negeri bali // Journal of Physics: Conference Series. – IOP Publishing, 2018. – Vol. 953, No. 1. – P. 012069.

5. Masse M. REST API design rulebook: designing consistent RESTful web service interfaces // O'Reilly Media, Inc. – 2011. – 93 p.

6. Li L., Chou W. Design and describe REST API without violating REST: A Petri net based approach // 2011 IEEE International Conference on Web Services. – IEEE, 2011. – PP. 508–515.

7. Momjian B. PostgreSQL: introduction and concepts. – New York: Addison-Wesley, 2001. – Vol. 192.

8. Guliato D. et al. PostgreSQL-IE: An image-handling extension for PostgreSQL // Journal of digital imaging. – 2009. – Vol. 22, No. 2. – PP. 149–165.

Гомбоин Александр Булатович
Студент каф. АСУ ТУСУРа
Эл. почта: a.gomboin@mail.ru

Кураков Андрей Борисович
Студент каф. АСУ ТУСУРа
Эл. почта: sniperaft@bk.ru

УДК 004.921

М.Д. Пахмурин, В.С. Шелихов

Разработка программного обеспечения обработки DICOM файлов

Разработан комплекс локальной гипертермии «Феникс-2», предназначенный для лечения онкологических заболеваний путём нагрева (до 45 °С в режиме локальной гипертермии и более 60° в режиме термоабляции). Этот комплекс сейчас применяется в НИИ онкологии Томского НИМЦ. Применение комплекса в режиме локальной гипертермии подразумевает использование игольчатых нагревателей. Соответственно, хирурги хотят заранее предусмотреть места и направление их введения.

Ключевые слова: DICOM, визуализация.

Для моделирования воздействия на ткани человека проводится предварительное сканирование с помощью компьютерного томографа, результат которого записывается в файлы формата DICOM [1].

Нашей целью является создание инструмента для обработки файлов DICOM, построения и отображения трёхмерной модели, моделирования и отображения распространения тепла.

Формат DICOM

В формате DICOM содержатся элементы, состоящие из нескольких частей:

– Tag – номер элемента;

– VR – value representation – обозначение типа данных;

– Value length – длина поля Value;

– Value – поле, содержащее само значение.

Изображение слоя записывается несколькими элементами, основной из которых содержит в себе матрицу. Значение каждого элемента матрицы соответствует плотности ткани в конкретной точке. Остальные элементы, связанные с изображением, содержат в себе метаданные.

Обработка изображений

Для работы с изображениями были реализованы функции изменения яркости и контрастности, функции преобразования, реализуемые с помощью матриц перехода [2] (рис. 1): сдвиг, поворот, растяжение.

Например, для поворота используется переход по формуле

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

где x, y – координаты точки, θ – угол вращения.

Для взаимодействия с разными срезами было реализовано переключение между ними.

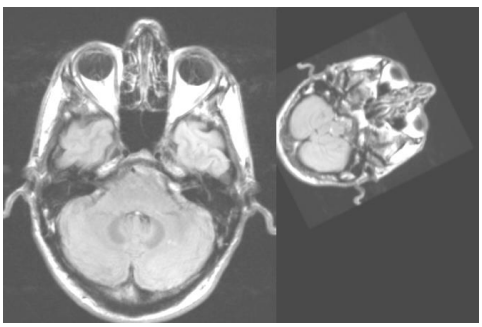


Рис. 1. Вывод изображения слоя

Трёхмерное моделирование

В настоящее время ведется разработка части программы, отвечающей за отрисовку трехмерной модели.

Модель строится путем наложения изображений срезов друг на друга (рис. 2). Так как изображение представляет собой матрицу, в каждой ячейке которой содержится значение цвета, то результат можно интерпретировать как трехмерную кубическую (воксельную) сетку.

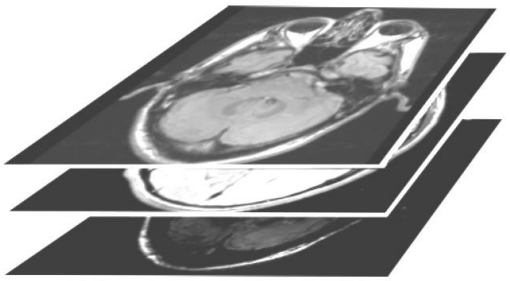


Рис. 2. Создание модели

Для отрисовки модели используется OpenGL – графический API, позволяющий использовать видеокарту для выполнения параллельных вычислений. Для каждого пикселя экрана запускается шейдер – специальная программа, выполняющаяся на видеопроцессоре.

В шейдере описывается алгоритм, который ищет пересечения луча, выпущенного из камеры пользователя на виртуальной сцене, с трехмерной сеткой, с помощью алгоритма Raymarching [3], основанного на функциях расстояния, возвращающих кратчайшее расстояние до поверхности, после чего луч передвигается на шаг в том же направлении на это расстояние, и так до тех пор, пока не произойдет пересечение с поверхностью.

Для поиска пересечений луча внутри самой сетки используется алгоритм Fast Voxel Traversal [4], в котором сначала ищется первое пересечение с сеткой по всем осям, после чего луч перемещается на расстояние, равное шагу сетки (рис. 3).

Такой подход, в отличие от привычного полигонального представления, позволяет избежать перестроения модели при её изменении. Становится возможным отображение полупрозрачных вокселей, уменьшается и используемый объем видеопамяти. Результат представлен на рис. 4.

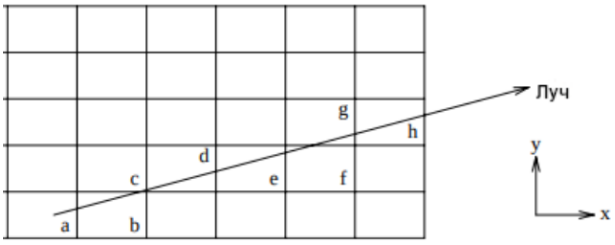


Рис. 3. Двухмерное представления алгоритма FVTA

Рис. 4. Отображение воксельной модели

Заключение

В данной работе была рассмотрена разработка программного обеспечения для обработки DICOM файлов. Разработанное программное обеспечение на данном этапе является лишь прототипом и основой для дальнейшего развития. В будущем планируется реализация генерации модели и моделирование распространения тепла.

Литература

1. DICOM PS3.5 2013. Data Structures and Encoding [Электронный ресурс]. – Режим доступа: https://dicom.nema.org/dicom/2013/output/chtml/part05/PS3.5.html (дата обращения: 15.11.2021).

2. Матрица перехода [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Матрица_перехода (дата обращения: 15.11.2021).

3. Поля расстояний Raymarching-a, Хабр [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/503554/ (дата обращения: 26.05.2022).

4. A Fast Voxel Traversal Algorithm for Ray Tracing, John Amanatides, Andrew Woo [Электронный ресурс]. – Режим доступа: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.42.3443&rep=rep1&type=pdf (дата обращения: 26.05.2022).

Пахмурин Михаил Денисович
Студент каф. АСУ ТУСУРа
Эл. почта: mdpakhmurin@gmail.com

Шелихов Всеволод Сергеевич
Студент каф. АСУ ТУСУРа
Эл. почта: vsevolod-777@mail.ru

УДК 621.396.41

А.В. Соколов

Методика и программа оценки положения головы человека около монитора

В нынешних реалиях ставится множество задач, связанных с определением характеристик человека. Эти данные можно использовать для улучшения работ систем, взаимодействующих с пользователем. Такими системами являются программы, которые нуждаются в отслеживании головы человека. Это могут быть системы правопорядка и контроля, здравоохранения, виртуальная реальность или конференции с видеосвязью. Одним из корректных и подходящих методов является использование базового метода Виолы–Джонса, статистических связей между антропометрическими точками лица и 2D-каркасов.

Ключевые слова: система, определение положения, методика, человеческое лицо, каркасы, лицевые точки.

Существуют методы, опирающиеся на эмпирические признаки [1]. Формируя ряд эвристик, анализируя наличие и взаиморасположение данных, система определяет лицо, его наличие на кадре. Существует своеобразный набор правил. В него зачастую входят понятия того, что лицо подвергается логике симметричности относительно координат, учитывание перепадов света на лице, а также то, что черты лица имеют взаимоположение и размеры (относительные).

Метод сканирующего окна (метод Виолы–Джонса)

Поступающие на вход кадры с видеопотока или одиночные изображения могут иметь объекты, которые выделяются, если определенные признаки в окне сканирования будут совпадать с признаками из выделенной части изображения.

Данный подход к обнаружению комбинируется и может базироваться на концепции простейших функций-признаков прямоугольной формы (признаки Хаара). Это также совмещается с пулом множественных функций при помощи каскадного классификатора.

Алгоритм отмечает найденные объекты, выделяя признаки. Поиск ведется областями при помощи признаков. Полученные данные, полученные на основе анализа зоны изображения признаком, будут записываться в переменную (1):

$$vecP = [x, y, w, h, a], \quad (1)$$

где w и h – ширина и высота изображения, x и y – координаты центра признака, a – угол наклона признака по отношению к вертикальной оси.

Метод является сканирующим окном, который будет применяться к кадру с предположительно расположенным лицом человека, после чего применяется полученный классификатор к каждому из положений и позиций.

Вейвлет Хаара

Функции Хаара реализованы на основе вейвлетов Хаара [2]. Это система, основанная на принципах ортогональных систем функций. Их построение (2) сводится к задаванию родительской вейвлет-функции, которая будет определять детали сигнала.

$$\psi(x) = \begin{cases} 1, & \\ -1, & \text{где} \\ 0, & \end{cases} \begin{cases} 0 \le x < 1, \\ 1/2 \le x < 1, \\ x \notin [0,1). \end{cases} \quad (2)$$

Определим масштабирующую функцию (3), которая будет аппроксимировать сигнал:

$$\varphi(x) = \begin{cases} 1, & \text{где} \\ 0, & \end{cases} \begin{cases} 0 \le x < 1, \\ x \notin [0,1). \end{cases} \quad (3)$$

Признаки Хаара

Методология Виолы–Джонса [3], базирующаяся на признаках Хаара, использует основы из описанных выше вейвлетов. Вейвлеты Хаара представляют собой прямоугольные волны, имеющие одинаковую длину. Выражая это в двумерном виде, прямоугольная волна выглядит как два соседних прямоугольника – темный и светлый. В свою очередь, признак Хаара (рис. 1) это и есть подобие вейвлета Хаара.

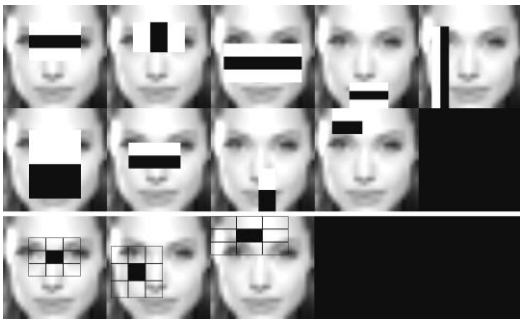


Рис. 1. Вейвлеты Хаара на лице человека

Упор идет на противопоставлении темных и светлых областей в конкретных областях. Каждая маска характеризуется размерами темных и светлых зон, их пропорциями. Эти признаки дают точечное значение для перепадов яркости. Значения признака представляются как разность суммы значений яркостей точек, которые покрываются светлой зоной признака, и суммы значений яркостей точек, которые покрываются темной зоной признака.

Интегральное представление изображения

Для работы с алгоритмом Виолы–Джонса необходимо перевести данные в интегральное представление (4):

$$M(x,y)=\sum_{i=0;j=0}^{i\le x;\,j\le y} D(i,j). \quad (4)$$

Получаемая матрица представляет собой набор яркостей пикселей изображения. Элементы хранят сумму интенсивностей пикселей, стоящих выше и левее по отношению к взятому элементу.

Схема обучения

Для каждой взятой маски (их определенное число, база масок) и для каждой рассматриваемой конфигурации будет тренироваться слабый конфигуратор при условии, чтобы он давал минимальную ошибку. С каждой итерации обучения отбираются конфигураторы, которые складываются в результирующую базу данных.

Возьмем n за число тестовых изображений. Тогда мы имеем множество тестовых изображений (5) и значения функций (6), отражающих присутствие лица на изображении.

$$X=[x_1,x_2,...,x_n], \quad (5)$$

$$Y=[y_1,y_2,...,y_n]. \quad (6)$$

Таким образом, элемент функции, отвечающий за наличие лица, примет значение единицы, если лицо присутствует. Также введем прикладное значение признака (7):

$$j=[\text{маска},\text{положение},\text{размер}]. \quad (7)$$

Выразим слабый классификатор (8):

$$h_j(x)=\begin{cases}1, \text{ если } p_j f_j(x) < p_j \vartheta_j;\\ 0, \text{ в остальных случаях.}\end{cases} \quad (8)$$

Откликом признака $f_j(x)$ в данном случае будет являться разница пикселей в светлых и темных областях. Используя процедуры обучения, создаются структуры, именуемые сильным классификатором (9), представляющие базу из множества слабых классификаторов:

$$H(x)=\begin{cases}1, \text{ если } \sum_{t=1}^{T} a_t h_j{}^t(x) \ge \frac{1}{2}\sum_{t=1}^{T} a_t;\\ 0, \text{ в остальных случаях.}\end{cases} \quad (9)$$

Определим целевую функцию (10), исходя из поставленных условий поиска классификатора:

$$T,h_j^1,h_j^2,\ ...,a_1,\ ...,a_T =$$

$$=\min\sum_{i=1}^{n}\left|H(x_i,T,h_j^{(1)},\ ...,h_{j(T)},a_1,\ ...,a_T)\right|. \quad (10)$$

Обучение классификатора

Необходимо классифицировать объекты. Это производится путём нумерации или прямого наименования класса, что будет соотносить объект с классом. Определение классификатора заключается в функции обобщения и упрощения, построения описательной функции, которая будет говорить, к какому классу определять объект.

Возьмем множество **X**, которое будет обозначать описательные свойства объекта. Тогда за **Y** мы возьмем конечное множество определенных номеров, которые будут принадлежать классам. Между данными множествами можно провести зависимость типа **X** => **Y**. Обучающей выборкой в нашем случае будет являться база данных (11):

$$X_n=[(x_1,y_1),\ ...,\ (x_n,y_n)]. \quad (11)$$

Для дальнейшего обучения подбирается и формируется функция *f*, что, будучи зависимой от вектора признаков **X**, она будет в качестве результата демонстрировать ответ для выбранного **X** в качестве наблюдения, а также способна отнести к классу объект из множества.

Получение данных трехмерных координат на основе антропометрических точек

Если брать за базовый метод для обнаружения лиц метод Виолы–Джонса, то сам процесс обнаружения особых состояний лица (повороты, наклоны и т.п.) можно определять, опираясь на вытекающие данные из распознанного лица. В данной концепции учитываются статистические связи между расположением антропометрических точек лица (рис. 2) [6].

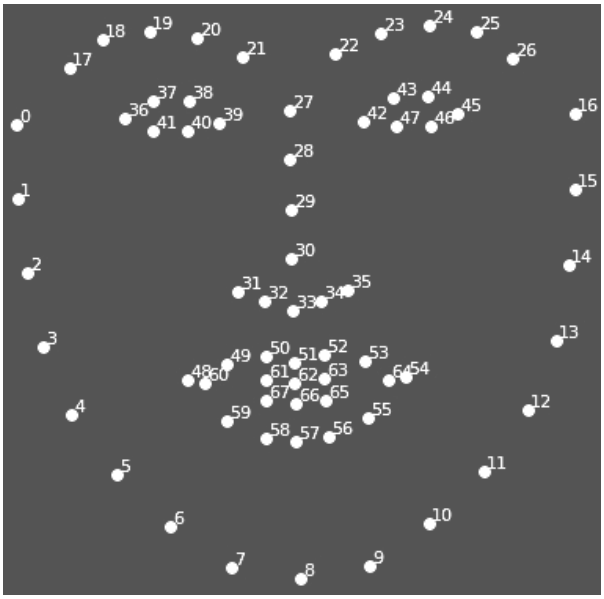


Рис. 2. Пронумерованные точки лица

Конечно, если лица при сравнении находятся в одном и том же положении, можно использовать точечные 2D-каркасы. Однако существуют и точечные 3D-каркасы. Преимущество у данных каркасов состоит в том, что расстояния между определенными точками будут проецироваться на изображение под различными искажениями, учитывая зависимость положения лица относительно камеры. Признаками для идентификации являются определяемые, исходя из статистических связей или расстояний, закономерности, называемые расположениями антропометрических точек лица [4] (рис. 3).

Если рассматривать конкретную ситуацию, то, например, при повороте головы человека расстояние по вертикали от подбородка до переносицы будет изменяться больше в отличие от расстояния по горизонтали двух крайних точек лица. То есть нужно

проводить нормализацию признаков. Нужно соотносить признаки к размерам между точками на линиях, которые параллельны тем, которые будут проходить через некоторые точки определяемого в будущем признака. Наши точки лица должны относиться к пространственной системе координат (рис. 4).

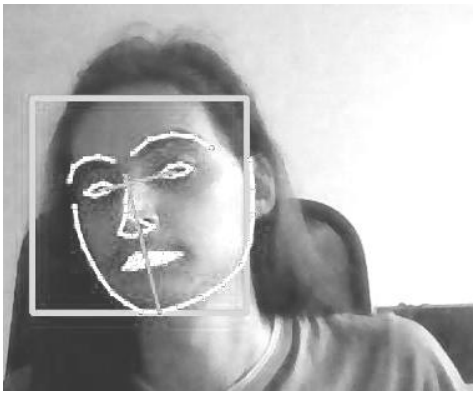


Рис. 3. Расположенные на лице человека антропометрические точки

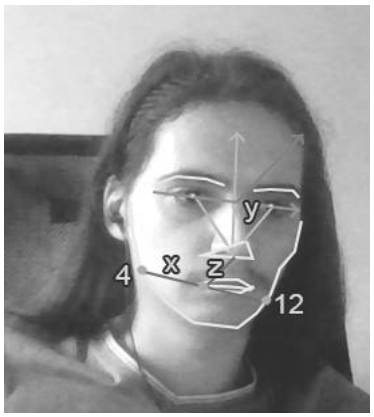


Рис. 4. Модель системы координат с нанесенной парой лицевых точек

Ось *x* проходит через точки, относящиеся к центрам глаз. Ось *y* проходит через точки, относящиеся к средней точке между центрами глаз и точки 33. Ось *z* проходит через точки, относящиеся к средней точке между центрами глаз и точки *27*.

Такие точки, которые определяют систему координат, были выбраны по причине условий взаимной перпендикулярности осей координат и симметрии пар лицевых точек, если составлять определенное отношение к плоскости *y* и *z*.

По данной системе координат можно определить координаты любых других точек. Например, возьмем точки *4* и *12*. Соединяем их и вычисляем середину отрезка. Из полученной середины проводим отрезок до *y* так, чтобы он был параллелен *z*. Отрезки ломанной от точки *4* до начала координат будут отображать координаты (*x*, *y*, *z*) для нашей точки *4*. Перейдем к подробному алгоритму определения 3D-координат.

В нашей системе координат начало будет определено посередине, между глазами (рис. 5). Направление осей сделаем более привычным (ось *y* вверх).

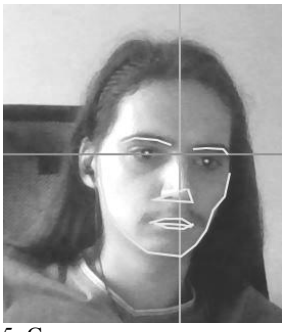


Рис. 5. Система координат с началом в середине отрезка между глаз

Система координат будет определяться на основе лицевых точек *42*, *45*, *36*, *39* (12):

$$\begin{aligned} X_{\text{left}} &= (X[\text{Point36}] + X[\text{Point39}])/2, \\ Y_{\text{left}} &= (Y[\text{Point36}] + Y[\text{Point39}])/2, \\ X_{\text{right}} &= (X[\text{Point42}] + X[\text{Point45}])/2, \\ Y_{\text{right}} &= (Y[\text{Point42}] + Y[\text{Point45}])/2, \\ X_{\text{Zero}} &= (X_{\text{left}} + X_{\text{right}})/2, \\ Y_{\text{Zero}} &= (Y_{\text{left}} + Y_{\text{right}})/2. \end{aligned} \tag{12}$$

Далее можно вычислить определенные направления осей координат СК, которые привязаны к лицевым точкам. Воспользуемся угловыми коэффициентами прямой в плоскости (рис. 6).

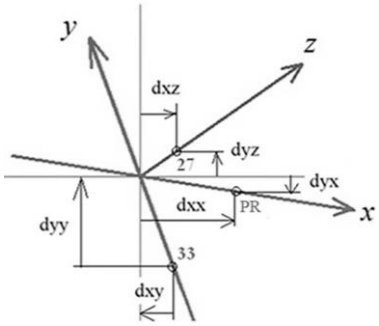


Рис. 6. Система координат, соотнесенная с лицевыми точками

Номера лицевых точек *33* и *27* используются для определения осей. Определим выражения (13), которые в дальнейшем будем использовать для нахождения точек пересечения проекций осей:

$$\begin{aligned} K_x &= dyx/dxx, \\ K_y &= dyy/dxy, \\ K_z &= dyz/dxz. \end{aligned} \tag{13}$$

Смещения типа *d*** (14) определены координатами точек *PR*, *33* и *27*:

$$\begin{aligned} dxx &= X_{\text{right}} - X_{\text{zero}}, \\ dyx &= -Y_{\text{right}} - Y_{\text{zero}}, \\ dxy &= -X[\text{Point33}] - X_{\text{zero}}, \\ dyy &= Y[\text{Point33}] - Y_{\text{zero}}, \\ dxz &= X[\text{Point27}] - X_{\text{zero}}, \\ dyz &= -Y[\text{Point27}] - Y_{\text{zero}}. \end{aligned} \tag{14}$$

Необходимые нам смещения определяются в системе координат (15) с центром в точке ($X_{\text{zero}}, Y_{\text{zero}}$) и осью *Y*, которая направляется вверх:

$$\begin{aligned} X &= X_{\text{оск}} - X_{\text{zero}}, \\ Y &= -Y_{\text{оск}} - Y_{\text{zero}}. \end{aligned} \tag{15}$$

$X_{\text{оск}}, Y_{\text{оск}}$ – это текущие координаты лицевых точек в оконной системе координат.

Затем определим проекции отрезков координатной ломаной для каждой пары симметричных лицевых точек. Возьмем наши прежние точки *4* и *12*. Определим необходимое нам расстояние между точками, используя теорему Пифагора (рис. 7) (16):

$$\begin{aligned} DX12 &= |X1 - X2|, \\ DY12 &= |Y1 - Y2|, \\ D &= \sqrt{(DX12 \times DX12 + DY12 \times DY12)}. \end{aligned} \tag{16}$$

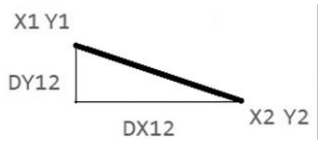


Рис. 7. Рассматриваемый треугольник

Далее, следует перейти к рассмотрению проекций координат *x* точек *12* и *4* (17):

$$\begin{aligned} x12p &= D/2, \\ x4p &= -x12p. \end{aligned} \tag{17}$$

Перейдем к определению проекций координаты *z* точек. Найдем середину отрезка между точками (18):

$$\begin{aligned} X_{\text{mid}} &= (X[\text{Point4}] + (X[\text{Point12}] - X[\text{Point4}]))/2, \\ Y_{\text{mid}} &= (Y[\text{Point4}] + (Y[\text{Point12}] - Y[\text{Point4}]))/2. \end{aligned} \tag{18}$$

Осуществляем переход к системе координат с центром в начальных точках и направленной вверх осью *y* (19):

$$\begin{aligned} X_{\text{mid}} &= X_{\text{mid}} - X_{\text{zero}}, \\ Y_{\text{mid}} &= -(Y_{\text{mid}} - Y_{\text{zero}}). \end{aligned} \tag{19}$$

Через данные точки необходимо провести прямую, параллельную проекции оси *z* до пересечения с проекцией оси *y*. Найдем координаты точки пересечения этих прямых (20):

$$\begin{aligned} X_{\text{int}} &= (Y_{\text{mid}} - K_z \times X_{\text{mid}})/(K_y - K_z), \\ Y_{\text{int}} &= K_y \times X_{\text{int}}. \end{aligned} \tag{20}$$

Третья точка ломаной будет находиться в начале координат. По трем точкам определим необходимые проекции координат *z* и *y* для точек *12* и *4* по теореме Пифагора (21):

$$\begin{aligned} y3p &= \sqrt{(X_{\text{int}} \times X_{\text{int}} + Y_{\text{int}} \times Y_{\text{int}})}, \\ z3p &= \sqrt{(X_{\text{int}} - X_{\text{mid}})^2 + (Y_{\text{int}} - Y_{\text{mid}})^2}. \end{aligned} \tag{21}$$

От данных проекций переходим к нормализованным признакам-координатам (22):

$$\begin{aligned} x12 &= x12p/LX, \\ x4 &= -x12, \\ y4 &= y4p/LY, \\ z4 &= z4p/LZ. \end{aligned} \tag{22}$$

LX, *LY* и *LZ* – это отрезки на осях, которые отсекаются точками *PR*, *33* и *27*. Вычислим их по теореме Пифагора (23):

$$\begin{aligned} LX &= \sqrt{dyx^2 + dxx^2}, \\ LY &= \sqrt{dyy^2 + dxy^2}, \\ LZ &= \sqrt{dxz^2 + dyz^2}. \end{aligned} \tag{23}$$

Далее, опираясь на текущие формулы, можно построить трехмерную модель (рис. 8) и высчитать необходимые координаты интересующей пары точек. Реализация данной методики, алгоритмов и применения вышеописанных функций была выполнена на языке C++ в связке с библиотекой OpenCV. В ней содержится более полный и идеальный каскадный классификатор. На его основе проводились вычисления координаты точек по вышеописанным формулам.

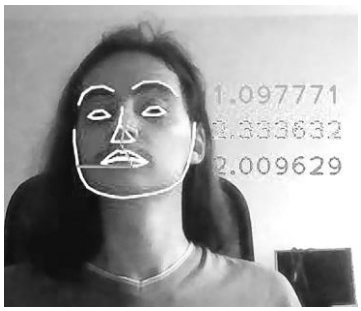


Рис. 8. Вычисление координаты точки *4*

Литература

1. Эмпирические методы распознавания лица [Электронный ресурс]. – Режим доступа: http://network-journal.mpei.ac.ru/cgi-bin/main.pl?l=ru&n=17&pa=11&ar=2, свободный (дата обращения: 17.09.2022).

2. Алгоритм вейвлет-преобразования Хаара в конечном поле / И.А. Калмыков, А.А. Ложечкин, А.В. Гапочкин, М.И. Калмыков // Современные наукоемкие технологии. – 2014. – № 11(4). – С. 18–23.

3. Robust real-time face detection / P. Viola, M.J. Jones // International Journal of Computer Vision. – 2014. – No. 2(57). – PP. 137–154.

4. Применение цифровой лицевой антропометрии / Г.А. Кухарев, Н. Казиева // Научно-технический вестник информационных технологий, механики и оптики. – 2019. – Т. 19, вып. 2, № 11(4). – С. 255–270.

Соколов Александр Васильевич
Студент каф. АСУ ТУСУРа
Эл. почта: burtonsocolov@gmail.com

УДК 004.942

В.В. Болдырев, М.А. Мельниченко, Н.А. Мурзин

Идентификация роботизированного технологического процесса механической обработки с применением графической нотации IDEF0

С помощью графической нотации IDEF0 произведена идентификация роботизированного технологического процесса механической обработки для рассмотрения возможности проведения оптимизационных мероприятий по повышению производительности и сокращения издержек на проведение технологического процесса. Для рассматриваемого роботизированного комплекса приведена структурная схема, на основе которой сформированы процессные модели нулевого и первого уровней, а также предложена процессная модель модернизированного процесса с учетом разделения последовательностей технологических операций между сотрудниками.
Ключевые слова: идентификация, технологический процесс, механическая обработка, промышленный робот, IDEF0, процессная модель.

На сегодняшний день задача повышения эффективности функционирования роботизированных технологических комплексов (РТК) имеет высокую актуальность, поскольку от интегральных показателей отдельных роботизированных комплексов зависят экономическая стабильность и эффективность работы промышленного предприятия [1].

Основными критериями повышения эффективности работы РТК являются максимизация производительности (быстродействия) и минимизация издержек на производство (энергетическая эффективность и рациональное использование ресурсов) [2, 3], которые могут быть достигнуты за счет применения интеллектуальных алгоритмов управления [4], оптимизации расположения траекторий в рабочей зоне [5], применения генетических алгоритмов [6, 7].

Для определения целесообразности применения перечисленных инструментов к выбранному роботизированному технологическому процессу (РТП) необходимо провести его идентификацию. При детальной и качественно выполненной идентификации видны уязвимые точки процесса, требующие проведения оптимизационных мероприятий в условиях существующих ограничений, которые также могут быть выявлены с помощью широкого инструментария методов идентификации.

Целью работы является идентификация роботизированного технологического процесса механической обработки с целью рассмотрения возможностей проведения оптимизационных мероприятий технологического и организационного плана для повышения эффективности функционирования рассматриваемого процесса.

Методология

В современном мире существует огромное количество методов идентификации технологических процессов (ТП). Среди наиболее простых и универсальных инструментов идентификации ТП можно выделить: Business Process Model and Notation (BPMN), Unified Modeling Language (UML), ICAM Definition (IDEF) [8, 9]. Каждый из подходов описывает определенные процессы или объекты.

Среди перечисленных методов функционального моделирования для идентификации роботизированного технологического процесса механообработки наиболее целесообразным представляется использование нотации IDEF0 [10], поскольку данная нотация, основана на рекомендациях стандарта ISO9001 [11] по управлению качеством технологических процессов с учетом специфики процессов производства.

IDEF0 – методология функционального моделирования и графическая нотация, предназначенная для формализации и описания большинства технических процессов. В основе методологии лежат четыре основных понятия: функциональный блок, интерфейсная дуга, декомпозиция, глоссарий.

Результаты

В качестве объекта исследования принят роботизированный технологический процесс механической обработки. Структурная схема роботизированного технологического комплекса (РТК) механической обработки представлена на рис. 1 [12, 13].

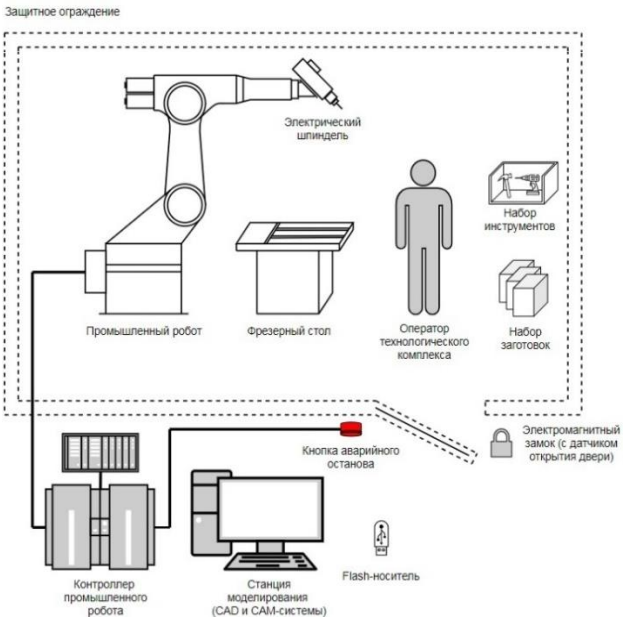


Рис. 1. Структурная схема РТК механической обработки

Роботизированная ячейка механической обработки представляет собой комплекс из промышленного робота, дополнительной внешней оси (позиционера), управляющего контроллера, оснастки, фрезерного стола и систем безопасности. Данный комплекс осуществляет фрезерование и гравировку объёмных / плоских изделий. В качестве материалов используется дерево, поликарбонат, литой акрил, модельный пластик, алюминий и другие материалы. Управление роботизированной ячейкой осуществляет оператор технологического комплекса [14].

Согласно методологии IDEF0, на основе представленной структурной схемы РТК механической обработки сформирован нулевой (контекстный) уровень процессной модели исследуемого технологического процесса (рис. 2).

Рис. 2. Контекстный уровень процессной модели роботизированной механической обработки

Первый уровень процессной модели исследуемого технологического процесса представлен на рис. 3.

В рамках рассматриваемого ТП все отдельные операции процесса выполняет оператор технологического комплекса, при этом, стоит отметить, что создание 3D-модели и набор подготовительных операций (подготовка заготовки, закрепление заготовки и калибровка базы) не зависят друг от друга. В связи с этим, производительность процесса может быть увеличена за счет разделения обязанностей между оператором технологического комплекса и дополнительным специалистом по подготовке ТП (инженером-технологом), как показано на рис. 4.

Результатом проведения данных организационных мероприятий является повышение производительности роботизированного комплекса при незначительных затратах на дополнительную оплату труда инженера-технолога в условиях массового производства деталей.

Заключение

Таким образом, в данной работе произведена идентификация роботизированного технологического процесса механической обработки. Нотация IDEF0 позволила создать модель и провести детальную декомпозицию исследуемого РТП. В результате были выявлены основные технологические операции процесса, а также ключевые показатели, влияющие на производительность роботизированного комплекса, такие как время на подготовку, время на разработку управляющей программы и время на проведение ТП.

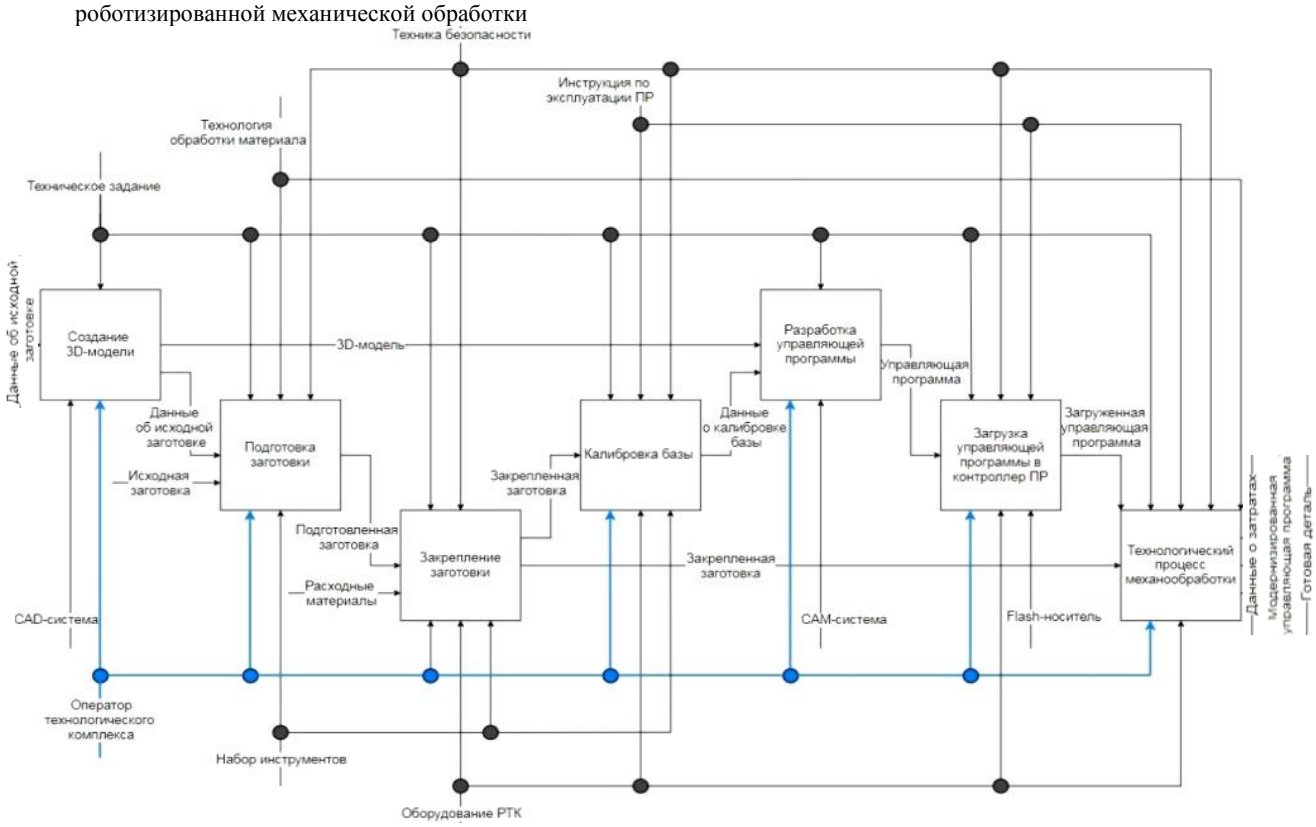


Рис. 3. Первый уровень процессной модели роботизированной механической обработки

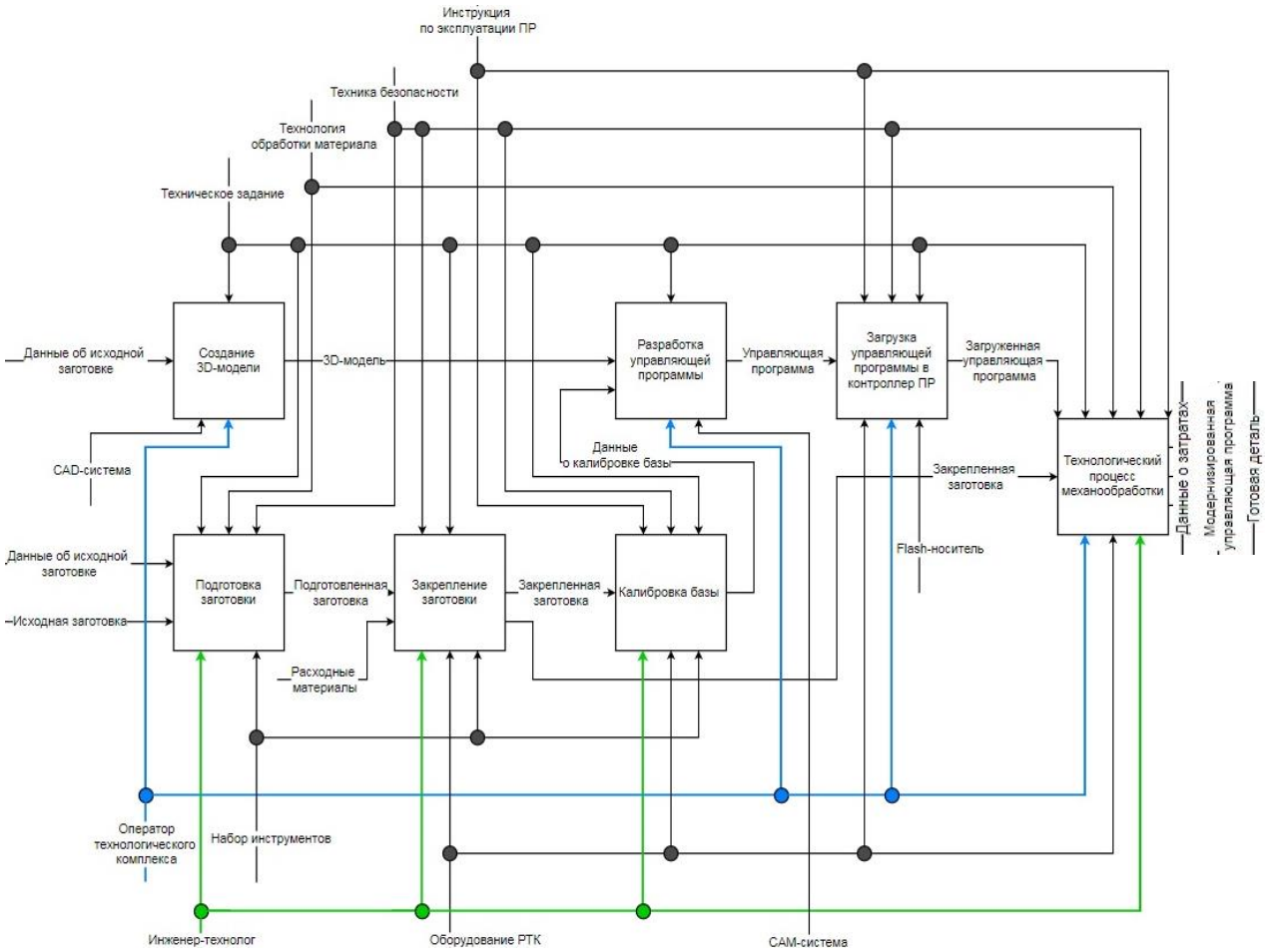


Рис. 4. Первый уровень процессной модели роботизированной механической обработки с участием инженера-технолога

Также при идентификации технологического процесса были выявлены возможности повышения производительности комплекса за счет разделения обязанностей между оператором технологического комплекса и дополнительным специалистом по подготовке отдельных операций.

Заключение

Таким образом, в данной работе произведена идентификация роботизированного технологического процесса механической обработки. Нотация IDEF0 позволила создать модель и провести детальную декомпозицию исследуемого РТП. В результате были выявлены основные технологические операции процесса, а также ключевые показатели, влияющие на производительность роботизированного комплекса, такие как время на подготовку, время на разработку управляющей программы и время на проведение ТП. Также при идентификации технологического процесса были выявлены возможности повышения производительности комплекса за счет разделения обязанностей между оператором технологического комплекса и дополнительным специалистом по подготовке отдельных операций.

Литература

1. Прогнозирование и измерение значений отдельных параметров роботизированного технологического процесса / А.Ю. Ефимов, М.А. Горькавый, В.П. Егорова // Матер. конф.: Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований. – Комсомольск-на-Амуре: КНАГУ, 2020. – С. 229–233.

2. Ворощенко В.Д., Горькавый М.А. Оптимизация отдельных параметров роботизированного процесса // Молодежь и наука: актуальные проблемы фундаментальных и прикладных исследований. – Комсомольск-на-Амуре: КНАГУ, 2021. – С. 203–206.

3. Егорова В.П., Горькавый М.А. Идентификация и моделирование роботизированных производственных процессов // Производственные технологии будущего: от создания к внедрению: матер. V Междунар. науч.-практ. конф. – Комсомольск-на-Амуре: КНАГУ, 2022. – С. 196–199.

4. Интеллектуальное управление активно-адаптивными сетями с применением верхнеуровневых алгоритмов / А.Г. Васьков, Ю.В. Кудрявцева, Ф.В. Молотов, Ю.П. Симаков, Т.А. Шестопалова, В.А. Юриков // Вестник кыргызско-российского славянского университета. – 2020. – С. 63–70.

5. Ермолов И.Л. Повышение эффективности манипуляторов роботов путём оптимизации расположения траекторий в рабочей зоне // Десятая Всерос. мультиконф. по проблемам управления (МКПУ–2017). – Таганрог: ЮФУ, 2017. – С. 32–34.

6. Применение генетических алгоритмов в интеллектуальных системах управления технологическим процессом / Е.В. Сиротина, Е.В. Шварев, К.А. Крышко // Инновации. Наука. Образование. – 2021. – С. 40–43.

7. Оптимизация структур роботизированных комплексов резки и упаковки листового стекла / В.А. Ива-

щенко, О.О. Мугин, Д.Ю. Петров, С.А. Черкасов // Матер. XIV Междунар. конф. по электромеханике и робототехнике «Завалишинские чтения – 2019». – Курск, 2019. – С. 247–252.

8. Проектирование информационных систем с CA ERwin Modeling Suite 7.3: учеб. пособие / В.И. Горбаченко, Г.Ф. Убиенных, Г.В. Бобрышева. – Пенза: Изд-во ПГУ, 2012. – 154 с.

9. Процессный подход к управлению. Моделирование бизнес-процессов / В. Репин, В. Елиферов. – М.: Манн, Иванов и Фербер, 2013. – 544 с.

10. Исследование особенностей применения метода функциональной декомпозиции при разработке имитационных моделей цифрового двойника роботизированного процесса / М.А. Горькавый, В.П. Егорова, А.И. Горькавый, М.А. Мельниченко // Ученые записки Комсомольского-на-Амуре государствеСс. 27– 34.

11. Национальный стандарт Российской Федерации ГОСТ Р ИСО 9001–2015. Системы менеджмента качества.

12. Мельниченко М.А. Особенности формирования автоматизированного технологического процесса и управляющих программ для робототехнического комплекса механической обработки // Производственные технологии будущего: от создания к внедрению. – Комсомольск-на-Амуре: КНАГУ, 2021. – С. 227–231.

13. Внедрение интеллектуальных роботизированных систем в производственные процессы судостроительного предприятия / М.А. Горькавый, А.И. Горькавый, А.С. Гудим, М.А. Мельниченко, В.П. Егорова // Морские интеллектуальные технологии. – 2021. – № 2-2 (52). – С. 97– 104.

14. Optimization of technological parameters of robotized mechanical processing processes of aviation products / A.Y. Efimov, M.A. Gorkavyy, V.P. Egorova, A.I. Gorkavyy // Lecture Notes in Networks and Systems. – 2021. – Vol. 200. – PP. 204–215.

Болдырев Владислав Вячеславович
Ст. преп. каф. управления инновационными процессами и проектами Комсомольского-на-Амуре государственного университета
Эл. почта: Boldurev16@gmail.com

Мельниченко Маркел Андреевич
Аспирант, преп. каф. управления инновационными процессами и проектами Комсомольского-на-Амуре государственного университета
Эл. почта: markel.96dk@mail.com

Мурзин Никита Алексеевич
Студент каф. управления инновационными процессами и проектами Комсомольского-на-Амуре государственного университета
Эл. почта: resaujt@mail.ru

УДК 621.396.41

Д.А. Русаков

Методика и программа определения общего содержания газовой составляющей атмосферы

На сегодняшний день важно отслеживание процессов, происходящих с нашей планетой, в особенности тех вещей, которые непосредственно влияют не только на качество нашей жизни, но и на возможность жизни на планете Земля. Одним из таких процессов является газовая составляющая нашей планеты, а именно различное содержание газовых компонент в атмосфере Земли. Изучение пространственно-временной изменчивости газов в атмосфере представляет собой актуальную задачу в связи с влиянием газовых компонент на климат. Атмосферные газы оказывают существенное влияние на климат Земли различными способами. Таким образом, важнейшая задача исследований состояния газового содержания атмосферы и трендов атмосферных параметров – постоянный мониторинг изменений в трендах газов естественного и антропогенного происхождения, а также содержания и распределения газовых составляющих в атмосферах.

Ключевые слова: система, газовая составляющая, методика, мониторинг, TROPOMI, Sentinel-5P.

Для измерения и отслеживания содержания данных газовых составляющих в атмосфере [1] используются различные средства, в данной работе в качестве средства сбора информации быил выбраны спутник Sentinel-5P [2] и прибор TROPOMI [3], установленный на нем. Прибор TROPOMI позволяет составить карту множества следовых газов, таких как диоксид азота, озон, формальдегид, диоксид серы, метан, угарный газ и аэрозоли – все они влияют на воздух, которым мы дышим, а значит, на наше здоровье и климат [4].

Характеристики и особенности приборов сбора информации

Sentinel-5P – это спутник на низкой околоземной орбите, оснащенный небольшими пусковыми установками, включая VEGA и ROCKET, причем ROCKET является выбранной ракетой-носителем. Sentinel-5P рассчитан на семилетний срок службы с 80 кг гидразинового топлива.

Механическая платформа состоит из шестиугольной конструкции, поддерживающей электрические блоки платформы и блок управления приборами TROPOMI (ICU), и подключается к стандартному интерфейсному кольцу ракеты-носителя.

Основными подсистемами спутника являются:

– двигательная подсистема (включая бак с гидразином);

– подсистема электроснабжения (EPS), состоящая из трех развертываемых солнечных батарей, обеспечивающих питание космического аппарата и полезной нагрузки;

– подсистема терморегулирования (TCS), включая нагреватели, необходимые для поддержания тепловой среды на платформе;

– система управления ориентацией и орбитой (AOCS), состоящая из трех головок звездного трекера, GPS-приемников, реактивных колес, магнитоторкеров, датчика грубого заземления и магнитометров;

– подсистема обработки данных (DHS), содержащая бортовой компьютер с памятью объемом на 72 ч для хранения данных спутникового обслуживания и команд в течение не менее 7 дней работы;

– подсистема связи S-диапазона (TT&C), обеспечивающая два резервных транспондера S-диапазона для приема телекоманд и передачи данных спутникового обслуживания;

– подсистема обработки и передачи данных полезной нагрузки (PDHT), обеспечивающая объем массовой памяти для хранения «научной» телеметрии и двух резервных передатчиков X-диапазона.

В табл. 1 приведены основные характеристики спутниковой платформы.

Таблица 1

Основные характеристики спутниковой платформы

Характеристика	Описание
Шина	AstroBus L 250 M от Airbus Defence & Space
Контроль ориентации	Стабилизированный по трем осям с опциональным рулевым управлением по рысканию
Масса при запуске	820 кг
Мощность	1500 Вт, средняя потребляемая мощность 430 Вт
Емкость аккумулятора	156 А·ч
Емкость хранения данных	(Резервная) массовая память объемом 480 Гбит для хранения телеметрии. Емкость хранилища должна обеспечивать нисходящую передачу данных наблюдений TROPOMI без потерь на основе одного прохода станции X-диапазона на орбиту

Sentinel-5P использует орбиту с высоким наклоном (приблизительно 98,7 °C). Наклон орбиты – это угловое расстояние плоскости орбиты от экватора.

Орбита Sentinel-5P представляет собой околополярную солнечно-синхронную орбиту с пересечением экватора восходящего узла в 13:30 по среднему местному солнечному времени. На солнечно-синхронной орбите поверхность всегда освещается под одним и тем же углом к солнцу.

Орбитальный цикл составляет 16 дней (14 витков в день, 227 витков за цикл). Цикл орбиты – это время, необходимое спутнику для прохождения над одной и той же географической точкой на Земле.

Исходная высота орбиты составляет приблизительно 824 км.

Важной особенностью миссии Sentinel-5P является синергетическое использование одновременных измерений данных тепловизора с набора видимых / инфракрасных тепловизоров и радиометров (VIIRS), запущенных на спутнике Suomi-NPP (национальное партнерство по полярной орбите) NASA/NOAA (миссия NPP началась 28 октября 2011 г.). VIIRS сможет предоставлять данные о маске облака с более высоким пространственным разрешением, чем маска облака, вычисленная по данным Sentinel-5P. Орбита Sentinel-5P выбрана таким образом, чтобы она отставала от Suomi-NPP на 3,5 мин по восходящему узлу местного времени (LTAN), что позволяет полосе наблюдения SENTINEL-5P оставаться в пределах пространства, наблюдаемого Suomi-NPP. Зоны наблюдения продемонстрированы на рис. 1.

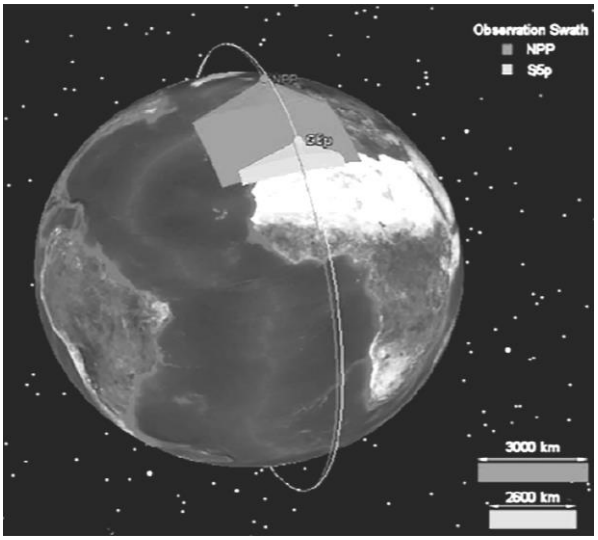


Рис. 1. Площадь, покрываемая Sentinel-5P и Suomi-NPP

Благодаря своей широкой полосе 108 °C (приблизительно 2 600 км на земле) прибор TROPOMI / Sentinel-5P обеспечивает полный ежедневный охват поверхности измерениями яркости и отражения для широт > 7 °C и < –7 °C, а также покрытие более 95% для широт в интервале [–7 °C, 7 °C].

Ранние дневные наблюдения TROPOMI / Sentinel-5P (LTAN: 13:30) будут иметь сильную синергию с утренними наблюдениями в рамках Global Ozone Monitoring Experiment 2 (GOME-2/Met-OP). Хотя GOME-2 имеет более низкое пространственное разрешение и не имеет спектрального диапазона SWIR, комбинация TROPOMI / Sentinel-5P и GOME-2 / Met-OP позволяет наблюдать суточные изменения, как успешно показано с помощью комбинации OMI / SCIAMACHY.

Спутник Sentinel-5P переносит один инструмент мониторинга TROPOMI (прибор для мониторинга тропосферы).

Инструмент для мониторинга тропосферы (TROPOMI) был совместно профинансирован ESA и Нидерландами. Ключевые организации из Нидерландов включают KNMI (Королевский метеорологический институт Нидерландов), SRON (Организация космических исследований Нидерландов), TNO (Нидерландская организация прикладных научных исследований) и Airbus DS-NL от имени NSO (Нидерландское космическое управление).

KNMI и SRON отвечают за разработку продуктов уровня 1B и некоторых продуктов уровня 2 (например, метан, диоксид азота). Airbus DS-NL был основным подрядчиком по изготовлению приборов, и многие европейские отрасли промышленности внесли свой вклад в создание подсистем. Бельгийский институт космической аэрономии (BIRA), Немецкий аэрокосмический центр (DLR) и Лаборатория Резерфорда Эпплтона (RAL) также вносят свой вклад в разработку продукта TROPOMI Level-2. Обработка оперативных данных осуществляется в DLR (от имени ESA и Европейской комиссии).

Прибор TROPOMI – это космический спектрометр с надирным обзором и формированием изображений, охватывающий диапазон длин волн от ультрафиолетового до коротковолнового инфракрасного диапазона.

Прибор, являющийся единственным инструментом, переносимым спутником Sentinel-5P, использует пассивные методы дистанционного зондирования для достижения своей цели путем измерения в верхней части атмосферы (TOA) солнечной радиации, отраженной от Земли и излучаемой ею.

Прибор работает в конфигурации push-broom (без сканирования) с шириной полосы ~2600 км на поверхности Земли. Типичный размер пикселя (вблизи надира) составит 7×3,5 км^2 для всех спектральных диапазонов, за исключением диапазона УФ1 (7×28 км^2), и коротковолновое инфракрасное излучение (7×7 км^2).

Основные характеристики TROPOMI

Тип: спектрометр с пассивной решеткой.

Конфигурация: push-broom (без сканирования) при просмотре в надире.

Ширина полосы обзора: 2 600 км.

Пространственная выборка: 7×7 км^2.

Спектральные каналы: 4 спектрометра, каждый с электронным разделением на две полосы (2 – в УФ, 2 – в видимом спектре, 2 – в близкой инфракрасной области, 2 – в коротковолновом инфракрасном излучении).

Радиометрическая точность (абсолютная): от 1,6% (коротковолновом инфракрасном излучении) до 1,9% (УФ) от измеренного спектрального отражения Земли.

Общая масса: 204,3 кг без учета ICU (16,7 кг), который интегрирован на платформе отдельно от прибора.

Размеры: 1,40×0,65×0,75 м.

Срок службы конструкции: 7 лет.

Средняя потребляемая мощность: 155 Вт.

Объем генерируемых данных: 139 Гбит на полную орбиту.

Существует три уровня обработки TROPOMI.

Уровень-0: упорядоченная по времени, необработанная спутниковая телеметрия без временного перекрытия, включая данные датчиков 4 спектрометров как для атмосферных, так и для калибровочных измерений. Также включены инженерные данные, вспомогательные данные спутника и метаданные о получении, необходимые для наземной обработки и задач мониторинга. Эти данные не предоставляются общественности.

Уровень-1B: географически привязанные и радиометрически скорректированные излучения верхней части атмосферы Земли во всех спектральных диапазонах, а также солнечное излучение.

Уровень-2: геолоцированные суммарные столбы озона, диоксида серы, диоксида азота, монооксида углерода, формальдегида и метана; геолоцированные тропосферные столбы озона; геолоцированные вертикальные профили озона; геолоцированная информация об облаках и аэрозолях.

Также в табл. 2 представлен спектральный диапазон спектрометров TROPOMI.

Таблица 2

Спектральный диапазон спектрометров

Спектрометр	Спектральный диапазон (нм)
Ультрафиолетовое излучение (UV)	270–300
	300–320
Ультрафиолетово-видимый (UVIS)	320–405
	405–500
Ближнеинфракрасная область (NIR)	675–725
	725–775
Коротковолновое инфракрасное излучение (SWIR)	2305–2345
	2345–2385

Метод обработки данных

Данные солнечного излучения можно отобразить с помощью полосы пропускания солнечного излучения.

Взаимодействие полосы пропускания солнечного излучения фильтра приемника приведено на рис. 2.

Для расчёта отклонения газовой составляющей сначала найдем интенсивность излучения (1) и (2):

$$I_{S_1} = C_1 \int\limits_{\Delta\lambda_1}^{\infty} I_0(\lambda) \times I_{f_1}(\lambda) \times$$

$$\times \exp\left\{ -\int\limits_0^H (\alpha_g(\lambda,h) + \alpha_a(\lambda,h) + \alpha_{ms}(\lambda,h))dh \right\}, \qquad (1)$$

$$I_{S_2} = C_2 \int\limits_{\Delta\lambda_2}^{\infty} I_0(\lambda) \times I_{f_2}(\lambda) \times$$

$$\times \exp\left\{ -\int\limits_0^H (\alpha_g(\lambda,h) + \alpha_a(\lambda,h) + \alpha_{ms}(\lambda,h))dh \right\}, \qquad (2)$$

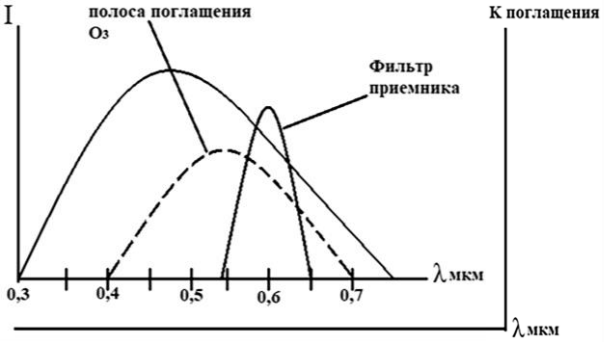


Рис. 2. Взаимодействие полосы пропускания солнечного излучения фильтра приемника

Теперь рассчитаем отклонения газовой составляющей (3).

$$W_{O_3} \sim \frac{I_{S_2}}{I_{S_1}} \times \frac{C_2}{C_1}, \qquad (3)$$

λ – длина волны, α – коэффициент поглощения, I – интенсивность излучения, C – фильтр приемника.

Опираясь на формулы вычисления, можно вычислить коэффициенты газовых составляющих в атмосфере Земли по данным, полученным спутником. Реализация методики выполнена на языке Python.

Литература

1. Hay W.W. Experimenting on a Small Planet: A History of Scientific Discoveries, a Future of Climate Change and Global Warming // Springer. – 2016. – PP. 426–819.

2. Sentinel-5P [Электронный ресурс]. – Режим доступа: https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-5p, свободный (дата обращения: 17.09.2022).

3. TROPOMI [Электронный ресурс]. – Режим доступа: http://www.tropomi.eu/mission-status, свободный (дата обращения: 17.09.2022).

4. Загрязнение воздуха. Источник и контроль / К. Уорк, С. Уорнер. – 1980.

Русаков Денис Алексеевич
Магистрант каф. АСУ ТУСУРа
Эл. почта: deniskarul500@mail.ru

Секция 15

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

Председатель секции – ***Шелупанов Александр Александрович****,*
президент ТУСУРа, профессор, д.т.н.

УДК 004.056.53

Д.С. Лазорин, Д.И. Правиков

О подходах к обеспечению информационной безопасности цифровых двойников

Рассмотрены существующие подходы к обеспечению информационной безопасности цифровых двойников и формулированию предложений по выбору средств противодействия различным угрозам, прежде всего угрозе искажения цифрового двойника. Показано, что ни один из существующих подходов и методов не обеспечивает полноценной безопасности цифровых двойников. Предложена реализация комплексного подхода для обеспечения безопасности цифровых двойников в рамках актуальных для них специфических моделей угроз и нарушителя.

Ключевые слова: цифровой двойник, цифровизация, информационная безопасность, модель угроз и нарушителя, компьютерная модель, многоуровневая система, цифровой сертификат, шифрование данных, криптографические алгоритмы, машинное обучение, токенизация, криптомодули.

Сегодня понятие «цифровой двойник» прочно заняло позиции в современных технологиях. Данный термин многогранен. Так или иначе – это виртуальная или компьютерная модель, способная выполнять и воспроизводить действия исходного объекта.

Термин «цифровой» основан на математический формулах, описывающих рассматриваемый объект и его действия либо же процессы. Отсюда вытекает термин «цифровизация». Это внедрение современных технологий в определенные области компании, которые позволяют автоматизировать её деятельность. Именно цифровизация позволяет вывести промышленность на абсолютно новый уровень [1].

Цифровой двойник связывает виртуальный (программный) мир с физическим. Это возможно с помощью информации, которая должна передаваться с различного рода оборудования. На помощь приходит огромное количество современных датчиков. Они могут быть установлены на объекте, конструкции или же на нефтеперерабатывающем заводе.

Однако количество поступающей информации велико, требуются огромные вычислительные мощности для их обработки. На помощь приходит цифровой двойник, который позволяет имитировать и предсказывать параметры того или иного объекта (конструкции, установки). Его особенность – обработка и обновление информации, получаемой с объекта, в реальном времени [2].

Документация Министерства энергетики Российской Федерации подчеркивает востребованность цифровых двойников на рынке [3].

При помощи технологий цифрового двойника появляется возможность контролировать, следовательно, и устранять негативные последствия еще до того, как они могут произойти. Новые решения на производстве должны обладать определённой мерой защиты, которая должна быть детализирована и продумана до мелочей.

Приоритетной задачей ставится необходимость обеспечения информационной безопасности цифровых двойников, применяемых на технологических процессах.

Элементы системы «Физический объект – цифровой двойник» (ФОЦД)

Обозначим главные части системной целостности, которые включают виртуальный и физический объект:

1. Физический объект (ФО).

2. Виртуальный объект или специальная модель, которая должна соответствовать физическому объекту по определённым параметрам (цифровой двойник, ЦД).

3. Взаимосвязи в системе, рассматриваемые в рамках виртуального и физического объекта.

Специфические свойства цифровых двойников

Необходимо рассмотреть специфические свойства и общие требования, которые должны относиться к ФОЦД.

Наиболее важным свойством является целостность, которая подразумевает под собой неразрывную взаимосвязь вышеперечисленных компонентов. Ведь именно от неё зависит полноценная работа целого комплекса процессов. Другое свойство – актуальность параметров, которые передаются цифровому двойнику.

Технология цифрового двойника может не соответствовать тому устройству, комплексу или физическому объекту, который управляется с его помощью.

Отметим важный момент: отличие стандартной математической модели и цифрового двойника, которое состоит в том, что модель цифрового двойника основывается не на идеализации, а на действительном поведении самого объекта.

Подходы и методы обеспечения информационной безопасности цифровых двойников

В данной работе были проанализированы научные источники и выбраны основные современные подходы и методы обеспечения информационной безопасности, которые могут быть применены к технологии цифровых двойников: блокчейн-структура, подход к организации информационного взаимодействия в многоуровневой системе, система

иерархической токенизации элементов системы, Digital ID (цифровой сертификат доверия), технология распределённых реестров, средства шифрования данных при передаче по каналам связи, комплексный подход с использованием машинного обучения и искусственного интеллекта, обеспечение безопасности ЦД в рамках концепции многоуровневой платформы [4–7].

Результате приведены в таблице.

Одни из способов и методов обеспечения комплексной безопасности цифровых двойников

Блокчейн-структура	✓
Организация взаимодействий в многоуровневой системе	✓
Цифровой сертификат доверия (Digital ID)	✓
Технология распределённых реестров	✓
Шифрование данных при передаче по каналам связи	✓
Использование машинного обучения и искусственного интеллекта	✓
Информационное взаимодействие в многоуровневой системе	✓
Соответствие модели реальному объекту	X

Отечественные индустриальные криптомодули ViPNet SIES Core & Pack

На данный момент большинство нефтегазовых месторождений находятся на завершающей стадии эксплуатации и не оснащены порой элементарными средствами автоматики. Следовательно, некоторые АИУС нефтегазовые общества, а также АСУ ТП месторождения используют отечественное оборудование фирмы Infotecs. Существуют решения по анализу информационной безопасности АСУ ТП от компании Infotecs.

Нами было протестировано следующее программное обеспечение от данной фирмы: ViPNet Client 4. Это программное обеспечение для защиты трафика на рабочих местах пользователей. Оно фильтрует весь входящий и исходящий трафик компьютера и позволяет обмениваться данными с другими узлами ViPNet по защищенному VPN-каналу (сокр. Virtual Private Network).

В программе имеется доступ к сетевым фильтрам, которые предназначены для использования или блокировки трафика по различным признакам. По умолчанию они ограничивают входящий незашифрованный трафик за исключением некоторых протоколов (DHCP, NetBIOS, WINS). Сетевые фильтры являются настраиваемыми. Программа контролирует сетевую активность приложений, которые установлены на устройстве. При попытке какого-либо приложения получить доступ к сети появляется соответствующее предупреждение. Помимо этого, появляется возможность отправлять сообщения электронной почты и вложения пользователям других узлов ViPNet. Стандартно сообщения зашифрованы и подписаны электронной подписью.

Стоит обратить внимание на криптографический модуль ViPNet SIES Core. Это средства защиты данных интеллектуальных устройств автоматики, входящие в состав комплекса продуктов ViPNet SIES, предназначенные для защиты информации, обрабатываемой устройствами автоматизации индустриальных систем. Они обеспечивает защиту данных в устройствах автоматизации, например: программируемые логические контроллеры (PLC), устройства сбора и передачи информации (RTU), промышленные контроллеры автоматизации (PAC), сенсоры, датчики, счетчики, различные исполнительные устройства.

Защищаемое устройство со встроенным ViPNet SIES Core может реализовывать различные сценарии защиты данных в зависимости от модели угроз и нарушителя информационной безопасности, разработанной для защищаемого устройства или индустриальной системы. Например, с помощью ViPNet SIES Core можно реализовать следующие сценарии: обеспечение целостности при передаче данных по существующим каналам связи; обеспечение конфиденциальности при передаче данных по существующим каналам связи; защита от навязывания индустриальной системе ложных данных; защита от повторов и навязывания индустриальной системе устаревших данных; доверенное обновление программного обеспечения (ПО) и конфигурации защищаемого устройства; доверенное хранение данных о функционировании защищаемого устройства [8].

На данный момент нами проводится тестирование криптографического модуля ViPNet SIES Core.

Заключение

Проведённый анализ выяснил, что ни один из рассматриваемых способов в отдельности не обеспечивает комплексной безопасности в рамках актуальных для цифровых двойников моделей угроз и нарушителя. Перспективным направлением является методология обеспечения информационной безопасности ЦД в соответствии с концепцией многоуровневой платформы.

Применение некоторых технологий защиты на сегодняшний день оправдано, но направлено на решение частных случаев. Для части методов (соответствие виртуальной модели реальному объекту) необходимо использовать другие подходы и средства, которые будут предложены в следующих работах.

Помимо вышеперечисленного, был разработан веб-сайт проекта (https://securedt.ru), на котором будет актуализироваться информация о научной работе, проведена демонстрация результатов тестирования вышеперечисленных способов и методов обеспечения информационной безопасности цифровых двойников.

Литература

1. Намиот Д.Е. Цифровые двойники и системы дискретно-событийного моделирования / Д.Е. Намиот, О.Н. Покусаев, В.П. Куприяновский, М.Г. Жабицкий // International Journal of Open Information Technologies ISSN. – 2021. – Vol. 9, No. 2. – PP. 2307–8162.

2. Цифровое зеркало «Газпром Нефть» [Электронный ресурс]. – Режим доступа: https://www.gazprom-neft.ru/press-center/sibneft-online/archive/2018-september-projects/1863687/ (дата обращения: 12.01.2022).

3. Паспорт: «Программы инновационного развития» ПАО «Газпром» до 2025 года. Министерство энергетики: инновационное развитие отраслей ТЭК [Электронный ресурс]. – Режим доступа: https://minenergo.gov.ru/node/4844 (дата обращения: 13.01.2022).

4. Дроговоз П.А., Кошкин М.В. Проекты внедрения технологий блокчейн и интернета вещей в трансграничных цепочках поставок // Управление научно-техническими проектами: сб. матер. III Междунар. науч.-техн. конф. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2019. – С. 153–156.

5. Building a digital twin for additive manufacturing through the exploitation of blockchain / C. Mandolla, A.M. Petruzzelli, G. Percoco, A. Urbinati // Comput. Ind. – 2019. – Vol. 109. – PP. 134–152.

6. Воробьев А.В. Концепция информационного пакетного взаимодействия в многоуровневой системе цифровых двойников // Изв. Сарат. ун-та. – Сер.: Математика. Механика. Информатика. – 2021. – Т. 21, вып. 4. – С. 532–543. DOI: 10.18500/1816-9791-2021-21-4-532-543

7. К вопросу о формулировании системного подхода к исследованиям в области цифровых платформ, распределенных реестров и цифровых активов / Д.И. Правиков, А.В. Глейм, В.И. Егоров, А.А. Рязанова, А.Ю. Щербаков // Вестник современных цифровых технологий. – 2021. – № 9. – С. 5–14.

8. Защита информации для АСУ ТП и M2M // Infotecs [Электронный ресурс]. – URL: https://infotecs.ru/resheniya/zashchita-informatsii-dlya-asu-tp-i-m2m.html (дата обращения: 11.09.2022).

Лазорин Данил Сергеевич
Студент Российского государственного университета нефти и газа (Национальный исследовательский университет) им. И.М. Губкина
(РГУ нефти и газа (НИУ) им. И.М. Губкина)
Эл. почта: lazorindanya@yandex.ru

Правиков Дмитрий Игоревич
Канд. техн. наук, зав. каф. комплексной безопасности критически важных объектов РГУ нефти и газа (НИУ) им. И.М. Губкина
Эл. почта: dip@gubkin.pro

УДК 004.032.26, 004.056

Д.Е. Каширский, Ф.А. Якименко

Применение нейронных сетей для исправления ошибок в криптографических ключах, полученных методами квантовой криптографии

Рассматривается применение древовидных машин четности, являющихся особым видом нейронных сетей, для исправления ошибок в криптографических ключах установки квантового распределения ключей. Проведен анализ криптостойкости процедуры для древовидных машин четности различной структуры и криптографических ключей разной длины.

Ключевые слова: квантовые коммуникации, криптографический ключ, исправление ошибок, древовидная машина четности.

Квантовые коммуникации – в настоящее время бурно развивающееся направление. Криптостойкость систем квантовых коммуникаций обеспечивается законами квантовой физики [1]. Получаемые в результате квантового распределения ключей (КРК) у двух легитимных пользователей, именуемых Алиса и Боб, криптографические ключи являются непригодными для применения из-за наличия в них ошибок. Поэтому эти криптографические ключи подвергаются процедуре классической постобработки, включающей в себя исправление ошибок и усиление секретности.

В [2] было предложено использовать нейронные сети, а именно древовидные машины четности (ДМЧ) для исправления ошибок в криптографических ключах, полученных методами квантовой криптографии. В данной работе рассматривается применение этого подхода для исправления ошибок в криптографических ключах установки КРК [3].

Древовидная машина четности

ДМЧ состоит из одного выходного нейрона, K скрытых нейронов, каждый из которых связан с N входными нейронами (рис. 1).

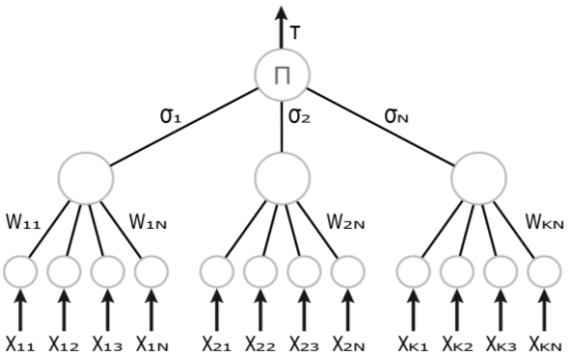


Рис 1. Структура древовидной машины четности

На вход нейронной сети подаются целые числа

$$x_{ij} \in \{-M, -M+1, \ldots, -1, 1, \ldots, M-1, M\}.$$

Веса между входными и скрытыми нейронами принимают значения

$$w_{ij} \in \{-L, -L+1, \ldots, L-1, L\}.$$

Здесь индекс $i = 1, \ldots, K$ означает номер скрытого нейрона ДМЧ, а индекс $j = 1, \ldots, N$ – номер входного нейрона, связанного с i-м скрытым нейроном. В данной работе максимальное значение входа M изменялось от 2 до 8, а максимальное значение веса L равнялось 8. K и N подбирались так, чтобы обеспечить требуемую длину криптографического ключа.

Выход i-го скрытого нейрона рассчитывается по формуле

$$\sigma_i = \operatorname{sgn}(h_i) = \begin{cases} +1, h_i \geq 0 \\ -1, h_i < 0 \end{cases}, \qquad h_i = \sum_{j=1}^{N} x_{ij} w_{ij}.$$

Выход ДМЧ определяется как произведение выходов скрытых нейронов

$$\tau = \prod_i \sigma_i .$$

Исправление ошибок в криптографических ключах

Процедура исправления ошибок в криптографических ключах основана на синхронизации двух ДМЧ посредством их обучения (рис. 2).

1. Алиса и Боб создают ДМЧ с заданными параметрами K, N, L.

2. Алиса и Боб преобразуют битовые последовательности своих криптографических ключей в целые числа, формируя веса ДМЧ.

3. Генерируется случайный входной вектор для ДМЧ.

4. Алиса и Боб подают входной вектор на входы своих ДМЧ и рассчитывают значения выходов ДМЧ.

5. Алиса и Боб обмениваются значениями выходов своих ДМЧ.

6. Если значения выходов равны, то Алиса и Боб производят обновление весов своих ДМЧ в соответствии с правилом обучения Хэбба:

$$w_{ij}^{*} = \mathrm{g}\left(w_{ij} + x_{ij}\tau\theta(\sigma_i, \tau)\right);$$

$$\theta(\sigma_i, \tau) = \begin{cases} 1, \sigma_i = \tau; \\ 0, \sigma_i \neq \tau; \end{cases} \qquad (1)$$

$$g(w) = \begin{cases} L-1, w > L-1; \\ -L, w < -L; \\ w, -L \leq w \leq L-1. \end{cases}$$

7. Если синхронизация весов ДМЧ Алисы и Боба не достигнута, то осуществляется переход к п. 3. В противном случае полученные веса преобразуются в биты криптографического ключа.

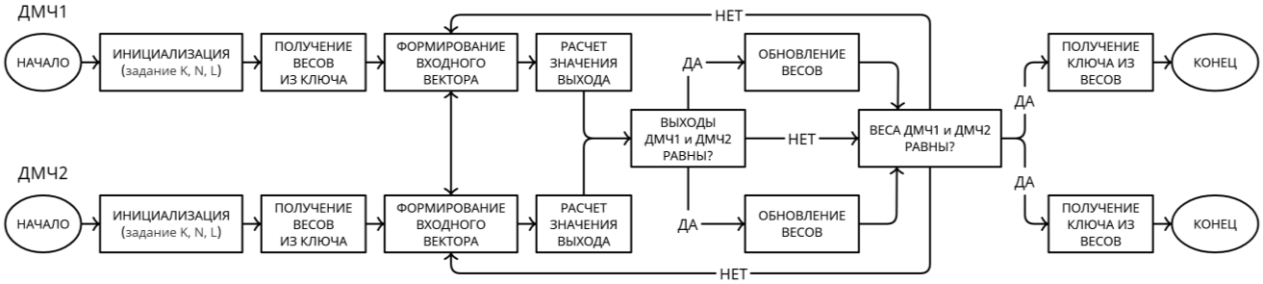


Рис. 2. Блок-схема процедуры исправления ошибок в криптографических ключах

Криптоаналитические атаки

Для проверки криптостойкости рассматривалось два варианта криптоаналитических атак [4]. Первая, простейшая атака «человек посередине», предполагает, что злоумышленник Ева создает такую же по структуре ДМЧ, как у Алисы и Боба, с заданными случайным образом весами. В качестве входного вектора Ева берет вектор, генерируемый на этапе 3 процедуры исправления ошибок, производит расчет выхода ДМЧ τ_E и сравнивает его с выходами ДМЧ Алисы τ_A и Боба τ_B. Если $\tau_A = \tau_B = \tau_E$, то Ева производит обновление весов своей ДМЧ согласно (1), в противном случае она не производит обновление весов своей ДМЧ.

Вторая, геометрическая атака, аналогична первой, только обновление весов ДМЧ Евы происходит и в случае, если $\tau_A = \tau_B \neq \tau_E$, но другим образом. Ева находит скрытый нейрон i, у которого $|h_i|$ имеет минимальное значение, и меняет значение выхода σ_i на противоположное по знаку. После это производится обновление весов ДМЧ Евы согласно (1).

Совершенная атака считается успешной, если Еве удалось синхронизировать свою ДМЧ с ДМЧ Алисы и Боба.

Криптографичекие ключи с ошибками

Используя установку КРК [3], были собраны длинные битовые последовательности Алисы и Боба, из которых были сформированы пары криптографических ключей разной длины для постобработки (табл. 1).

Таблица 1

Криптографические ключи с ошибками

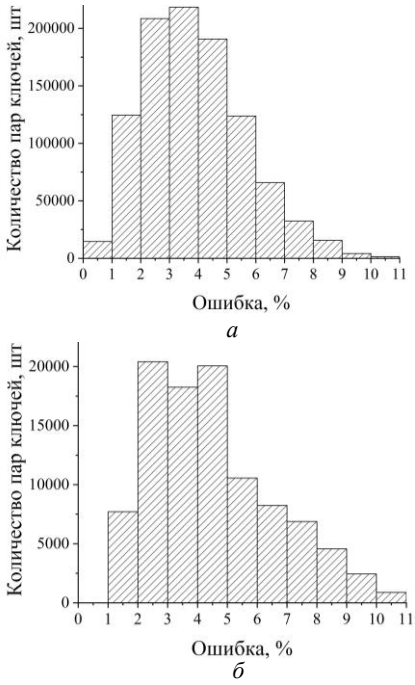
Длина, бит	Количество пар, шт.	Ошибка, %		
		мин.	средняя	макс.
512	1000000	0,00	3,89	10,94
1024	500000	0,29	3,89	10,94
2048	250000	0,68	3,89	10,99
4096	200000	0,81	4,51	10,99
8192	100000	1,06	4,51	11,00

На рис. 3 представлены распределения ошибок в криптографических ключах Алисы и Боба длиной 512 и 8192 бита.

Результаты

Проведенные эксперименты показали, что исправление ошибок в криптографических ключах с помощью ДМЧ стойко к простейшей атаке «человек посередине». Средняя по всем проведенным экспериментам с разной структурой ДМЧ максимальная синхронизация ДМЧ Евы с ДМЧ Алисы и Боба составила 77,38%.

В табл. 2 приведены результаты для случая геометрической атаки на процедуру исправления ошибок в ключах длиной 1024 бит. Для ключей с другой длиной наблюдается аналогичный результат. Проводилось по 1000000 экспериментов для каждой структуры ДМЧ. Видно, что количество успешных атак падает с возрастанием количества скрытых нейронов K в ДМЧ. Для M наблюдается обратная зависимость.



а

б

Рис. 3. Распределение ошибок в ключах Алисы и Боба длиной 512 бит (*а*) и 8192 бит (*б*)

Таблица 2

Результаты исправления ошибок в ключах длиной 1024 бит

K	N	M	Макс. синхронизация ДМЧ, %	Количество успешных атак, шт.
8	32	8	100,00	5120
8	32	6	100,00	1181
8	32	4	100,00	42
8	32	2	84,86	0
16	16	8	100,00	146
16	16	6	100,00	3
16	16	4	91,70	0
16	16	2	82,23	0
32	8	8	100,00	6
32	8	6	99,80	0
32	8	4	92,58	0
32	8	2	86,91	0

Отсутствие успешных атак еще не свидетельствует о их невозможности, поэтому нужно учитывать и величину синхронизации ДМЧ Евы с ДМЧ Алисы и Боба. Так, например, для случая с $K = 32$ и $M = 8$ не возникло ни одной успешной атаки, но величина максимальной синхронизации составила 99,80%, что свидетельствует о высокой вероятности успешности атаки на большем количестве экспериментов. Таким образом, для исправления ошибок в рассматриваемых вариантах криптографических ключей следует использовать ДМЧ с $K \geq 16$ и $M \leq 4$.

Заключение

Подход к исправлению ошибок в криптографических ключах, полученных методами квантовой криптографии, апробирован на реальной установке квантового распределения ключей. Показано влияние структуры древовидной машины четности на стойкость процедуры исправления ошибок к атакам.

Исследование выполнено при поддержке Программы развития Томского государственного университета (Приоритет-2030).

Литература

1. Бауместер Д. Физика квантовой информации / Д. Бауместер, А. Экерт, А. Цайлингер. – М.: Постмаркет, 2002. – 376 с.

2. Niemiec M. Error correction in quantum cryptography based on artificial neural networks // Quantum Information Processing. – 2019. – Vol. 18. – Art. 174.

3. Modular quantum key distribution setup for research and development applications / V.E. Rodimin, E.O. Kiktenko, V.V. Usova, M.Y. Ponomarev, T.V. Kazieva, A.V. Miller, A.S. Sokolov, A.A. Kanapin, A.V. Losev, A.S. Trushechkin, M.N. Anufriev, N.O. Pozhar, V.L. Kurochkin, Y.V. Kurochkin, A.K. Fedorov // Journal of Russian Laser Research. – 2019. – Vol. 40, No. 3. – PP. 221–229.

4. Klimov A. Analysis of Neural Cryptography / A. Klimov, A. Mityagin, A. Shamir // Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology. – Berlin, Heidelberg: 2002. – PP. 286–296.

Каширский Данила Евгеньевич
Канд. физ.-мат. наук, доцент каф. квантовой электроники и фотоники радиофизического факультета
Национального исследовательского
Томского государственного университета (НИ ТГУ)
Эл. почта: kde@mail.tsu.ru

Якименко Филипп Александрович
Магистрант каф. квантовой электроники и фотоники радиофизического факультета НИ ТГУ
Эл. почта: filipp.yakimenko.98@mail.ru

УДК 004.056.5

А.С. Звиденцева, Е.Ю. Костюченко

Использование методов ансамблирования для обнаружения DDoS-атаки

Рассматриваются стекинг и бэггинг для обнаружения DDoS-атаки на наборе данных CICDDoS2019. Результаты бэггинга были сравнены с результатами оригинальных методов с применением кроссвалидации и критерия Уилкоксона для оценки значимости расхождений при уровне значимости 0,01. Получены значения функции качества F1-мера.

Ключевые слова: DDoS-атака, ансамблирование, бэггинг, стекинг, CICDDoS2019, дерево решений, случайный лес, k-ближайших соседей, наивный байесовский алгоритм.

Компьютерные и информационные технологии сейчас являются неотъемлемой частью нашего мира. Внедряются различные решения, которые повышают уровень производительности организаций. Эта положительная сторона ходит бок о бок с отрицательной, наступает риск угрозы.

В данное время наблюдается увеличенное число атак, которые называются «отказ в обслуживании» (Denial of Service, DoS-атаки).

Методы машинного обучения используются в различных задачах информационной безопасности, в частности, аутентификации, анализе трафика, обнаружении вторжения и др. [1]. Сейчас в решении многих задач также применяют подходы построения ансамблей (композиций) методов машинного обучения.

Поэтому целью данной работы является повышение точности алгоритмов обнаружения DDOS-атаки с помощью ансамблирования.

Ансамблирование

Сейчас при решении многих задач используются методы ансамблирования алгоритмов машинного обучения. Применительно к решению задачи классификации процесс ансамблирования основан на компенсации ошибок простых классификаторов, называемых также базовыми, за счет различных способов комбинирования и комбинирования их классификационных решений.

В данной работе будет рассмотрено два метода: бэггинг и стекинг.

При стекинге используется несколько разнородных классификаторов. Их обучают и объединяют для построения прогноза, основанного на их результатах [2].

Работа этого типа ансамблей довольно проста. На вход всех классификаторов подаётся обучающий набор, затем каждый прогноз идёт к финальной модели, которая называется мета-ученик или мета-модель, после чего та вырабатывает финальный прогноз [2].

В бэггинге однородные модели обучают на разных наборах данных и объединяют. Получают прогноз путём усреднения либо голосования [2].

Основная идея бэггинга заключается в том, чтобы обучить несколько одинаковых моделей на разных образцах данных. Распределение выборки неизвестно, поэтому модели получаются разными [2].

Для начала генерируется несколько бутстрэп-выборок. Бутстрэп-выборка – это случайный выбор данных из датасета и представление их в модель. Затем данные возвращаются в датасет, и процесс повторяется. После модели делают свои прогнозы на основе бутстрэп-выборок. В случае регрессии прогнозы просто усредняются. В случае же классификации применяется голосование [2].

Результат экспериментов

В качестве используемого набора данных была взята подвыборка DDoS-атак набора данных CICDDoS2019, состоящая из 279 279 записей, из них 25 761 относятся к классу «нет атаки», а остальные – к классу «есть атака».

Были заданы следующие параметры для моделей машинного обучения:

– KNeighborsClassifier (n_neighbors = 1);

– GaussianNB (var_smoothing = 0.2848035868435802);

– DecisionTreeClassifier (max_depth = 8, max_features = 8, min_samples_leaf = 2, min_samples_split = 9);

– RandomForestClassifier (max_depth = 12, min_samples_leaf = 2, min_samples_split = 3, n_estimators = 50).

При применении бэггинга были получены значения F1-меры, чтобы их проанализировать, были применены кроссвалидация и критерий Уилкоксона при уровне значимости 0,01.

На рис. 1–4 представлены гистограммы значений F1-меры кроссвалидации для четырех реализаций бэггинга и четырех «оригинальных» методов.

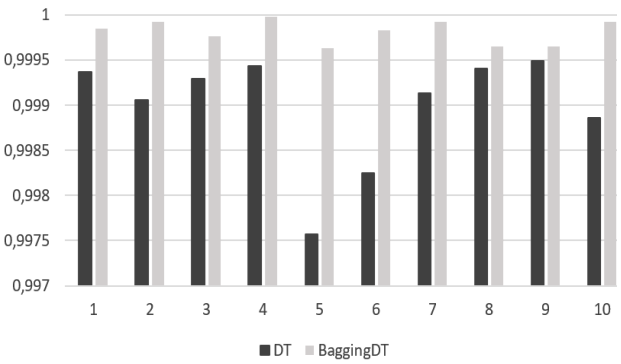


Рис. 1. Гистограмма кроссвалидации для метода дерева решений и бэггинга на его основе

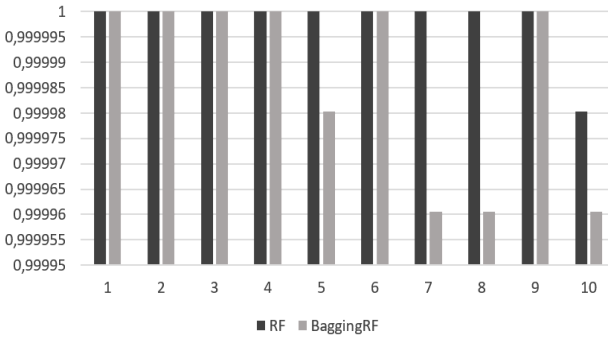


Рис. 2. Гистограмма кроссвалидации для метода случайного леса и бэггинга на его основе

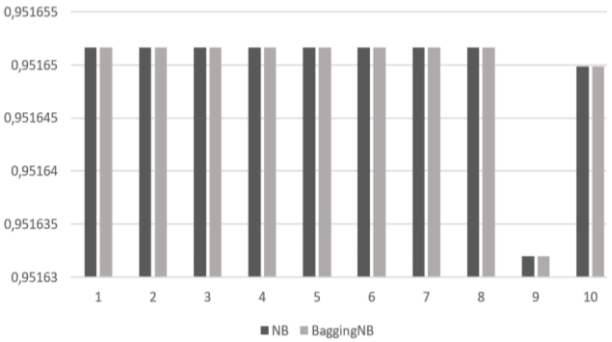


Рис. 3. Гистограмма кроссвалидации для метода наивного Байесовского классификатора и бэггинга на его основе

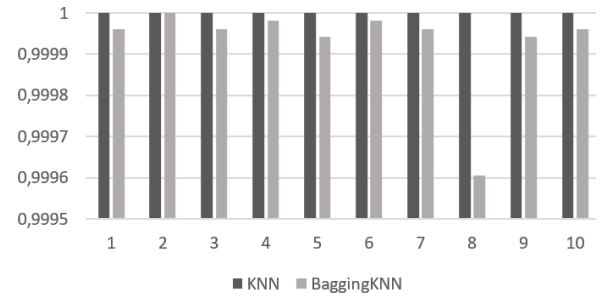


Рис. 4. Гистограмма кроссвалидации для метода К-ближайших соседей и бэггинга на его основе

Стоит отметить, что полученные результаты бэггинга для метода дерева решений дал значительно лучший результат по критерию Уилкоксона. Полученные результаты бэггинга для метода случайного леса незначимо лучше по критерию Уилкоксона. Полученные результаты для бэггинга и «оригинального» метода наивного Байесовского классификатора оказались идентичными. А результаты бэггинга для метода К-ближайших соседей оказались значимо хуже по критерию Уилкоксона.

Реализован также и метод ансамблирования стекинга для разных наборов классификаторов:

1. KNN при k = 1, k = 2, k = 3.
2. DT, NB, KNN при k = 1.
3. DT, NB, KNN при k = 1, k = 2, k = 3.
4. RF, NB, KNN при k = 1.
5. RF, NB, KNN при k = 1, k = 2, k = 3.

При этом тестовая выборка варьировалась от 10 до 90% от общего числа записей подвыборки с интервалом в 10%.

На рис. 5–8 представлены гистограммы значений F1-меры для стекинга пяти наборов классификаторов.

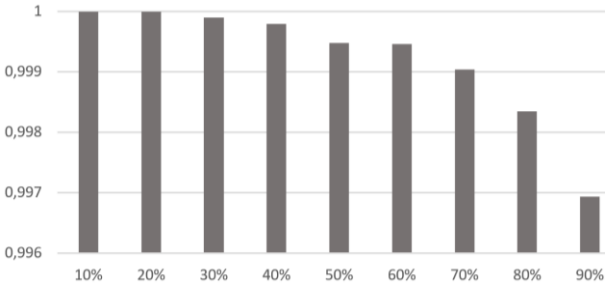


Рис. 5. Гистограмма значений F1-меры для стекинга набора классификаторов № 1

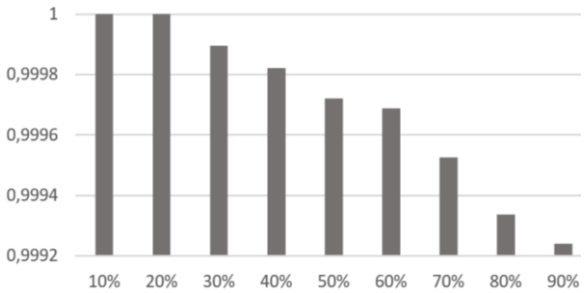


Рис. 6. Гистограмма значений F1-меры для стекинга набора классификаторов № 2

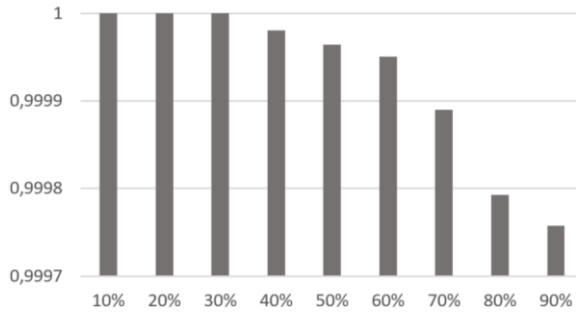


Рис. 7. Гистограмма значений F1-меры для стекинга набора классификаторов № 4

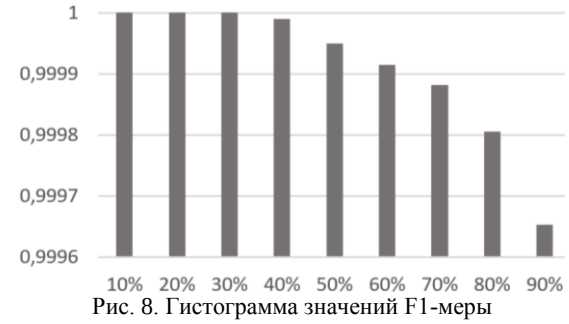


Рис. 8. Гистограмма значений F1-меры для стекинга набора классификаторов № 5

Также были найдены статьи, в которых авторы используют набор данных CICDDoS2019. Так как в них используются значения Accuracy (точность), то для 13 реализованных классификаторов были получены значения точности для тестовой выборки от 20 до 50% с шагом в 10%.

В статье [3] используется метод классификации случайного леса при тестовой выборке 20%. Авторы данной статьи получили значение точности 99,97%.

В нашем же случае при тестовой выборке 20% при использовании «оригинального» метода случайного леса, бэггинга этого классификатора и стекинга с использованием случайного леса были получены результаты точности 100%.

В статьях [4–6] при тестовой выборке 30% используется метод классификации k-ближайших соседей. Получены значения точности 81,94; 99,7; 99,19% соответственно. В нашем исследовании точность превышала значение 99,9%. В [4] авторы также использовали и классификатор случайного леса, получив 83,07% точности. В нашем же случае результаты точности достигали 100%. В статьях [5–8] авторы используют классификатор дерева решений и получают следующие значения точности соответственно: 99,85; 99,2; 97; 88,67%. В нашем исследовании данный классификатор при данной тестовой выборки достиг 99,89% точности, бэггинг данного метода достиг 99,99%, точность стекинга с использованием данного классификатора также превысила значение 99,92 и 100%. В статьях [6–8] используют наивный байесовский классификатор и получают значение точности: 98,92; 96,25; 99,08%. В нашем эксперименте точность не превысила 91% для «оригинального» метода и бэггинга, а стекинг достиг 100%.

В статьях [9–11] при тестовой выборке 40% используют классификатор дерева решений, наивный байесовский классификатор и случайный лес. Для дерева решений получили следующие значения точности: 99,82; 98,5; 99%. При использовании наивного байесовского классификатора: 99,61; 77; 45%. При использовании случайного леса: 99,83; 98,55; 99%. В нашем случае дерево решений, бэггинг и стекинг дали результаты, превышающие 99,9% точности. Наивный байесовский классификатор выдал точность 90,78%, бэггинг 90,81%, а стекинг с данным классификатором превысил 99,9%. Случайный лес показал точность классификации 99,99%, бэггинг достиг 100% точности, стекинг также показал 100% точность.

В статье [12] авторы используют классификаторы k-ближайших соседей и случайного леса при объеме тестовой выборки 50%, получив 99,81 и 95,42% точности. В нашем случае значения точности превысили 99,9% для всех вариантов исследования.

Выводы

При выполнении работы были изучены методы ансамблирования, реализованы бэггинг и стекинг. Работоспособность проверена на подвыборке DDoS-атак набора данных CICDDoS2019. Результаты бэггинга были сравнены с результатами оригинальных методов с применением кроссвалидации и критерия Уилкоксона для оценки значимости расхождений при уровне значимости 0,01.

Изучив результаты проделанной работы, можно сделать следующие выводы:

– результаты с применением бэггинга дерева решений значимо лучше по сравнению с чистым классификатором;

– результаты с применением бэггинга случайного леса оказался незначимо лучше по сравнению с чистым классификатором;

– результаты наивного байесовского классификатора и бэггинга оказались идентичны;

– результаты с применением бэггинга k-ближайших соседей (при k = 1) оказался значимо хуже;

– стекинг, построенный по методу k-ближайших соседей при k = 1, k = 2, k = 3, дает лучшей результат чем «оригинальные» методы;

– стекинг, построенный по методам дерева решений, наивного Байеса и k-ближайших соседей при k = 1, также показывает лучший результат;

– стекинг, построенный по методам дерева решений, наивного Байеса и k-ближайших соседей при k = 1, k = 2, k = 3, дает лучший результат;

– стекинг, построенный по методам случайного леса, наивного Байеса и k-ближайших соседей при k = 1, также дает нелучший результат по сравнению с методом случайного леса, но лучше методов наивного Байеса и k-ближайших соседей при k = 1;

– стекинг, построенный по методам случайного леса, наивного Байеса и k-ближайших соседей при k = 1, k = 2, k = 3, дает нелучший результат по сравнению с методом случайного леса, но лучше остальных методов;

– в большинстве случаев полученные результаты показывают лучшую точность при сравнительном анализе с другими исследованиями, в которых используется тот же набор данных.

Литература

1. Шелупанов А.А. и др. Актуальные направления развития методов и средств защиты информации // Доклады ТУСУР. – 2017. – Т. 20, № 3. – С. 11–24.

2. Ансамблевые методы машинного обучения [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/571296/, свободный (дата обращения: 18.02.2022).

3. A new proactive feature selection model based on the enhanced optimization algorithms to detect DRDoS attacks. [Электронный ресурс]. – Режим доступа: http://ijece.iaescore.com/index.php/IJECE/article/view/25619 (дата обращения: 05.05.2022).

4. Kishore Babu Dasari, Nagaraju Devarakonda. TCP/UDP-Based Exploitation DDoS Attacks Detection Using AI Classification Algorithms with Common Uncorrelated Feature Subset Selected by Pearson, Spearman and Kendall Correlation Methods // Revue d'Intelligence Artificielle. – February, 2022. – Vol. 36, No. 1. – PP. 61–71.

5. A generalized machine learning model for DDoS attacks detection using hybrid feature selection and hyperparameter tuning [Электронный ресурс]. – Режим доступа: https://www.sciencedirect.com/science/article/abs/pii/S1389128621004394?via%3Dihub (дата обращения: 05.05.2022).

6. Security Analysis of DDoS Attacks Using Machine Learning Algorithms in Networks Traffic [Электронный ресурс]. – Режим доступа: https://www.researchgate.net/publication/356535476_Security_Analysis_of_DDoS_Attacks_Using_Machine_Learning_Algorithms_in_Networks_Traffic (дата обращения: 05.05.2022).

7. Thapanarath Khempetch, Pongpisit Wuttidittachotti. DDoS attack detection using deep learning // IAES International Journal of Artificial Intelligence (IJ-AI). – June 2021. – Vol. 10, No. 2. – PP. 382–388.

8. Detection of DDoS attacks with feed forward based deep neural network model [Электронный ресурс]. – Ре-

жим доступа: https://www.sciencedirect.com/science/article/abs/pii/S0957417420311647?via%3Dihub (дата обращения: 05.05.2022).

9. Detecting Distributed Denial of Service Attacks using Machine Learning Models. [Электронный ресурс]. – Режим доступа: https://www.researchgate.net/publication/303704348_Distributed_Denial_of_Service_DDoS_Attacks_Detection_Using_Machine_Learning_Prototype (дата обращения: 05.05.2022).

10. A Deep-Learning Model for Detecting Software Defined Healthcare IoT Networks Attacks [Электронный ресурс]. – Режим доступа: https://link.springer.com/ chapter/10.1007/978-3-030-86356-2_17?error=cookies_not_ supported&code=0fa64b0b-36f8-4f5f-902a-f2b249f2c1da (дата обращения: 05.05.2022).

11. AE-MLP: A Hybrid Deep Learning Approach for DDoS Detection and Classification [Электронный ресурс]. – Режим доступа: https://www.researchgate.net/ publication/355691427_AE-MLP_A_Hybrid_Deep_Learning_Approach_for_DDoS_Detection_and_Classification (дата обращения: 05.05.2022).

12. SDN-Based Architecture for Transport and Application Layer DDoS Attack Detection by Using Machine and Deep Learning [Электронный ресурс]. – Режим доступа: https://www.researchgate.net/publication/353780262_SDN-Based_Architecture_for_Transport_and_Application_Layer_DDoS_Attack_Detection_by_Using_Machine_and_Deep_Learning (дата обращения: 05.05.2022).

Звиденцева Анна Сергеевна
Студентка каф. безопасности информационных систем (БИС) ТУСУРа
Эл. почта: zas2302@mail.ru

Костюченко Евгений Юрьевич
Канд. техн. наук, доцент каф. комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС) ТУСУРа
Эл. почта: key@keva.tusur.ru, key@fb.tusur.ru

УДК-004.712

В.В. Ботвинко, Г.Р. Хазанкин

Разработка интерфейсного модуля для обработки сетевого трафика

Рассматривается возможность реализации в гибридной вычислительной системе решения для задач Data plane (уровня данных) с использованием технологии FPGA (программируемой пользователем вентильной матрицы, ППВМ). В ходе проведенного исследования были применены эффективные перспективные решения для тернарной ассоциативной памяти (TCAM) и для передачи сетевых пакетов с их метаданными.
Ключевые слова: Data Plane, FPGA, TCAM, Heterogeneous computing, сетевой интерфейс.

Постоянное развитие телекоммуникационных сетей приводит к потребности увеличения скорости обработки сетевого трафика. Современное решение CEF (Cisco Express Forwarding) использует CPU для определения наилучшего маршрута для перенаправления сетевых пакетов, а уже подготовленная таблица маршрутизации преобразуется в структуру данных, которая находится в Hardware CEF, представляющем собой тернарную ассоциативную память (TCAM). Данное архитектурное решение является общепринятым.

Перспективные решения

В создании решений для обработки больших объемов данных в высокоскоростном режиме одним из направлений является развитие гибридных вычислительных систем, распределяющих нагрузку между несколькими вычислительными устройствами, стремясь таким образом решить проблему «бутылочного горлышка» при обработке на единственном вычислительном устройстве, не успевающем за поступающими пакетами [1].

Среди таких решений перспективным является применение CPU совместно с GPU, позволяющее использовать большое количество физических потоков для параллельного выполнения задач перенаправления [6]. Одним из недостатков системы была выявлена проблема обновления таблицы маршрутизации, содержащейся в памяти GPU, без потери производительности системы, а также отставание по пропускаемой способности от «традиционных» решений для перенаправления сетевого трафика [2].

Традиционным подходом является использование специализированных аппаратурных решений (ASIC), выполняющих задачи, связанные с предобработкой получаемых пакетов, их фильтрацией и отнесением к определенному потоку данных.

Большинство аппаратурных решений для высокоскоростного перенаправления пакетов в маршрутизаторах делятся на две основные категории: тернарная ассоциативная память (TCAM) и динамическая / статическая оперативная память (решения на основе DRAM/SRAM). TCAM при этом имеет как очевидное преимущество в виде поиска искомой информации за один такт, так и фатальные недостатки в виде используемых ресурсов и сложности при параметризации по ширине и глубине [3].

Реализация TCAM на FPGA

Среди аппаратурных решений одним из наиболее высокоскоростных для обработки сетевого трафика является FPGA (Field Programmable Gate Array) – программируемая пользователем вентильная матрица, которая может быть перепрограммирована при необходимости изменить принципиальную схему, а за счет низкоуровневой обработки, задержки при использовании FPGA остаются сравнимыми с ASIC.

Использование ресурсов ППВМ для реализации TCAM является одним из способов решения проблемы недостаточной гибкости для использования в конкретных решениях. При этом среди реализаций тернарной ассоциативной памяти существуют различные способы её размещения на кристалле ППВМ. Среди них самой перспективной методологией является использование непосредственно LUTs – Look Up Tables и встроенных цепочек переноса фрагментов для одновременного сопоставления двух правил и логики их сопоставления с одной секцией ПЛИС [4]. Сравнение с предлагаемым решением от Xilinx представлено в таблице.

Сравнение используемых ресурсов архитектурами TCAM [4]

Архитектура	Размер TCAM	LUTRAMs	LUTs	BRAMs
BPR-TCAM I	512×40	0	2560	0
BPR-TCAM II	1024×144	0	18432	0
Xilinx	512×128	8875	27559	3

Вышеупомянутые работы позволяют рассмотреть возможность реализации решений для задач перенаправления с применением технологии ППВМ.

Описание интерфейсного модуля

Рассматривается возможность построения интерфейсного модуля с применением FPGA в качестве основного устройства для решения следующих задач:

– Чтение байт из буфера физического интерфейса и обработка заголовков кадра канального уровня на входном тракте.

– Передача в TCAM сформированных метаданных, перезапись заголовков кадра.

– Формирование кадра на выходном тракте для передачи в буфер физического интерфейса для отправки.

Уровни проектирования ППВМ

Само проектирование логики принципиальной схемы современных ППВМ возможно на 3 уровнях:

– низком, подразумевающем применение языка описания аппаратуры для реализации необходимой логики с точностью до такта;

– блочном, основанном на использовании готовых решений от производителей ППВМ;

– высоком, основанном на трансляции описанной логики на высокоуровневом языке программирования на аппаратурный уровень.

Подсистема перенаправления

При реализации захвата байт кадра с физического интерфейса и передачи кадра на него предлагается применить блочный уровень проектирования, задействовав предложенные решения [5].

Основной частью системы является Forwarding Engine – подсистема перенаправления, выполняющая обработку заголовков кадра, обращаясь к аппаратурным таблицам перенаправления для определения дальнейших действий с кадром.

Реализованный прототип подсистемы перенаправления обрабатывает заголовки канального, сетевого и транспортного уровней в соответствии с рис. 1. За обработку на каждом из уровней отвечает отдельно реализованные на языке описания аппаратуры модули, каждый из которых сообщает о наличии инкапсуляции соответствующего уровня модулю и передает необходимые метаданные для дальнейшего перенаправления.

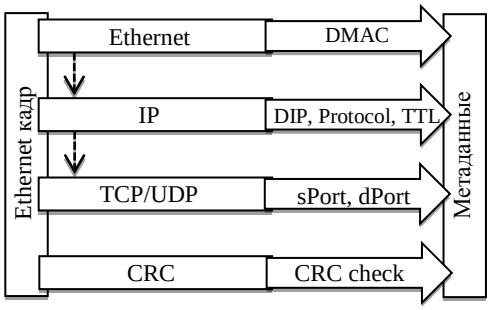


Рис. 1. Функциональная схема предобработки

Каждый из модулей отдельно и прототип были протестированы в симуляции поведения описанной логики. Для этого с применением несинтезируемой логики на языке описания аппаратуры были созданы testbench-модули, которые симулируют тактовую частоту, сигналы сброса и передачу данных.

Применимость FPGA

В рамках данной системы предусматривается использование ППВМ для решения задач на уровне перенаправления трафика (Data Plane), а уровень управления (Control Plane) остается за более гибким решением в виде CPU и/или GPU. Использование существующих решений для прямого доступа к памяти при помощи DMA-контроллера на блочном уровне проектирования [5] позволит соответствовать принципу передачи данных zero-copy. При этом использование DMA позволит масштабировать систему по количеству используемых устройств на Data Plane.

Анализ использованных ресурсов ППВМ

Для расчета использованных ресурсов кристалла ППВМ использовалась построенная после этапа синтезирования схема для устройства Xilinx Artix 7-series 28 nm XC7A100TCSG324-1. Выбор данного устройства обусловлен использованием его спецификации при расчете используемых ресурсов перспективными реализациями TCAM [4]. Также данное седьмое поколение устройств является актуальным решением, полностью поддерживаемым Vivado v2021.2.

Для отладки реализованной логики отдельных модулей и системы обработки сетевого кадра в целом использовались захваченный на домашнем устройстве трафик при помощи приложения Wireshark [7], основанного на библиотеке libpcap [8], и свободно распространяемые .cap-файлы на ресурсе PacketLife.net [9]. Для отладки отдельных модулей производилось декапсулирование пакетов уровня соответствующему уровню обработки модулем.

Количество используемых ресурсов в синтезированной схеме, в первую очередь, зависит от значения параметра ширины входного порта данных. Помимо необходимого количества элементов для соединения портов, это также обусловлено реализованной логикой поиска полей заголовков, которая за один такт обрабатывает данные размером в заданную ширину порта. Используемые идентификаторы для сравнения со значениями полей являются частью данной логики.

На рис. 2 изображен график зависимости, используемой синтезированной схемой LUTs, от значения ширины порта входных данных. На основе приведенных результатов было выявлено несколько закономерностей. На рис. 3 изображен график изменения количества используемых синтезированной схемой триггеров от параметра ширины порта входных данных.

Рис. 2. Зависимость количества используемых схемой LUTs от ширины входного порта данных

Во-первых, ширина, значение которой не является четным числом байт, является менее эффективной по сравнению с большей шириной, имеющей четное число байт. Во-вторых, количество используемых LUTs увеличивается пропорционально увеличению ширины входного порта отдельно для четных и нечетных значений. Учитывая общее количество

LUTs рассматриваемого устройства, относительное использование данных элементов достигает значения порядка 1% при ширине 14 байт.

Рис. 3. Зависимость используемых синтезированной схемой Flip Flops от ширины входного порта данных

Количество используемых Flip-Flops, главным образом, обусловлено количеством состояний реализованной последовательной логики. Выявленное колебание значения количества используемых Flip-Flops связано с расположением необходимой синхронизирующей логики относительно размещенной в LUT, оптимально синтезированной для архитектуры Artix 7-series. Учитывая общее количество триггеров рассматриваемого устройства, относительное использование данных элементов составляет 0,05%.

Выводы

Анализ использования ресурсов построенным решением для Data Plane показал, что данное решение совместимо с эмуляцией TCAM. Однако ещё необходимо исследовать зависимость использованных ресурсов относительно количества правил, применяемых для обработки сетевого трафика.

В заключение данной работы можно подвести итог: при эффективном использовании ресурсов кристалла FPGA на нем возможно построить решение для Data Plane в гибридной системе высокоскоростной обработки сетевого трафика с возможностью поддержки zero-copy.

Литература

1. Анализ сетевого трафика в режиме реального времени: обзор прикладных задач, подходов и решений / А.И. Гетьман, Е.Ф. Евстропов, Ю.В. Маркин // ИСП РАН. – 2015. – Вып. 28. – С. 1– 52 (препринт).

2. PacketShader: a GPU-accelerated software router / S. Han, K. Jang, K. Park, and S. Moon // In SIGCOMM, 2010.

3. Deep M. Network Routing Algorithms, Protocols, and Architectures / Medhi Deep, Ramasamy Karthik. – 2 ed. – Cambridge, United States: Elsevier, 2018. – 982 p.

4. BPR-TCAM-Block and Partial Reconfiguration based TCAM on Xilinx FPGAs / Ullah Anees, Zahir Ali, Khan Noaman, Ahmad Waleed, Ramos Alexis, Reviriego Pedro // Electronics. – 2020. – Vol. 9. – 353 p.

5. Xilinx Documentation Portal: сайт. – URL: https://docs.xilinx.com/ (дата обращения: 25.04.2022)/

6. Khazankin G.R. et al. System architecture for deep packet inspection in high-speed networks // 2017 Siberian Symposium on Data Science and Engineering (SSDSE). – IEEE, 2017. – PP. 27–32.

7. Wireshark [Электронный ресурс]. – URL: https://www.wireshark.org/ (дата обращения: 05.04.2022).

8. Man page of pcap [Электронный ресурс]. – URL: https://www.tcpdump.org/manpages/pcap.3pcap.html (дата обращения: 05.04.2022).

9. PacketLife.net [Электронный ресурс]. – URL: https://packetlife.net/captures (дата обращения: 05.04.2022).

Ботвинко Виталий Вячеславович
Ассистент каф. общей информатики (КОИ)
Новосибирского государственного университета (НГУ)
Эл. почта: botvinko@nsuacad.net

Хазанкин Григорий Романович
Ст. преп. КОИ НГУ
ORCID: https://orcid.org/0000-0002-8465-1593
Эл. почта: khazankin@nsuacad.net

УДК 621.396

Ю.Э. Корчагин, К.Д. Титов, О.Н. Завалишина

Различение двух узкополосных радиосигналов с неизвестными амплитудами и начальными фазами

Выполнен синтез и анализ алгоритма различения двух узкополосных радиосигналов произвольной формы с неизвестными начальными фазами и амплитудами, наблюдаемых на фоне белого гауссовского шума. Для синтеза алгоритма различения использовался метод максимального правдоподобия, согласно которому на основе наблюдаемых данных формировался логарифм функционала отношения правдоподобия в зависимости от неизвестных параметров сигнала. Исследованы статистические характеристики случайных величин и получено выражение для средней вероятности ошибки, характеризующее эффективность синтезированного алгоритма различения.
Ключевые слова: скрытность, различение, узкополосный, радиосигнал, средняя вероятность ошибки.

Для повышения помехоустойчивости и скрытности работы современных систем связи и передачи данных необходимо снижать мощность передающего устройства, тем самым уменьшая энергию передаваемых сигналов. Однако это затрудняет обнаружение сигнала на приёмной стороне, а в условиях высокой плотности абонентов на приемник средства радиосвязи или передачи данных поступает множество сигналов со схожей структурой, вследствие чего простые обнаружители с большой вероятностью будут выносить ошибочное решение и принимать сигнал с большей энергией [1].

Для достоверной обработки полезного сигнала необходимо использовать сложные комбинированные алгоритмы обнаружения и оценки параметров сигнала, поэтому для повышения эффективности функционирования и упрощения структуры приемного устройства применяют схемы различения сигналов. Использование алгоритмов различения сигналов позволяет установить факт наличия в принимаемой реализации других сигналоподобных помех, адаптировать устройство к приёму такой аддитивной смеси и оптимальным образом осуществить обработку полезного сигнала.

В работе [2] подробно исследованы оптимальные алгоритмы различения узкополосных радиосигналов с неизвестными начальными фазами, однако в условиях сложной помеховой обстановки при пространственном разнесении абонентов часто оказываются неизвестными амплитуды сигналов. В связи с этим в данной работе выполнен синтез и анализ алгоритмов различения двух узкополосных радиосигналов с различными модулирующими функциями, неизвестными начальными фазами и амплитудами.

Постановка задачи

Рассмотрим задачу различения двух узкополосных радиосигналов с неизвестными амплитудами a и начальными фазами φ. Пусть на интервале времени $t \in [0,T]$ доступна наблюдению реализация

$$\xi(t) = s_k(t, a_{0k}, \varphi_{0k}) + n(t), \qquad (1)$$

аддитивной смеси одного из двух возможных сигналов $s_k(t, a_{0k}, \varphi_{0k})$, $k = 1,2$, в котором a_{0k}, φ_{0k} – истинные значения амплитуды и начальной фазы сигнала, k – номер сигнала, который в действительности присутствует в принятой реализации, и гауссовского белого шума $n(t)$ с односторонней спектральной плотностью N_0. Различаемые сигналы представляют собой узкополосные радиосигналы вида

$$s_i(t, a_i, \varphi_i) = \begin{cases} a_i f_i(t)\cos(\omega_i t - \varphi_i), & 0 \le t \le \tau_i, \\ 0, & t < 0, t > \tau_i \end{cases} \qquad (2)$$

и различаются модулирующими функциями $f_i(t)$, частотами формирующего гармонического колебания ω_i, а также длительностями τ_i, где $i = 1,2$. Амплитуду a_i и начальную фазу φ_i различаемых сигналов будем считать неизвестными. Если полосы частот $\Delta\omega_i$ и частоты ω_i сигналов (2) удовлетворяют условию $\Delta\omega_i \ll \omega_i$, то сигналы (2) являются узкополосными, а $f_i(t)$ – их огибающими [2, 3]. Случайные фазы φ_i считаются независимыми и равномерно распределёнными в интервале $[-\pi, \pi]$. Располагая принятой реализацией (1), необходимо решить оптимальным образом, какой из двух возможных сигналов присутствует на входе приёмного устройства.

Сформулируем задачу различения в терминах теории проверки статистических гипотез. Введём в рассмотрение гипотезы H_i, соответствующие наличию в наблюдаемой реализации сигнала с номером $i = 1,2$. Обозначим $p_i = P(H_i)$ – априорные вероятности гипотез. Поскольку априори неизвестны амплитуды и начальные фазы различаемых сигналов, гипотезы являются сложными. Алгоритм различения на основе анализа принятой реализации должен выносить решение в пользу одной из гипотез.

Синтез алгоритма различения

Синтез алгоритма различения узкополосных радиосигналов выполним методом максимального правдоподобия (МП) [2, 4]. Логарифм функционала отношения правдоподобия (ФОП) для различаемых сигналов зависит от номера i и неизвестных амплитуд и начальных фаз

$$L_i(a_i, \varphi_i) = \frac{2}{N_0}\int_0^{\tau_i} \xi(t) s_i(t, a_i, \varphi_i)\,dt - \frac{1}{N_0}\int_0^{\tau_i} s_i^2(t, a_i, \varphi_i)\,dt. \qquad (3)$$

Решение в пользу одной из гипотез выносится на основе сравнения с порогом отношения правдоподобия

$$\frac{L_2}{L_1} \underset{H_2}{\overset{H_1}{\gtrless}} h, \qquad (4)$$

где $L_i = \sup\limits_{a_i,\varphi_i} L_i(a_i,\varphi_i)$ – абсолютный (наибольший) максимум решающей статистики (4) [2]. Так как для задач различения чаще более обоснованным является применение критерия идеального наблюдателя [2], то все рассуждения будем вести, базируясь именно на этом критерии, согласно которому порог должен быть выбран как $h = p_1/p_2$.

Подставив в выражение (4) явный вид узкополосного радиосигнала (2) с номером i, перепишем логарифм ФОП в виде

$$L_i(a_i,\varphi_i) = a_i\left(X_i\cos\varphi_i + Y_i\sin\varphi_i\right) - \frac{a_i^2 Q_i}{2}, \quad (5)$$

где $X_i = \frac{2}{N_0}\int\limits_0^{\tau_i}\xi(t)\,f_i(t)\cos\omega_i t\;dt,$

$$Y_i = \frac{2}{N_0}\int\limits_0^{\tau_i}\xi(t)\,f_i(t)\sin\omega_i t\;dt, \quad Q_i = \frac{1}{N_0}\int\limits_0^{\tau_i} f_i^2(t)dt. \quad (6)$$

Для получения решающей статистики выполним аналитически максимизацию логарифма ФОП (6) по переменным a_i и φ_i. Составив и решив систему уравнений правдоподобия и далее подставив полученные решения в (6), имеем

$$L_i = \sup_{a_i,\varphi_i} L_i(a_i,\varphi_i) = \frac{X_i^2 + Y_i^2}{2Q_i}. \quad (7)$$

Анализ алгоритма различения

Выполним анализ алгоритма различения (5), т.е. найдём среднюю вероятность ошибки. При функционировании приёмного устройства в соответствии с алгоритмом различения (5) могут присутствовать ошибки 1-го и 2-го рода, вероятности появления которых обозначим $\alpha = P\left(H_2\middle|H_1\right)$ и $\beta = P\left(H_1\middle|H_2\right)$ соответственно. Тогда средняя вероятность ошибки определяется выражением $p_e = p_1\alpha + p_2\beta$. Поскольку приёмное устройство функционирует в условиях полной априорной неопределённости, положим далее, что априорные вероятности гипотез совпадают, т.е. $p_1 = p_2 = 1/2$. Тогда порог различения $h = 1$, средняя вероятность ошибки равна $p_e = (\alpha+\beta)/2$, а для вероятностей ошибок 1-го и 2-го рода можно записать

$$\alpha = P\left(L_2 > L_1\middle|H_1\right), \quad \beta = P\left(L_1 > L_2\middle|H_2\right).$$

Обозначим $W_2\left(l_1,l_2\middle|H_j\right)$ – условная совместная плотность вероятности случайных величин (СВ) L_1 и L_2 при справедливости гипотезы H_k. Тогда согласно [4, 5] для вероятностей ошибок справедливы выражения

$$\alpha = \int\limits_{-\infty}^{\infty}\int\limits_{l_1}^{\infty} W_2\left(l_1,l_2\middle|H_1\right)dl_2dl_1, \beta = \int\limits_{-\infty}^{\infty}\int\limits_{l_2}^{\infty} W_2\left(l_1,l_2\middle|H_2\right)dl_1dl_2. \quad (8)$$

Случайные величины L_1 и L_2 зависят от СВ X_i, Y_i, которые являются гауссовскими, так как представляют собой линейные преобразования гауссовского белого шума $n(t)$. Следовательно, СВ X_i, Y_i полностью описываются первыми двумя моментами. Представим величины X_i, Y_i в виде сумм детерминированных и случайных составляющих при условии справедливости гипотезы H_k, т.е. когда в наблюдаемой реализации присутствует сигнал $s_k(t,a_k,\varphi_k)$.

$$X_i = S_{xik} + N_{xi}, \; Y_i = S_{yik} + N_{yi},$$

где введены следующие обозначения:

$$S_{xik} = \frac{2a_k}{N_0}\int\limits_0^{\min(\tau_i,\tau_k)} f_i(t)\,f_k(t)\cos(\omega_k t - \varphi_k)\cos\omega_i t\,dt,$$

$$S_{yik} = \frac{2a_k}{N_0}\int\limits_0^{\min(\tau_i,\tau_k)} f_i(t)\,f_k(t)\cos(\omega_k t - \varphi_k)\sin\omega_i t\,dt,$$

$$N_{xi} = \frac{2}{N_0}\int\limits_0^{\tau_i} n(t)\,f_i(t)\cos\omega_i t dt, N_{yi} = \frac{2}{N_0}\int\limits_0^{\tau_i} n(t)\,f_i(t)\sin\omega_i t dt.$$

Выполнив усреднение, находим математические ожидания $\left\langle X_i\middle|H_k\right\rangle = S_{xik}$, $\left\langle Y_i\middle|H_k\right\rangle = S_{yik}$ и корреляционную матрицу $\mathbf{K}$ СВ X_i, Y_i.

Введём аналогично [3] в выражении (7) замену переменных

$$U_1 = Q_1Y_1, U_2 = Q_2Y_2, U_3 = Q_1X_1, U_4 = Q_2X_2, \quad (9)$$

которая позволяет представить решающую статистику (7) в виде

$$L_1 = \frac{U_1^2 + U_3^2}{2Q_1^3}, \quad L_2 = \frac{U_2^2 + U_4^2}{2Q_2^3}. \quad (10)$$

По определению (9) СВ U_s, $s = 1,2,3,4$ представляют собой линейные комбинации гауссовских СВ X_i, Y_i, а следовательно, и сами являются гауссовскими с математическими ожиданиями

$$m_{1k} = \left\langle U_1\middle|H_k\right\rangle = Q_1S_{y1k}, m_{2k} = \left\langle U_2\middle|H_k\right\rangle = Q_2S_{y2k},$$

$$m_{3k} = \left\langle U_3\middle|H_k\right\rangle = Q_1S_{x1k}, m_{4k} = \left\langle U_4\middle|H_k\right\rangle = Q_2S_{x2k}$$

и элементами корреляционной матрицы $K_{ij} = \left\langle\left(U_i - \left\langle U_i\right\rangle\right)\left(U_j - \left\langle U_j\right\rangle\right)\right\rangle$, одинаковыми при обеих гипотезах. С учётом найденных моментов совместная гауссовская плотность вероятности СВ U_s определяется формулой

$$W_U\left(u_1,u_2,u_3,u_4\middle|H_k\right) = \frac{1}{4\pi^2\sqrt{\det\mathbf{K}}}\times \\ \times\exp\left\{-\frac{1}{2}\sum_{i=1}^{4}\sum_{j=1}^{4}C_{ij}(u_i - m_{ik})(u_j - m_{jk})\right\}, \quad (11)$$

где C_{ij} – элементы матрицы $\mathbf{C} = \mathbf{K}^{-1}$, обратной к корреляционной. Переходим далее от СВ U_s к новым переменным L_1, L_2 (10) и $L_3 = \operatorname{arctg} Y_1/X_1$, $L_4 = \operatorname{arctg} Y_2/X_2$. Учитывая, что $U_1 = \Psi_1\left(L_1,L_3\right)$, $U_2 = \Psi_2\left(L_2,L_4\right)$, $U_3 = \Psi_3\left(L_1,L_3\right)$, $U_4 = \Psi_4\left(L_2,L_4\right)$, и используя правила замены переменных в плотностях вероятностей [3], выполним переход в выражении (11) от совместной плотности вероятности СВ U_s к совместной плотности вероятности СВ L_s

$$W_L\left(l_1,l_2,l_3,l_4|H_k\right)=W_U\left(\Psi_1,\Psi_2,\Psi_3,\Psi_4|H_k\right)|D|, \quad (12)$$

где якобиан преобразования $|D|=\left|\partial\Psi_i/\partial l_j\right|=Q_1^3Q_2^3$, причём $L_1,L_2\geq 0$, $L_3,L_4\in[-\pi,\pi]$. Подставляя в выражение (12) $\Psi_1(l_1,l_3)$, $\Psi_2(l_2,l_4)$, $\Psi_3(l_1,l_3)$, $\Psi_4(l_2,l_4)$, находим совместную плотность вероятности $W_L\left(l_1,l_2,l_3,l_4|H_k\right)$, интегрируя которую по переменным l_3,l_4, получим условную плотность вероятности случайных величин (11).

$$W_2\left(l_1,l_2|H_k\right)=\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}W_L\left(l_1,l_2,l_3,l_4|H_k\right)dl_3dl_4 . \quad (13)$$

Подставив плотность вероятности (13) в формулы (8), находим среднюю вероятность ошибки различения p_e при условии, что априорные вероятности гипотез совпадают, и порог различения $h=1$.

В качестве примера рассмотрим различение двух узкополосных радиосигналов, формы модулирующих функций которых имеют прямоугольную форму. При расчете средней вероятности ошибки различения было введено следующее обозначение:

$$z^2=a_{01}^2\tau_1\Delta/N_0=a_{02}^2\tau_2/N_0 , \quad (14)$$

общее для двух сигналов отношение сигнал/шум (ОСШ) на выходе приёмника МП для двух узкополосных радиосигналов с амплитудами a_{01}, a_{02} и длительностями τ_1, τ_2 соответственно [2, 3], а также отношение энергий двух сигналов, численно равное Δ.

На рис. 1 приведена зависимость средней вероятности ошибки различения p_e от ОСШ z (14) при различных отношениях энергий двух сигналов Δ. Для различаемых сигналов были выбраны различные значения начальной фазы сигналов $\varphi_1=\pi/6$ и $\varphi_2=\pi/3$. Сплошная кривая соответствует равенству энергий сигналов $\Delta=1$, штрихпунктирная – $\Delta=2$, штриховая – $\Delta=0{,}5$.

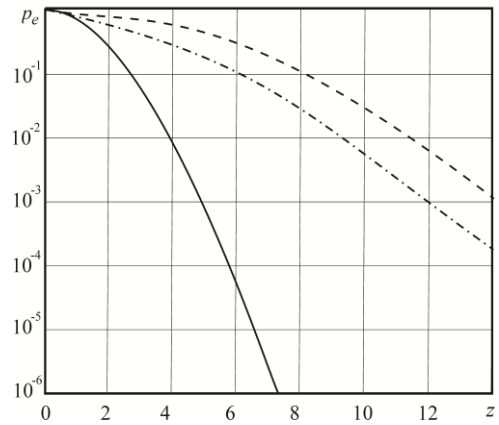


Рис. 1. Зависимость средней вероятности ошибки различения от ОСШ

Из рис. 1 видно, что превышение энергии одного из сигналов в 2 раза относительно другого приводит к существенному росту средней вероятности ошибки. Синтезированный алгоритм различения при равенстве энергий сигналов имеет выигрыш в эффективности более чем в 10^4 раз. Оптимальные алгоритмы обнаружения функционируют с требуемой достоверностью при условии значительного превышения уровня полезного сигнала над помеховым, а в случае равенства уровней сигналов оказываются неэффективными. Синтезированный алгоритм различения (5), наоборот, достигает наилучших характеристик при одинаковых уровнях мощности сигналов.

Таким образом, при организации распределенной сети радиосвязи и передачи данных в условиях высокой плотности абонентов, то есть при наличии мешающих сигналов одного уровня мощности, лучше использовать синтезированный алгоритм различения. Достоверность полученных результатов подтверждена статистическим моделированием.

Заключение

В работе исследован максимально правдоподобный алгоритм различения двух узкополосных радиосигналов, наблюдаемых на фоне гауссовского белого шума. Предполагалось, что сигналы модулируются различными функциями произвольной формы и обладают неизвестными амплитудой и начальной фазой. Получены структура и статистические характеристики алгоритма различения.

Полученные в статье результаты могут быть использованы для повышения скрытности, помехоустойчивости и эффективности функционирования существующих систем связи и передачи данных, а также при проектировании перспективных систем, работающих в условиях сложной помеховой обстановки.

Работа выполнена в рамках гранта Президента Российской Федерации для государственной поддержки молодых российских ученых – кандидатов наук (проект МК-767.2021.1.6).

Литература

1. Корчагин Ю.Э. Обнаружение сверхширокополосного квазирадиосигнала на фоне мешающих сигналов / Ю.Э. Корчагин, К.Д. Титов // Радиотехника. – 2020. – № 9. – С. 72–82.

2. Тихонов В.И. Оптимальный прием сигналов. – М.: Радио и связь, 1983. – 320 с.

3. Корчагин Ю.Э., Титов К.Д. Синтез и анализ алгоритмов обработки сверхширокополосных квазирадиосигналов. – Воронеж: Изд. дом ВГУ, 2019. – 128 с.

4. Теория обнаружения сигналов / П.С. Акимов, П.А. Бакут, В.А. Богданович и др.; под ред. П.А. Бакута. – М.: Радио и связь, 1984. – 440 с.

5. Хелстром К. Статистическая теория обнаружения сигналов. – М.: Изд-во иностранной лит-ры, 1963. – 432 с.

Корчагин Юрий Эдуардович
Д-р физ.-мат. наук, доцент, зав. каф. радиофизики Воронежского государственного университета (ВГУ)
Эл. почта: korchagin@phys.vsu.ru

Титов Константин Дмитриевич
Канд. физ.-мат. наук, доцент каф. радиофизики ВГУ
ORCID: 0000-0002-6981-7209
Эл. почта: titovkd@gmail.com

Завалишина Ольга Николаевна
Магистрант каф. радиофизики ВГУ
ORCID: 0000-0001-6664-3146
Эл. почта: onzavalishina@mail.ru

УДК 004.056.53

Я.В. Калинкин, И.А. Тутов

Модуль информационной безопасности в сетях Modbus RTU

Рассматривается статистика инцидентов информационной безопасности АСУ ТП, анализируется защищенность коммуникационного протокола Modbus RTU, рассматриваются существующие решения средств информационной безопасности АСУ ТП, а также описаны промежуточные результаты разработки устройства, осуществляющего анализ и фильтрацию трафика в технологических сетях Modbus RTU.
Ключевые слова: АСУ ТП, ЛСУ, кибербезопасность, Modbus, удаленный доступ.

Управление каким-либо технологическим процессом подразумевает постоянный перенос информации – сигналов управления, значений технологических параметров, данных исполнительных механизмов и т.д. Для осуществления правильной взаимосвязи между исполнительными механизмами и управляющим оборудованием используют автоматизированные системы управления технологическим процессом (АСУ ТП) и создают технологические сети с имеющими конкретное назначение узлами и сегментами. Для управления отдельными контурами практикуется создание локальных систем управления (ЛСУ), входящих в общую АСУ ТП предприятия. Подобные системы включают в себя комплекс измерительных приборов, исполнительных механизмов и контроллерного оборудования, обеспечивая целостность контура и исключая из алгоритма управления присутствие человека.

Параметры технологического процесса могут быть переданы на контроллерное оборудование ЛСУ посредством физического (RS-232, RS-485, Ethernet и т.п.) или беспроводного интерфейса (Wi-Fi, ZigBee и т.п.) с использованием соответствующих модулей приема-передачи [1, 2]. По выбранному интерфейсу транслируются сообщения какого-либо протокола взаимодействия для осуществления записи и чтения информации, обрабатываемой контроллерным оборудованием.

Каждый коммуникационный протокол имеет стандартизированный формат и характерные особенности, которые могут быть как его функциональным достоинством, так и представлять собой уязвимость, в совокупности с возможностью беспроводного доступа к ЛСУ, приводящей к недостаточной защищенности как отдельного контура управления, так и всей АСУ предприятия, вследствие которой становятся возможными инциденты кибербезопасности АСУ.

Изучение статистики инцидентов кибербезопасности АСУ возможно благодаря данным, предоставляемым «Лабораторией Касперского» [3]. Для наглядности на рис. 1 приведена столбчатая диаграмма успешных блокировок вредоносных объектов на компьютерах АСУ РФ за 2020–2022 гг.

В соответствии с предоставленной статистикой за 2021 г., на 42,27% компьютеров АСУ в РФ были зафиксированы атаки вредоносным ПО. 28,67% зафиксированных атак было произведено с использованием сети Интернет. Из всего количества зафиксированных случаев наличия вредоносного влияния на компьютерах АСУ 21,40% были успешно обнаружены и заблокированы. Данный показатель был действителен во втором квартале, в четвертом квартале он уменьшился более чем вдвое.

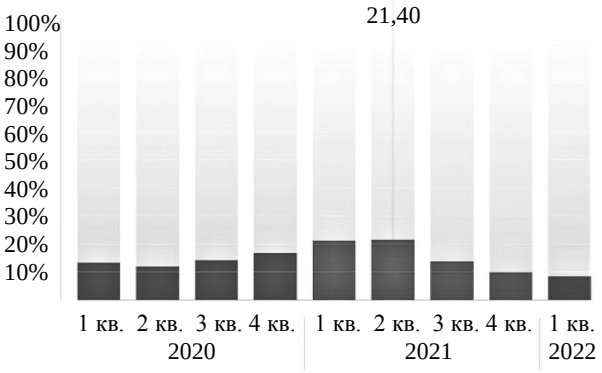


Рис. 1. Статистика компьютеров АСУ в РФ, успешно заблокировавших вредоносное ПО

Таким образом, ссылаясь также на слова директора экспертного центра по промышленной кибербезопасности «Лаборатории Касперского» А. Шипулина, можно оценить общий уровень защищенности предприятий и критической инфраструктуры в РФ как недостаточный. Дополнительным фактором, негативно влияющим на защищенность АСУ, является нежелание предприятий проводить разделение технического и корпоративного сегментов сети предприятия, поскольку для этого предприятию требуется присвоить высокий уровень важности объекта критической инфраструктуры, что требует дополнительных затрат [4].

Исходя из вышеописанных данных, недостаточная защита АСУ ТП в РФ в сфере информационной безопасности является актуальной и значимой проблемой.

На рынке отечественных наложенных средств защиты АСУ ТП представлены такие комплексные решения, как Kaspersky Industrial CyberSecurity (KICS), PT Industrial Security Incident Manager (PT ISIM), DATAPK [5]. Данные решения обеспечивают комплексную информационную защиту АСУ ТП, осуществляя контроль подозрительных действий персонала, целостности программного и аппаратного обеспечения, сетевого окружения, выявляя инциденты информационной безопасности, а также фор-

мируя необходимую отчетность и распространяя ее в другие сопутствующие системы защиты.

Тем не менее представленные средства на уровне прикладных протоколов взаимодействия работают только с копией трафика (KICS, PT ISIM), проводят анализ трафика только в режиме индикации, осуществляя неинтрузивное инспектирование (KICS, DATAPK), либо на данный момент не осуществляют обработку трафика по отдельно выбранному протоколу (например, поддержка Modbus TCP при построении сетей Ethernet в решении KICS и отсутствие поддержки Modbus RTU).

Как было ранее упомянуто, каждый коммуникационный протокол имеет свои достоинства и недостатки. Одним из самых распространенных коммуникационных протоколов является Modbus RTU. Данный протокол прост в реализации и отладке, поддерживается большинством современных ПЛК, позволяет составлять объемные сети и обеспечивает сравнительно надежную и достоверную передачу информации [6]. К недостаткам протокола можно отнести ограниченность принципом «ведущий-ведомый» с единственным ведущим устройством, а также отсутствие шифрования.

Последний из упомянутых недостатков представляет собой серьезную уязвимость протокола при удаленной передаче информации, заключающейся в возможной передаче конфиденциальной информации в незашифрованном виде, что при получении доступа к удаленной точке нарушителем позволит выполнить команды запуска, остановки, выгрузки и загрузки данных на устройстве [7]. Данная уязвимость характеризуется как критически опасная и требует отдельных организационных мер для устранения на этапе конфигурирования ЛСУ.

Таким образом, применение протокола Modbus RTU может нести определенные риски, связанные с утратой целостности, конфиденциальности и доступности информации.

В связи с наличием данной уязвимости была разработана концепция устройства, предназначенного для непосредственного анализа сетевого трафика Modbus RTU, обнаружения и предотвращения вредоносной активности в технологических сетях.

Предполагаемый алгоритм работы устройства следующий: устройство на базе микроконтроллера получает фрейм Modbus RTU по одному физическому интерфейсу и анализирует его в соответствии с допустимым содержанием полей. В случае соответствия запроса всем определенным в проекте АСУ критериям запрос передается на целевой ПЛК через второй физический интерфейс. Далее по второму физическому интерфейсу устройство получает ответ ПЛК, анализирует его на наличие аномалий и в случае удачной проверки отправляет по исходному адресу запроса через первый интерфейс. В случае обнаружения аномалий ответ ПЛК блокируется устройством и не отправляется по адресу запроса.

Если фрейм не соответствует установленным значениям полей, то запрос блокируется устройством, а по третьему физическому интерфейсу, соединенному с терминалом специалиста по информационной безопасности, отправляется сообщение о соответствующем нарушении в трафике сети.

Для наглядности принципиальная схема расположения устройства в сети приведена на рис. 2.

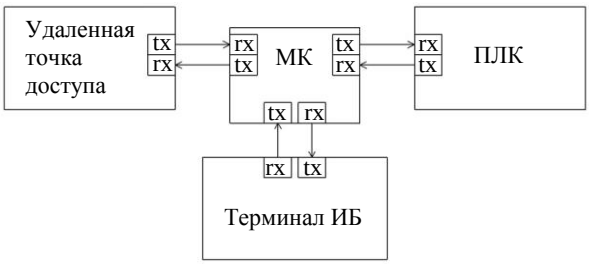


Рис. 2. Принципиальная структурная схема сегмента технологической сети с установленным устройством информационной безопасности на базе микроконтроллера (МК)

Таким образом, устройство предложенной концепции осуществляет прямой анализ сетевого трафика Modbus RTU и представляет собой программно-аппаратный комплекс (ПАК), наличие которого в составе сети не может быть определено злоумышленником программно.

Ввиду установки устройства в прямую технологическую линию, устройство позволяет осуществлять прямой контроль и анализ сетевого трафика, что может быть использовано для контроля ввода в машинные носители информации сети и контроллерное оборудование и информационную сеть в общем. Подобные требования зафиксированы в приказе ФСТЭК № 31 от 14 марта 2014 г. для АСУ класса защиты К1 [8], следовательно, предлагаемое устройство может быть использовано в совокупности с существующими ПАК информационной безопасности для построения защищенный АСУ класса К1.

Для реализации устройства был принят микроконтроллер семейства AVR ATmega162. Данный МК оснащен двумя аппаратными универсальными асинхронными приемопередатчиками (УАПП, UART). Для реализации конфигурации устройства, представленного на рис. 2, необходимо три UART, следовательно, дополнительный интерфейс UART должен быть реализован программно, однако в дальнейшем возможен переход на более продвинутые модели МК с расширенными аппаратными возможностями, например, увеличенным объемом памяти и большим количеством аппаратных UART.

На базе интерфейса UART реализуется протокол Modbus RTU. Фрейм протокола имеет структуру, приведенную на рис. 3 [9].

Адрес ведомого устройства	Функция и данные сообщения	CRC16

Рис. 3. Структура фрейма Modbus RTU

Максимальная длина фрейма составляет 255 байт – 1 байт – на адрес ведомого устройства, 2 байта – на

контрольную сумму CRC16 и до 253 байт – на набор данных сообщения. Анализу подлежит каждое из выделенных полей, при несоответствии которых установленным в проекте следует заблокировать запрос и передать на терминал специалиста по информационной безопасности соответствующее сообщение. Для разработки устройства на данном этапе приняты сообщения следующего формата:

1. «Запрос к незарегистрированному устройству!» («Request to an unregistered device!»).

2. «Код функции не зарегистрирован!» («Function code is not registered!»).

3. «Запись незарегистрированного регистра!» («Writing an unregistered register!»).

4. «Значение записи вне диапазона!» («The value of the writing is out of range!»).

5. «Пакет поврежден (CRC)» («Frame is damaged (CRC)»).

Приведенные сообщения передаются при обнаружении соответствующих нарушений в трафике сети. Следует также уточнить, что повреждение пакета скорее указывает на ошибки при передаче фрейма и наличие аномалий в сети, чем на наличие направленной вредоносной активности.

Для разработки и отладки устройства используются такие программные средства, как среда разработки Microchip Studio, симулятор мастер-запросов Modbus Poll, графическая оболочка программатора SinaProg и графическая оболочка для работы с COM-портом, терминал t1.9b.

Для симуляции работы устройства создан тестовый проект, обрабатывающий функции записи состояния дискретного выхода. В проект введена запись трех последовательных выходов с состояниями «0» и «1», т.е. реализована обработка 6 команд и при получении первым UART микроконтроллера фрейма, не соответствующего одной из 6 зарегистрированных команд, по другому UART (и по другому COM-порту соответственно) отправляется сообщение о нарушении, считываемое с терминала.

Тестирование работы алгоритма проводится при помощи аппаратных возможностей микроконтроллера – порты ввода-вывода симулируют дискретные выводы, т.е. запись состояний «1» и «0» порта микроконтроллера соответствует высокому и низкому логическому уровню. К портам микроконтроллера подведены светодиоды, следовательно, об успешной записи состояния симулированного дискретного выхода в состояние «1» свидетельствует включенный светодиод, в состояние «0» – погашенный.

Для прошивки микроконтроллера и сборки системы для тестирования использована отладочная плата Pinboard RII.

Ниже приведены промежуточные результаты тестирования разработки и тестирования алгоритма (рис. 4–7):

При попытке записать состояние нескольких дискретных выводов за одну команду (команда записи нескольких выходов не зарегистрирована) терминал, подключенный к другому COM-порту отладочной платы, получает сообщение о несоответствии.

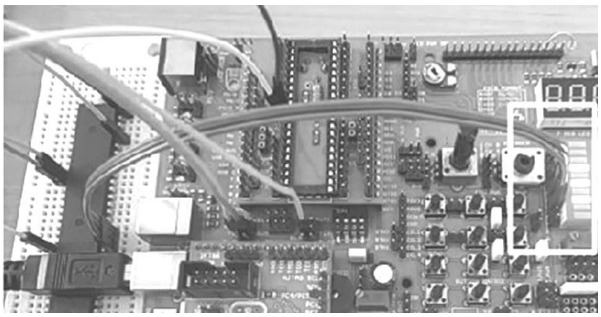


Рис. 4. «Дискретные выходы» микроконтроллера в состоянии «1»

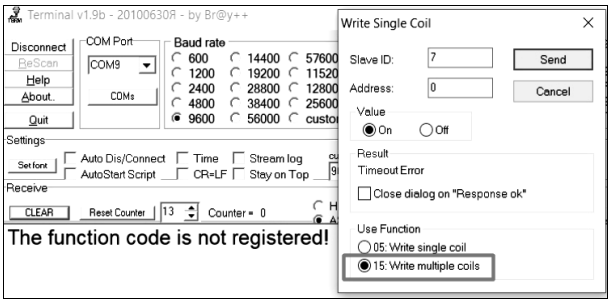


Рис. 5. Сообщение о незарегистрированной команде в полученном фрейме

При попытке записать состояние дискретного выхода, который не зарегистрирован в проекте, терминал получает иное сообщение.

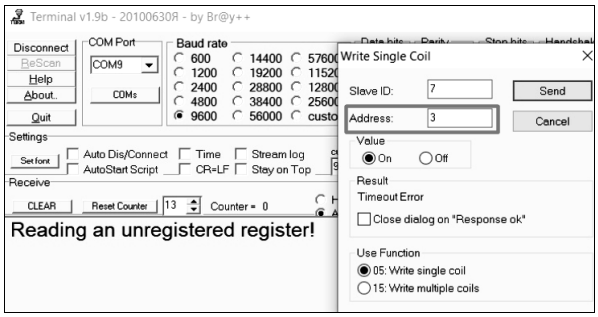


Рис. 6. Сообщение о попытке записи состояния незарегистрированного в проекте выхода

При разработке тестового проекта микроконтроллеру был присвоен конкретный адрес ведомого устройства. Соответствующее сообщение посылается на терминал при попытке обратиться к незарегистрированному устройству.

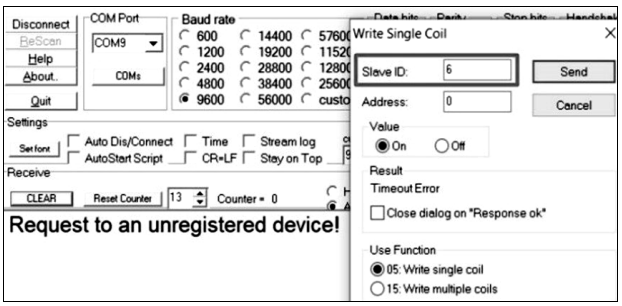


Рис. 7. Обращение к незарегистрированному устройству

Подобные ситуации могут возникать, если удаленный доступ возможен к нескольким ПЛК и злоумышленнику недостаточно хорошо известна конфигурация ЛСУ. Данную возможную уязвимость также необходимо контролировать.

Таким образом, промежуточными результатами работы являются разработанная концепция устройства, первичный алгоритм анализа трафика Modbus RTU. Протестированы получение и проверка фрейма Modbus RTU с соответствующей динамикой устройства в зависимости от содержания фрейма – запись состояния дискретного выхода, отправка сообщения на терминал.

Литература

1. Беспроводные локальные сети // Энциклопедия АСУ ТП – 2014. – URL: https://www.bookasutp.ru/Chapter2_11_1.aspx (дата обращения: 06.05.2022).

2. МКОН преобразователь протокола Modbus // Оборудование для автоматизации ОВЕН – 2022. – URL: https://owen.ru/product/mkon (дата обращения: 27.04.2022).

3. Статистика // Kaspersky ICS cert – 2022. – URL: https://ics-cert.kaspersky.ru/statistics/ (дата обращения: 26.04.2022).

4. Ульянов Н. Интернет угрожает // ЭкспертРУ. – 2021. – URL: https://expert.ru/expert/2021/40/internetugrozhayet/ (дата обращения: 27.04.2022).

5. Небайкин М. Кибербезопасность АСУ ТП. Обзор специализированных наложенных средств защиты. – 2022. – URL: https://www.anti-malware.ru/analytics/Market_Analysis/ICS-security-review#part61 (дата обращения: 27.04.2022).

6. Промышленный протокол ModBus // Lazy Smart. – 2016. – URL: http://lazysmart.ru/osnovy-avtomatiki/promyshlenny-j-protokol-modbus/ (дата обращения: 06.05.2022).

7. BDU:2019-02550: Уязвимость протокола Modbus микропрограммного обеспечения программируемых логических контроллеров, позволяющая нарушителю выполнить команды запуска, остановки, выгрузки и загрузки данных на устройстве // Банк данных угроз безопасности информации – 2017. – URL: https://bdu.fstec.ru/vul/2019-02550 (дата обращения: 05.05.2022).

8. Приказ ФСТЭК России от 14 марта 2014 г. № 31 // ФСТЭК России – 2021. – URL: https://fstec.ru/normotvorcheskaya/akty/53-prikazy/868-prikaz-fstek-rossii-ot-14-marta-2014-g-n-31 (дата обращения: 27.04.2022).

9. Как общаются машины: протокол Modbus // Хабр. – 2019. – URL: https://habr.com/ru/company/advantech/blog/450234/ (дата обращения: 06.05.2022).

Калинкин Ян Васильевич
Студент ОАР ИШИТР НИ ТПУ
Эл. почта: yvk36@tpu.ru

Тутов Иван Андреевич
Студент отд. автоматизации и робототехники (ОАР),
ст. преп. ОАР ИШИТР НИ ТПУ
Эл. почта: ivantutov@tpu.ru

УДК 004.065.53

А.А. Чумаков, Р.Я. Панцыр, Н.О. Басков

Методы и средства защиты устройств средств вычислительной техники от несанкционированного доступа, реализуемого при помощи уязвимостей Intel Management Engine

Дано описание подсистемы Intel Management Engine, а также её место в средствах вычислительной техники. На основе анализа функций и размещения компонентов подсистемы проведён анализ основных возможных уязвимостей, возникающих при использовании средств вычислительной техники с чипсетами, в которые интегрирована подсистема Intel Management Engine. На основе анализа подготовлено заключение о том, что использование подсистемы ставит под угрозу конфиденциальность, целостность и доступность обрабатываемых в средствах вычислительной техники данных, а также позволит осуществить несанкционированный доступ к ним. Были предложены методы и средства, позволяющие обезопасить средство вычислительной техники от угрозы несанкционированного доступа, при эксплуатации уязвимостей Intel Management Engine.

Ключевые слова: Intel Management Engine, СВТ, НСД, угрозы, BIOS/UEFI, Intel, доверенная программно-аппаратная среда.

Intel Management Engine (IME) – это подсистема, представляющая собой сопроцессор, реализованный в виде микроконтроллера, встроенного в кристалл чипсета (PCH), который обеспечивает выполнение функций для управления корпоративными активами (модули ядра AMT), технологии защиты от кражи (Anti-Theft), функции процессора, не позволяющие запускать прошивки, не выпущенные производителем системы (Boot Guard), технологии доверия к платформе (PTT), предоставление возможности создания виртуального адресного пространства, защищенного от чтения и записи извне, включая ядро операционной системы (SGX EPID) и службы защищённого мультимедийного тракта [1].

Сопроцессор использует для функционирования флеш-память BIOS/UEFI. Там же находится часть конфигурации IME [2-3].

Подсистема встраивается с 2008 г. во все чипсеты Intel. Для серверов данная технология имеет название Intel SPS, а для мобильных – Intel TXE [4]. Также подобная технология встраивается с 2013 г. во все процессоры AMD (AMD Secure Techonology) [1]. В открытых источниках имеются данные о встраивании схожей с IME технологии в процессоры ARM (VideoCore) [5].

Начиная с 11 версии IME в качестве микроконтроллера используется 32-битный x86-совместимый процессор на технологии Intel Quark. Среди особенностей можно выделить наличие встроенной операционной системы (ОС) MINIX 3, выделенного RAM (~32 Мб), специальной выделенной шины взаимодействия, региона SPI флэш с прошивкой IME [6, 7].

Подсистема IME способна функционировать при выключенном средстве вычислительной техники (СВТ). Достаточно наличия дежурного напряжения на материнской плате или питания от встроенного аккумулятора. Это объясняется тем, что чипсет имеет постоянное подключение к источнику питания.

По причине отсутствия задокументированных принципов работы подсистемы и невозможности проанализировать исходный код IME (применяется обфускация) невозможно однозначно утверждать о безопасности СВТ с данной подсистемой. Пристальное внимание компании Intel к вопросу сокрытия от потребителя информации, содержащейся в прошивке IME, даёт основания полагать, что Intel намерено пытается скрыть данные прошивки.

Наличие подсистемы IME ставит под угрозу несанкционированного доступа (НСД) к СВТ, так как нет оснований полагать, что злоумышленник, выступающий доверенным лицом Intel, не воспользуется шансом эксплуатирования предполагаемых скрытых недекларированных возможностей IME.

Нет официального документированного способа отключения IME, поэтому в данной работе помимо определения угроз произведёны поиск, анализ и подготовка методов и средств защиты СВТ от возможных угроз НСД при помощи IME.

Определение и анализ угроз НСД

На основе информации, имеющейся в открытых источниках, построена примерная схема взаимодействия IME с устройствами СВТ (рис. 1) [1, 3].

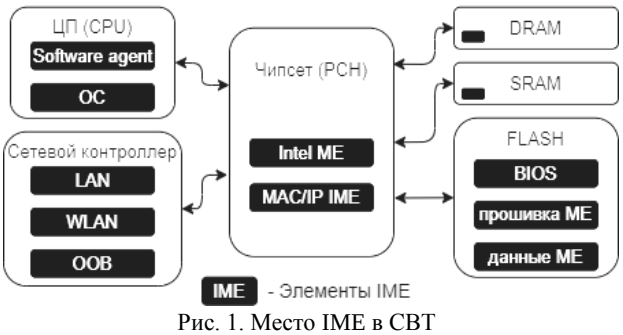


Рис. 1. Место IME в СВТ

На основе анализа схемы и найденной информации можно сделать вывод о том, что подсистема имеет полномочия, превышающие уровень ядра ОС, возможности доступа к специально зарезервированному и защищённому для доступа блоку оперативной памяти (DRAM) СВТ, а также некоторому объё-

му кеш-памяти центрального процессора (ЦП) [8]. По заверению компании Intel сделано это для более быстрой и эффективной работы [8].

Подсистемой могут быть использованы команды OOB (Out-of-band), которые позволяют осуществлять управление СВТ по каналу сетевой передачи данных. По заявлению Intel необходимо это для целей снижения потребления электроэнергии в момент, когда ОС не запущена или находится в режиме сна [8].

Из рис. 1 видно, что у IME имеется свой MAC и IP адрес, который используется для прямого доступа к сетевому контроллеру. Ввиду того, что IME имеет приоритет исполнения команд выше, чем ядро ОС, то, следовательно, подсистема игнорирует любые ограничения, которые могут быть установлены файерволом ОС. Пакеты переадресуются напрямую с сетевого контроллера на IME.

Имея доступ к FLASH памяти, используемой вместе с BIOS/UEFI у IME, появляется возможность контролировать ход запуска драйверов и службы, а также управлять процессом запуска ОС. Исходя из данных, которые можно найти в спецификациях Intel на прошивки FLASH памяти, можно сделать вывод, что с поколения Intel Apollo Lake была внедрена возможность установки сторонней прошивки BIOS/UEFI в обход базовым механизмам установки путём внедрения компанией Intel возможности IME перезаписи SPI-региона в обход запрета на доступ со стороны хоста [4].

Таким образом, можно сделать вывод, что наличие подсистемы IME ставит под угрозу безопасность всех обрабатываемых СВТ данных. Скрытое выделение пространства в SRAM, DRAM, FLASH, работа в невидимом для ОС режиме, возможность скрытого взаимодействия по каналам связи и отсутствие возможности анализа ПО, исполняемого на IME, позволяет сделать вывод, что данная подсистема может эксплуатироваться злоумышленником с целью осуществления НСД к СВТ.

Первыми, кто обратили внимание на возможность внедрения компанией Intel бэкдора, был эксперт по информационной безопасности Damier Zammit и Фонд электронных рубежей [9, 10]. Помимо сказанного, эксперты сообщают, что Intel, начиная с ЦП серии Core2, внедрила возможность в прошивку ME, которая позволяет остановить запуск СВТ при попытках пользователя отключить IME [9]. Это возможно, так как ME, будучи интегрированным в чипсет, обрабатывает большую часть данных, передаваемых между процессором и внешними устройствами, и, таким образом, имеет доступ к большинству данных СВТ [11]. Следовательно, ME абсолютно независим от ЦП [12].

Говорится также, что прошивка IME подписана Intel при помощи ключа RSA 2048, что предотвращает возможность стороннего вмешательства лиц, не являющихся партнёрами компании Intel [11].

Ввиду наличия неопровержимых фактов, наличия бэкдора были направлены запросы компании Intel с целью уточнения факта наличия интегрированных скрытых возможностей IME, на что был получен ответ от технического директора Intel Стива Гробмана: «...Корпорация Intel не использует лазейки в своих продуктах, и наши продукты не предоставляют корпорации Intel контроль или доступ к вычислительным системам без явного разрешения конечного пользователя... Однако, как мы все с болью осознаем, современный ландшафт угроз каждый год создает бесчисленные проблемы безопасности, нацеленные на системы в самых разных областях...» [13].

Intel в очередной раз отказывается признавать возможность присутствия уязвимости до тех пор, пока она не будет эксплуатирована, как, например, описанный случай признания и исправления уязвимости, связанной с возможностью удаленного доступа к Intel Trusted Execution Engine [14].

Чтобы окончательно убедится в том, что в IME присутствует бэкдор, обратимся к данным, которые были опубликованы экспертами Positive Technologies. В ходе анализа и поиска возможностей изменения прошивки IME с целью деактивации ME специалисты обнаружили в коде прошивки скрытый бит, который при изменении значения с «0» на «1» отключает ME после загрузки ЦП и первоначальной инициализации в момент старта на этапе запуска драйверов и утилит UEFI. Бит помечен как «reserve_hap» с комментарием «включение платформы High Assurance Platform (HAP)» [11, 15].

HAP – это программа Агентства национальной безопасности США (АНБ), описывающая ряд правил для запуска безопасных вычислительных платформ. Следовательно, можно предположить, что Intel добавила бит отключения ME по указанию АНБ, которому нужен был метод отключения ME в качестве меры безопасности для компьютеров, работающих в высокочувствительных средах. ME или любые уязвимости в его прошивке могли привести к утечке особо опасной информации, поэтому АНБ требовалось исключить возможные риски.

В ответ на найденную особенность прошивки Intel косвенно подтвердила, что они внедрили данную функцию из-за требований АНБ: «В ответ на запросы клиентов с особыми требованиями мы иногда изучаем возможность модификации или отключения определенных функций. В данном случае модификации были внесены по запросу производителей оборудования в поддержку оценки их заказчиками программы правительства США ″Платформа высокой надежности″» [15].

Таким образом, исходя из определения бэкдор (метод обхода обычной аутентификации в компьютерной системе, используемый для несанкционированного удаленного доступа к компьютеру) и основываясь на описанных выше угрозах размещения в СВТ IME, можно прийти к выводу, что Intel намеренно внедрила в свои решения программно-аппаратный инструмент, позволяющий осуществить негласный перехват информации с СВТ, а также

осуществить вывод его из строя. Таким образом, под угрозу ставит целость, конфиденциальность и доступность обрабатываемых на СВТ данных.

Для того чтобы определить, является ли СВТ подверженным одной из имеющихся на сегодняшний день уязвимостей, компанией Intel была представлена специальная утилита [16].

Таким образом, можно сделать вывод, что с помощью IME возможно осуществление НСД к СВТ. Далее в статье были предложены методы и средства защиты от угроз IME.

Методы и средства защиты

С момента обнаружения уязвимости основным способом защиты было удаление из FLASH BIOS прошивки ME до момента возможности обеспечения функционирования основных систем СВТ [12].

Однако, как говорилось выше, ввиду полной автономности IME и присутствия возможности удалённого изменения своей собственной прошивки, становится невозможным утверждать, что данный способ является безопасным.

Еще одним из способов является изменение значений скрытой функции «reserve_hap» в прошивке UEFI. Однако данные действия не могут гарантировать защиты и приводят к тому, что вызывают модификацию ПО.

Наиболее актуальным способом защиты, с точки зрения авторов, является использование доверенной программно-аппаратной среды. Одним из примеров может являться использование отечественных ЦП и материнских плат, в которых исключено наличие недекларированных возможностей, приводящих к утечке критической информации.

Для доверенных систем необходимо наличие полного комплекта конструкторской документации на аппаратные решения, наличие заключений об исследовании программных компонентов на предмет отсутствия в нём недекларированных возможностей, применение сертифицированных по требованиям безопасности информации ПО и программно-аппаратных модулей, обеспечение сертифицированных решений, обеспечивающих контроль от НСД и защиту от вирусных и вредоносных атак. Также необходимо использование решений, позволяющих реализовать полностью изолированную и замкнутую программную среду.

Стоит учитывать, что использование отечественных программно-аппаратных платформ, которыми являются «Эльбрус» и «Байкал», не является наиболее эффективным по причине того, что невозможно осуществить быстрый и крупномасштабный переход с имеющихся платформ INTEL и AMD как в серверном, так и потребительском сегменте.

Наиболее грамотным решением можно предложить использование аппаратно-программных модулей доверенной загрузки (АПМДЗ) совместно с криптографическими сетевыми интерфейсными адаптерами (КСИА), ввиду того, что АПМДЗ перехватывает управление UEFI на самом старте компьютера и следит за тем, чтобы не нарушалась целостность программно-аппаратной среды, а также осуществлять защиту от преднамеренного НСД к СВТ.

Использование КСИА позволит шифровать весь передаваемый трафик по сети, что сделает абсолютно невозможным перехват конфиденциальной информации.

Использование комбинированной защиты, сочетающей в себе использование КСИА и АПМДЗ, сертифицированных по требованиям безопасности информации, позволит обеспечить защиту от уязвимостей IME и создать доверенную программно-аппаратную среду.

Таким образом, анализируя сказанное в этом разделе, можно сказать, что единственным надёжным решением, которое может позволить защититься от НСД к СВТ, реализуемого при помощи IME, является внедрение доверенной программно-аппаратной среды.

Выводы

В ходе проведенного исследования было установлено, что IME имеет приоритет выше ядра ОС, а также способна:

– эксплуатировать зарезервированный скрытый объём SRAM и DRAM;

– передавать данные по сети в обход файервола и имеет свой собственный IP и MAC;

– скрытно обновлять свою прошивку;

– прекратить возможность запуска СВТ ввиду контроля над процессом;

– получать и управлять всеми данными, циркулирующими в СВТ ввиду связи через чипсет со всеми критическими узлами;

– функционировать и передавать данные в моменты, когда СВТ выключен, но имеет подключение сети питания

Из этого можно сделать вывод, что внедрённая компанией Intel программно-аппаратная платформа однозначно может быть использована для осуществления НСД к СВТ и данным, обрабатываемым в ней.

Таким образом, было обнаружено, что наличие IME даёт для злоумышленников возможность получения полного контроля над данными, обрабатываемыми в СВТ. Ввиду отсутствия возможности анализа кода ME нет гарантий, что Intel не поставляет со своими продуктами аппаратно-программные закладки, внедряемые АНБ, с которым Intel тесно сотрудничает.

Для обеспечения защиты СВТ от НСД, реализуемого через уязвимости IME, были рассмотрены соответствующие методы и средства. Так как ни один из имеющихся в открытом доступе методов не позволял гарантировать защиты, было предложено использование доверенной программно-аппаратной среды.

Таким образом, был предложен способ защиты, который комбинировал в себе использование КСИА и АПМДЗ, которые бы позволили обеспечить создание доверенной программно-аппаратной среды и обезопасить СВТ от возможности НСД путём эксплуатации уязвимостей IME.

Литература

1. Management Engine (ME) – Intel // WikiChip Chips & Semi [Электронный ресурс]. – URL: https://en.wikichip.org/wiki/intel/management_engine (дата обращения: 20.04.2022).

2. Обновить безопасность UEFI можно будет через Windows 10 // Windows school [Электронный ресурс]. – URL: https://windows-school.ru/blog/obnovit_bezopasnost_uefi/2019-01-29-284 (дата обращения: 20.04.2022).

3. Intel ME. Как избежать восстания машин? // Хабр [Электронный ресурс]. – URL: https://habr.com/ru/company/dsec/blog/282546/ (дата обращения: 20.04.2022).

4. Intel ME Manufacturing Mode – скрытая угроза, или что стоит за уязвимостью CVE-2018-4251 в MacBook // Хабр [Электронный ресурс]. – URL: https://habr.com/ru/company/pt/blog/425105/ (дата обращения: 20.04.2022).

5. Is there a management engine in the ARM CPUs? // Raspberry Pi [Электронный ресурс]. – URL: https://forums.raspberrypi.com/viewtopic.php?t=151542 (дата обращения: 20.04.2022).

6. Оголюк А.А. Анализ безопасности удаленного доступа средствами Intel management engine / А.А. Оголюк, А.В. Шабалин // Изв. вузов. Приборостроение. . – 2018. – Т. 61, № 1. – С. 41–46.

7. Intel Management Engine // Wikipedia. Свободная энциклопедия [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Intel_Management_Engine (дата обращения: 20.04.2022).

8. What is Intel® Management Engine? // Intel [Электронный ресурс]. – URL: https://www.intel.com/content/www/us/en/support/articles/000008927/software/chipset-software.html (дата обращения: 21.04.2022).

9. Intel x86 CPUs Come with a Secret Backdoor That Nobody Can Touch or Disable // Softpedia [Электронный ресурс]. – URL: https://news.softpedia.com/news/intel-x86-cpus-come-with-a-secret-backdoor-that-nobody-can-touch-or-disable-505347.shtml (дата обращения: 21.04.2022).

10. Intel's Management Engine is a security hazard, and users need a way to disable it // Electronic Frontier Foundation [Электронный ресурс]. – URL: https://www.eff.org/deeplinks/2017/05/intels-management-engine-security-hazard-and-users-need-way-disable-it (дата обращения: 21.04.2022).

11. Intel ME controller chip has secret kill switch // The Register [Электронный ресурс]. – URL: https://www.theregister.com/2017/08/29/intel_management_engine_can_be_disabled/ (дата обращения: 21.04.2022).

12. Management Engine и слежка за пользователем // Pikabu [Электронный ресурс]. – URL: https://pikabu.ru/story/management_engine_i_slezhka_za_polzovatelem_8164699 (дата обращения: 21.04.2022).

13. Is the Intel Management Engine a backdoor? // Techrepublic [Электронный ресурс]. – URL: https://www.techrepublic.com/article/is-the-intel-management-engine-a-backdoor/ (дата обращения: 25.04.2022).

14. Intel Patches Major Flaws in the Intel Management Engine // Extremetech [Электронный ресурс]. – URL: https://www.extremetech.com/computing/259426-intel-patches-major-flaws-intel-management-engine (дата обращения: 21.04.2022).

15. Researchers Find a Way to Disable Much-Hated Intel ME Component Courtesy of the NSA // Bleepingcomputer [Электронный ресурс]. – URL: https://www.bleepingcomputer.com/news/hardware/researchers-find-a-way-to-disable-much-hated-intel-me-component-courtesy-of-the-nsa/ (дата обращения: 21.04.2022).

16. Intel обнаружила критические уязвимости в Management Engine, ставящие под угрозу миллионы машин // Хакер.ру [Электронный ресурс]. – URL: https://xakep.ru/2017/11/21/intel-me-flaws/ (дата обращения: 21.04.2022).

Чумаков Андрей Александрович
Студент каф. информационной безопасности НИУ «МИЭТ»
Эл. почта: mr.jordan99@yandex.ru

Панцыр Родион Яковлевич
Канд. техн. наук, доцент каф. информационной безопасности НИУ «МИЭТ»
Эл. почта: rodionpan@yandex.ru

Басков Никита Олегович
Студент каф. информационной безопасности НИУ «МИЭТ»
Эл. почта: nikita.baskov@yandex.ru

УДК 001.891.572

А.Д. Калякин, А.С. Коваленко, В.С. Репкин, А.К. Новохрестов

Разработка сценария кибератаки на веб-портал предприятия

Рассмотрено создание сценария кибератаки, направленной на веб-портал предприятия. Данная кибератака содержит в себе следующие уязвимости: CWE-798 (использование жестко закодированных учетных данных); наличие PowerShell профиля, выполняемого администратором; отсутствие аудита изменений кода веб-портала. Впоследствии созданная атака была формализована с помощью следующих методов моделирования: графовая модель с использованием языка метаатак (Meta Attack Language), граф в виде «Дерева атак» и граф в виде Cyber Kill Chain.

Ключевые слова: информационная безопасность, сценарий кибератаки, MITRE ATT&CK, формализация кибератак, метаграф, дерево атак, Cyber Kill Chain.

В условиях глобальной информатизации всех сфер жизнедеятельности людей случаи кибератак на всевозможные системы и объекты критической информационной инфраструктуры (далее – КИИ) принимают огромный оборот [1]. В данной ситуации как никогда важно наличие грамотных специалистов по информационной безопасности (далее – ИБ). Для обучения таких специалистов уже сейчас существуют так называемые киберполигоны, основной идеей которых является развертывание на их базе виртуальной сети предприятия или объекта КИИ, на которые впоследствии происходит заранее подготовленная в виде скрипта кибератака. Основной проблемой всех киберполигонов на данный момент является отсутствие разнообразных сценариев атак для их последующей реализации в рамках киберполигона. Поэтому основной целью данной работы является создание сценария кибератаки. В качестве объекта атаки был выбран веб-портал предприятия, а сам сценарий создавался с помощью матрицы уязвимостей и угроз MITRE ATT&CK. Также разработанный сценарий был формально описан с помощью нескольких моделей.

В широком смысле сценарий кибератаки – это модель, отражающая последовательность действий злоумышленника по проникновению в систему, закреплении в ней и извлечение выгоды путем уничтожения данных, их копирования и т.д. Если же говорить о сценарии атаки в киберполигоне – это имитация реальной компьютерной атаки на инфраструктуру, проходящая посредством выполнения определенных скриптов, которые имитируют действия злоумышленника в киберполигоне.

Сами сценарии внутри киберполигона нужны для обучения специалистов информационной безопасности – путем прохождения сценариев они постоянно открывают для себя новые различные уязвимости в системе и сами находят пути их закрытия. В данном случае необходимо наличие разнообразия сценариев, содержащих большое количество непохожих друг на друга уязвимостей, чтобы специалисты ИБ развивались в различных направлениях и продолжали получать опыт в данной области.

Подходы к составлению сценариев

Так как сценарий кибератаки состоит из последовательных шагов злоумышленника, то необходимо каждый отдельный шаг рассматривать как реализацию нарушителем определенной уязвимости. Тогда для составления корректного сценария кибератаки необходимы источники уязвимостей, которыми могут послужить различные ресурсы, например специализированные банки уязвимостей, или же уязвимости можно вычленять из реальных кибератак, которые описаны на различных тематических сайтах.

К первому способу можно отнести матрицу MITRE ATT&CK, которая, в свою очередь, содержит техники, тактики, процедуры и приемы возможных атак, которые злоумышленники используют для проникновения в сеть, компрометации ИТ-систем и перемещения внутри среды без обнаружения. Также в данной матрице для каждой уязвимости есть дополнительные сведения: способы обнаружения и способы противодействии им. Благодарю такому объему информации для каждой уязвимости, возможно полноценное описание атаки злоумышленника с последующими способами определения и противодействия его шагам.

С другой же стороны, возможен разбор реально случившихся атак на компании или предприятия. Так как данная атака действительно производилась, это говорит о том, что уязвимости, используемые злоумышленниками, являются действенными, а также гарантирует, что построенный таким способом сценарий атаки лишен всех возможных изъянов и несостыковок, а также то, что он будет полноценно функционировать при его добавлении в киберполигон.

В качестве примера можно взять кибер-ограбление Cayman National Bank в 2016 г., которое было чревато для банка потерей сотни тысяч фунтов стерлингов его клиентов. Можно проанализировать данный случай и разбить атаку на конкретные шаги злоумышленников, в которых они реализовывали ту или иную уязвимость. Наличие последовательности реализованных угроз упростит в будущем процесс формального описания сценария атаки. Во-первых, злоумышленники получили доступ к офисному компьютеру двумя способами. Одна группа выполнила атаку на внешние удаленные службы, где был обнаружен эксплойт «Sonicwall SSL/VPN», что позволило им использовать удаленные службы для атаки на зараженный компьютер для проникновения в офисную зону. Другая группа эксплуатировала фишинго-

вое вложение «spear», которое было использовано пользователем, что дало доступ в офисную зону нарушителям. Далее манипулирование учетными записями позволило злоумышленникам следить за ходом расследования и оставаться в сети, а использование PowerShell позволяло им проводить манипуляцию передаваемыми данными в реальном времени [2].

Так как на основе матрицы MITRE ATT&CK возможно более глубокое изучение поведения злоумышленника в той или иной ситуации, данный способ предпочтительнее для использования и создания на его основе сценариев кибератак.

Сценарий кибератаки на веб-портал предприятия

Далее будет рассмотрен разработанный сценарий, в основе которого лежит атака на веб-портал предприятия, а также будут приведены все формализованные методы его описания.

Легенда:

Внешний злоумышленник находит в интернете веб-портал предприятия и решает провести атаку на него с целью перехвата информации работы пользователей на нем в реальном времени. После получения законных учетных данных грубой силой (составление карты внутренней сети) нарушитель заходит в систему, а далее повышает свои привилегии, благодаря чему получает возможность внесения изменений в код веб-портала.

Квалификация нарушителя высокая. Он умеет использовать инструментарий для проведения атак, а также знает техники постэксплуатации.

Шаги атаки:

1. Злоумышленник использует инструмент под названием «GET2 Penetrator», инструмент сканирования грубой силы, который ищет удаленный вход в систему и жестко закодированные учетные данные, а также бесплатный инструмент «EssentialNet», который служит для отображения сети и создания списка целей. Злоумышленник использует действительные учетные данные для получения доступа во внутреннюю сеть предприятия [3].

2. Злоумышленник сохраняет свое присутствие в системе и повышает привилегии, выполняя вредоносный код, запускаемый профилями PowerShell (сценарий, который запускается при запуске PowerShell и может использоваться в качестве сценария входа в систему для настройки пользовательских сред) учетной записью с более высокими привилегиями, например администратором домена [4].

3. Злоумышленник устанавливает код на веб-портале предприятия для захвата и передачи данных работы пользователей со скомпрометированной веб-страницы [5].

По ходу всех шагов атаки злоумышленника использовались следующие уязвимости системы предприятия:

1. CWE-798. Использование жестко закодированных учетных данных. Программное обеспечение содержит жестко закодированные учетные данные, такие как пароль или криптографический ключ, которые оно использует для собственной входящей аутентификации, исходящей связи с внешними компонентами или шифрования внутренних данных [6];

2. Наличие PowerShell профиля, выполняемого администратором;

3. Отсутствие аудита изменений кода веб-портала, при его изменении со стороны легальных учетных записей администратора или пользователя в системе.

Представление разработанного сценария кибератаки в формальном виде

Не стоит забывать и о формализованном описании сценариев, т.е. представлении их в наиболее удобном и конкретном виде, далее пригодном для перенесения сценария в киберполигон.

Формализация сценариев в данном случае направлена на решение двух задач: во-первых, в формализованном виде они без проблем масштабируются и изменяются, что позволяет упростить процесс составления новых сценариев; во-вторых, при их реализации в киберполигоне, тренирующиеся в нем специалисты постоянно получают огромное количество практического опыта в закрытии уязвимостей, в будущем, при их работе на предприятии, это помогает им устранять настоящие инциденты уже по пройденным во время тренировок шагам и быть готовыми к встрече с различными кибератаками.

Модели или объекты после их формализации должны удовлетворять определенным критериям, например: понятности, конкретности, полноте, однозначности. Разработанный здесь сценарий, конечно же, не является исключением. Для приведения сценариев к формализованному виду сначала нужно понять, какие виды описания и моделирования атак бывают.

Обычно классификация атак ориентирована на отдельные аспекты моделирования атак. Такими аспектами могут быть формализмы представления данных и знаний об атаках, уровни модели OSI (Open Systems Interconnection) или модели DoD (Department of Defense), типы атак, возможности масштабирования, способность учитывать динамические характеристики атак (например, время, параллельные процессы, цепочки взаимосвязанных инцидентов) и т.д.

Сначала можно обратиться к различным графовым представлениям: атрибутивные метаграфы рассмотрены в работе [7], также подобный способ задействуется и в других статьях [8, 9], которые использует данный метод. Не стоит забывать и о других моделях атак, таких как логические, имитационные, табличные и др. Также возможно использовать для данных целей и методы, основанные на дискретной математике.

В конечном итоге был сделан вывод, что наиболее удобным способом представления кибератак являются графические модели, приближенные к виду графов. Поэтому были доработаны некоторые уже существующие методы, что в конечном итоге

дало три способа представления сценариев атак: граф в виде Cyber Kill Chain (рис. 1), граф в виде «Дерева атак» (рис. 2) и графовая модель с использованием языка метаатак (Meta Attack Language) (рис. 3).

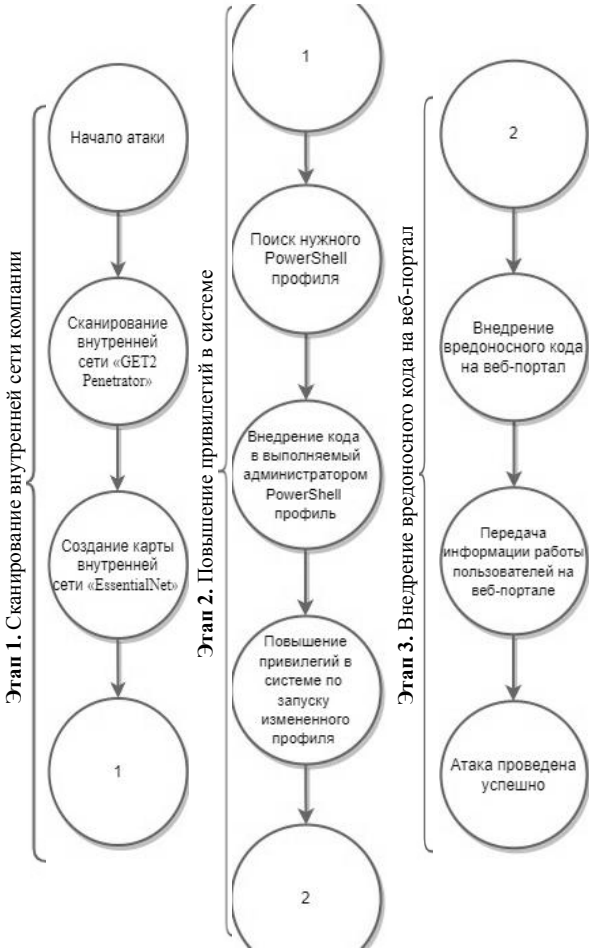


Рис. 1. Сценарий кибератаки, описанный с помощью графа Cyber Kill Chain

На рис. 1 продемонстрирован граф Cyber Kill Chain атаки нарушителя, который используется в основном для описания точечных шагов злоумышленника, разделяющихся на большие блоки основных шагов, описывающие в себе все необходимые условия для совершения данного действия.

В данном графе явно определяются 3 этапа шагов злоумышленника, а именно: сканирование внутренней сети компании, повышение привилегий в системе, внедрение вредоносного кода на веб-портал.

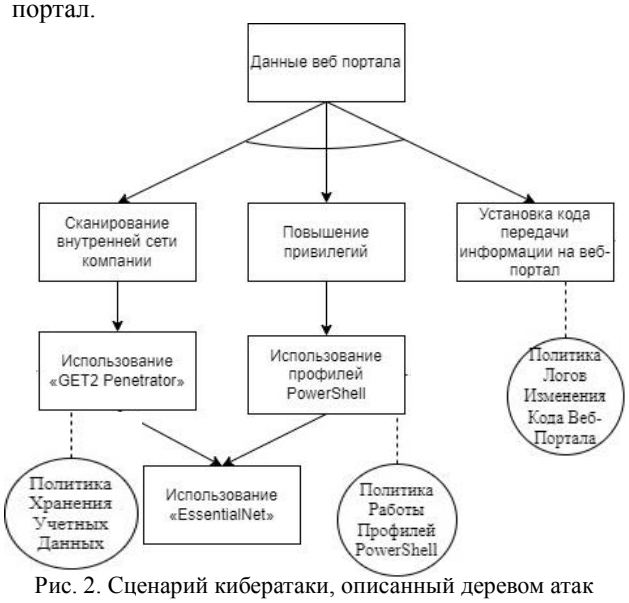


Рис. 2. Сценарий кибератаки, описанный деревом атак

На рис. 2 представлено дерево атак злоумышленника, с помощью которого становится возможно рассматривать параллельно все его действия, которые могут пересекаться в определенных схожих шагах. Так же, как и в метаграфе, в данном формальном представлении определены возможные противодействия для защиты системы от атак нарушителя.

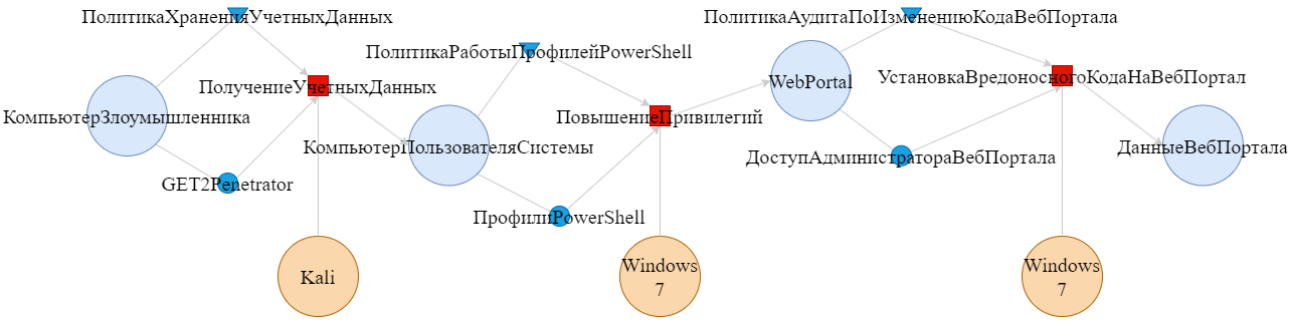


Рис. 3. Сценарий кибератаки, представленный в виде мета-графа

По данному дереву атак также определяются 3 этапа шагов злоумышленника, а именно: сканирование внутренней сети, повышение привилегий, установка кода передачи информации на веб-портал.

На рис. 3 показан метаграф кибератаки, на нем рассматриваются шаги злоумышленника, а также методы или программы, используемые им, и способы противодействия. Относительно каждого выполненного шага атаки нарушителя также известна среда работы, например операционная система.

Рассмотрев различные варианты формальных описаний разработанного сценария кибератаки, можно сделать вывод, что каждый из способов полностью отражает всю суть атаки, так как раскрывает основные шаги действий злоумышленника, изучив которые, можно сразу определить уровень нанесенного ущерба системе. Основываясь на этих данных, можно составить необходимые этапы противодействия всем проведенным атакам. У каждого способа описания есть свои особенности, благодаря которым

при необходимости можно рассматривать созданный сценарий с разных сторон.

Заключение

В результате данной работы был создан сценарий кибератаки на веб-портал предприятия, для его разработки использовалась матрица уязвимостей и угроз MITRE ATT&CK. Также разработанный в данной статье сценарий был формально описан с помощью трех различных моделей: графа «Cyber Kill Chain», «дерева атак» и графа метаатак. Формализация сценария кибератаки позволила конкретно и однозначно определить последовательные шаги злоумышленника, в каждом из которых он эксплуатировал ту или иную уязвимость системы. Проведя анализ и сравнение вышеприведенных методов формализации, можно сказать, что все они неплохо справились со своей задачей, и каждый из них может использоваться в зависимости от поставленной цели, также возможно их комбинирование в рамках одной задачи. Таким образом, можно сказать, что при наличии специалистов, способных реализовать каждую из уязвимостей на необходимом языке программирования, а также собрать подходящий шаблон сети предприятия, данный сценарий без каких-либо проблем интегрируется в киберполигон.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках базовой части государственного задания ТУСУРа на 2020–2022 гг. (проект № FEWM-2020-0037).

Литература

1. Киберпреступность в мире. Кибератаки–2022 [Электронный ресурс]. – Режим доступа: https://www.tadviser.ru/index.php/Статья:Киберпреступность_в_мире#.D0.9A.D0.B8.D0.B1.D0.B5.D1.80.D0.B0.D1.82.D0.B0.D0.BA.D0.B8, свободный (дата обращения: 17.09.2022).

2. How a bank got hacked [Электронный ресурс]. – Режим доступа: https://www.csoonline.com/article/3454443/how-a-bank-got-hacked-a-study-in-how-not-to-secure-your-networks.html, свободный (дата обращения: 17.09.2022).

3. FIN5 [Электронный ресурс]. – Режим доступа: https://attack.mitre.org/groups/G0053/, свободный (дата обращения: 17.09.2022).

4. Event Triggered Execution: PowerShell Profile [Электронный ресурс]. – Режим доступа: https://attack.mitre.org/groups/G0053/, свободный (дата обращения: 17.09.2022).

5. Input Capture: Web Portal Capture [Электронный ресурс]. – Режим доступа: https://attack.mitre.org/groups/G0053/, свободный (дата обращения: 17.09.2022).

6. CWE-798: Use of Hard-coded Credentials [Электронный ресурс]. – Режим доступа: https://cwe.mitre.org/data/definitions/798.html, свободный (дата обращения: 17.09.2022).

7. Model of Threats to Computer Network Software / A. Novokhrestov, A. Konev, A. Shelupanov // Symmetry. – 2019. – Vol. 11, Iss. 12. – 1506. – 13 p.

8. Модель угроз безопасности информации и ее носителей / А.К. Новохрестов, А.А. Конев, А.А. Шелупанов, Н.С. Егошин // Вестник Иркут. гос. техн. ун-та. – 2017. – Т. 21, № 12. – С. 93–104.

9. Computer network threat modelling / A. Novokhrestov, A. Konev, A. Shelupanov, A. Buymov // IOP Conf. Series: Journal of Physics: Conf. Series. – 2020. – Vol. 1488, 012002. – 6 p.

Калякин Антон Дмитриевич
Студент каф. безопасности информационных систем (БИС) ТУСУРа
Эл. почта: a.d.kalyakin@gmail.com

Коваленко Александр Сергеевич
Студент каф. БИС ТУСУРа
Эл. почта: a.s.kovalenko@mail.ru

Репкин Владимир Сергеевич
Студент каф. БИС ТУСУРа
Эл. почта: repkin_vova@mail.ru

Новохрестов Алексей Константинович
Доцент каф. комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС) ТУСУРа
ORCID: 0000-0002-0916-105X
Эл. почта: nak@fb.tusur.ru

УДК 001.891.573

Н.С. Егошин, Н.И. Сермавкин, В.С. Репкин, А.Д. Калякин

Моделирование сценариев кибератак на базе киберполигогна Ampire

Приведен обзор на состав сценариев киберполигона Ampire и разобрана структура построения сценариев от их разработчиков. Была изучена логическая схема сценария, интегрированная в систему Ampire для каждого из готовых сценариев. Описаны преимущества и недостатки логической схемы. Учитывая недостатки логической схемы, были разработаны и приведены альтернативы формального описания сценариев кибератак на основе таких методологий моделирования, как Meta-Attack Langue (MAL) и «Дерево атак». Созданные модели в будущем могут стать усовершенствованной альтернативой для формального описания кибератак при проведении киберучений на базе киберполигона Ampire.

Ключевые слова: формализация, сценарий атаки, киберполигон, киберучения, методологии моделирования атак.

Ampire – это программный комплекс обучения методам обнаружения, анализа и устранения последствий компьютерных атак. Комплекс предоставляет пользователю специализированное ПО для обнаружения следов компьютерных атак и инструменты для повышения уровня защищённости информационной системы, на которой проходит тренировка [1]. Все действия атакующего автоматизированы в рамках сценария, поэтому для проведения полноценных киберучений нет необходимости привлекать «живую» атакующую команду.

Ampire решает следующие задачи: помогает отработать навыки выявления компьютерных атак и расследования инцидентов информационной безопасности, учит оценивать защищённость элементов информационных сетей, налаживает взаимодействие между подразделениями, показывает способы нейтрализации и предупреждения компьютерных атак [2]. На текущий момент ресурсов для реализации полноценной структурированной атаки довольно много, но специалистов, способных из фрагментов программного кода собрать готовое решение и интегрировать его в систему, категорически не хватает.

Интегрированные сценарии атак

На данный момент в структуру киберполигона Ampire интегрировано шесть готовых решений для имитации атаки:

1. Сценарий «Защита баз данных предприятия».
2. Сценарий «Защита контроллера домена предприятия».
3. Сценарий «Защита данных файлового сервера».
4. Сценарий «Защита данных сегмента АСУ ТП».
5. Сценарий «Защита научно-технической информации предприятия»;
6. Сценарий «Защита корпоративного портала от внутреннего нарушителя».

Каждый сценарий подстроен под разработанный шаблон, т.е. под набор виртуальных машин (серверы, рабочие станции, сетевое оборудование и т.д.), которые моделируют работу типовой сети организации (например, офис, банк, предприятие). Состав шаблонов в комплексе определяется на момент поставки, но может быть расширен по требованиям заказчика. Шаблон для представленных шести сценариев представляет собой смоделированную виртуальную инфраструктуру организации и состоит из нескольких сегментов: сетевая инфраструктура, центр обработки данных, демилитаризованная зона, автоматизированная система управления технологическим процессом, сегмент пользователей и разработчиков.

Структура описания сценария построена таким образом:

1. Общая информация.

Здесь определяется:

– уровень сложности прохождения сценария (уровень сложности определяется «экспертным» мнением, учитываются особенности сценария действий нарушителя, например, количество и тип уязвимостей, количество узлов шаблона информационной системы, тип нарушителя и т.д.);

– уровень опасности злоумышленника (также экспертное мнение, однако здесь оценка будет зависеть от того инструментария, которым умеет оперировать нарушитель, а также от информации о том, владеет ли нарушитель техниками постэксплуатации);

– его вид (внутренний либо внешний нарушитель);

– какие средства обнаружения вторжений используются для детектирования вторжения (например: ViPNet IDS, TIAS).

2. Описания сценария. Здесь представляется подробная и пошаговая инструкция действий злоумышленника, а также логическая схема сценария.

3. Перечень уязвимостей. Здесь представлены несколько (в типовом варианте 3) уязвимостей, которые эксплуатируются злоумышленником при прохождении сценария. Каждая из этих уязвимостей поэтапно реализуется для достижения нарушителем своей цели. Также представлено подробное описание каждой уязвимости.

4. Способ детектирования атаки нарушителя. Здесь показано, как с использованием указанных ранее средств обнаружения вторжений выявить подозрительную активность и определить её как начало атаки на систему.

5. Установка сценария. В этом пункте определены параметры и характеристики виртуальных машин, которые должны функционировать при про-

хождении сценария. Описание виртуальной машины представлено параметрами: название, операционная система, объем оперативной памяти, объем жесткого диска. Также представлено название файлов-чекеров, которые проверяют факт закрытия каждой уязвимости.

Описание сценария

Далее будет разобран один из интегрированных сценариев. Для примера был выбран сценарий «Защита баз данных предприятия».

После обнаружения и эксплуатации уязвимости нарушитель получает доступ к серверу, который помимо основной информационной задачи предоставляет пользователям компании инструмент для генерации отчетов. С помощью этого вектора нарушитель пробует получить доступ на рабочие машины сотрудников. Главная цель – сделать дамп корпоративной базы данных. Схема первого сценария представлена на рис. 1.

Квалификация нарушителя средняя. Он умеет использовать инструментарий для проведения атак, а также знает техники постэксплуатации. Рассматриваемые уязвимости:

1) drupalgeddon2 (CVE-2018-7600);

2) слабый ssh пароль доступа к SQL серверу;

3) пароль к базе данных в отрытом виде в файле .bash_history.

В сценарии действия злоумышленника можно разбить на 3 этапа:

1. Нарушитель проводит сканирование сети 185.88.181.0/24 и находит сервер с CMS Drupal. Эксплуатируя уязвимость Drupalgeddon2, нарушитель получает контроль над хостом. Далее нарушитель генерирует reverse shell, загружает на сервер и ожидает его запуска администратором.

2. Администратор загружает вредоносный файл reporttool.py и запускает его.

3. Нарушитель, получив контроль над компьютером администратора, ищет маршрут к сети 10.10.2.0/27, далее сканирует сеть и находит ssh сервер. После проброса порта нарушитель с помощью утилиты medusa перебирает ssh пароль по словарю и получает доступ к SQL серверу. После успешного соединения нарушитель читает файл .bash_history, где хранится история команд, и среди команд находит пароль для подключения к базе данных. Нарушитель подключается к базе данных и выгружает ее.

Формальное описание сценариев с использованием существующих методологий моделирования

Формальное описание с использованием методологий моделирования является неотъемлемой частью процесса описания сценария, так как такой подход является самым утилитарным способом наглядно и понятно описать алгоритм действий злоумышленника во время проведения атаки, также такой подход поможет выделить существующие этапы кибератаки и подчеркнуть пошаговую структуру прохождения всего сценария. На рис. 1 представлена логическая схема сценария, такой подход к формальному описанию тоже имеет место быть, однако в различных подходах в разработке сценария необходим и разносторонний подход к его представлению, так как использованием различных методологий моделирования разработчиком ставятся и различные цели, достичь которые поможет лишь правильно выбранная методология [3].

Логическая схема сценария, интегрированная в систему Ampire, максимально подробно описывает структуру шаблона организации, имеющиеся в ней узлы и среды, однако по этим причинам является излишне громоздкой. Также с использованием такой схемы невозможно определить: какая всё же атака происходит на указанные узлы. Не ознакомившись с легендой и текстовым описанием сценария, работа с такой схемой будет являться довольно трудоемкой задачей. Стоит также отметить, что в логической схеме не указаны и контрмеры по противодействию атаке злоумышленника, что может стать проблемой как на этапе разработки сценариев, так и во время прохождения киберучений.

Для оптимизации процесса обучения при проведении киберучений была поставлена цель в нахождении методологии, которая будет максимально достоверно описывать процесс проведения кибератаки, в то же время снижать трудоемкость при её исследовании.

Для решения поставленной задачи были разработаны две модели для формального представления сценария «Защита баз данных предприятия». Подходы к моделированию были выбраны на основе изученной литературы и базируются на методах моделирования, связанных с построением деревьев и графов [4–6].

Формальное описание с использованием методологии языка Meta-Attack Language (MAL)

MAL (Meta Attack Language) – это языковая структура моделирования угроз, которая сочетает в себе графы вероятностных атак и защиты с объектно-ориентированным моделированием (рис. 2), которое, в свою очередь, может использоваться для создания DSL (digital subscriber line) и автоматизации анализа безопасности экземпляров моделей в каждом домене [7].

Формальное описание с использованием методологии дерева атак

Деревья атак являются формальным методом моделирования реализации угроз информационной безопасности в отношении информационных систем [8]. Атаки представляются в виде деревьев, где корнем является цель атаки, ближайшие узлы – подцелями, а листья – способами достижения подцелей и реализации атаки на основную цель (рис. 3).

Используя инструменты MAL, можно избавиться от лишней громоздкости. Также, задавая вектор атаки на конкретный домен, можно заносить результат в базу знаний и использовать уже полученные ранее знания об инциденте кибербезопасности в практических целях.

Используя такой подход, можно снизить вступительный порог для исследователей, а также сразу идентифицировать конкретные атаки. Однако, как и в случае с логической схемой, не решается проблема по указанию контрмер по противодействию атаки злоумышленника.

Методология языка MAL также отлично подходит для интегрирования подобных моделей в систе-

му обучения, однако и она имеет ряд особенностей, из-за которых для работы с такими моделями нужно иметь некоторый опыт в сфере ИБ.

Использование деревьев атак является обычной практикой при моделировании атак, вследствие чего существует обширный список инструментария для реализации такого подхода. Недостатками такого подхода можно назвать ограниченность в подробном описании того или иного шага.

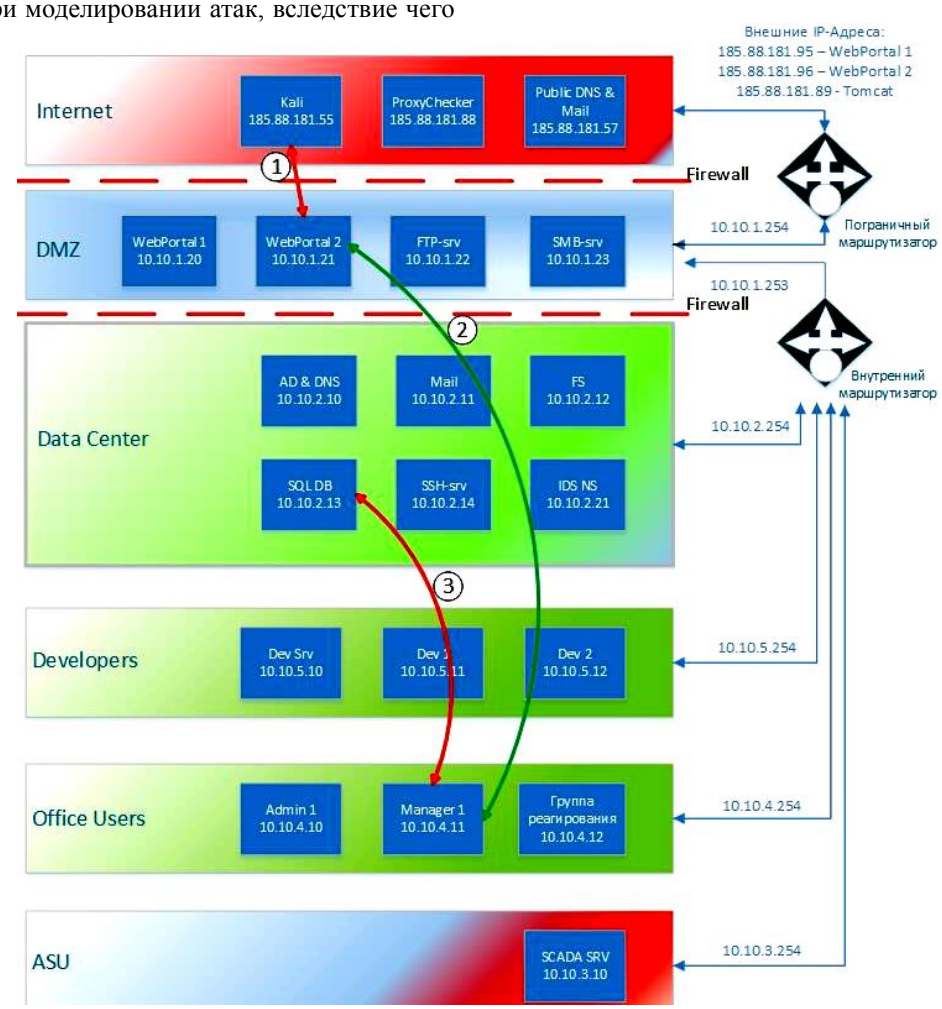


Рис. 1. Логическая схема сценария «Защита баз данных предприятия»

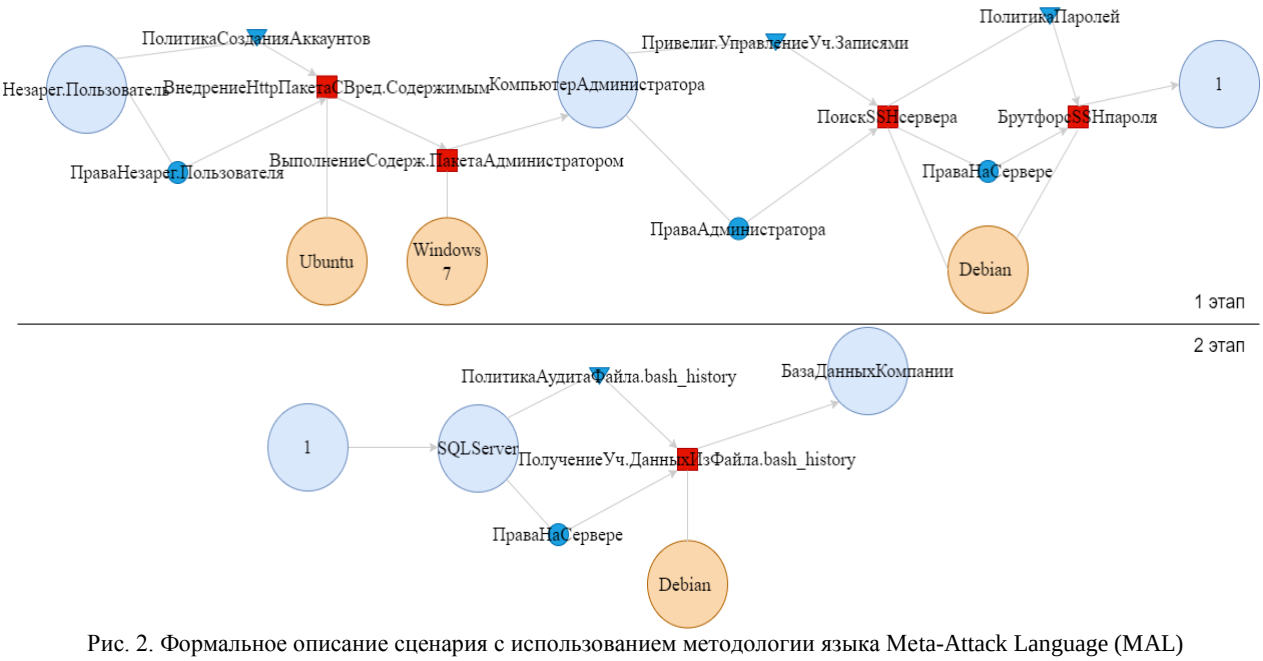


Рис. 2. Формальное описание сценария с использованием методологии языка Meta-Attack Language (MAL) «Защита баз данных предприятия»

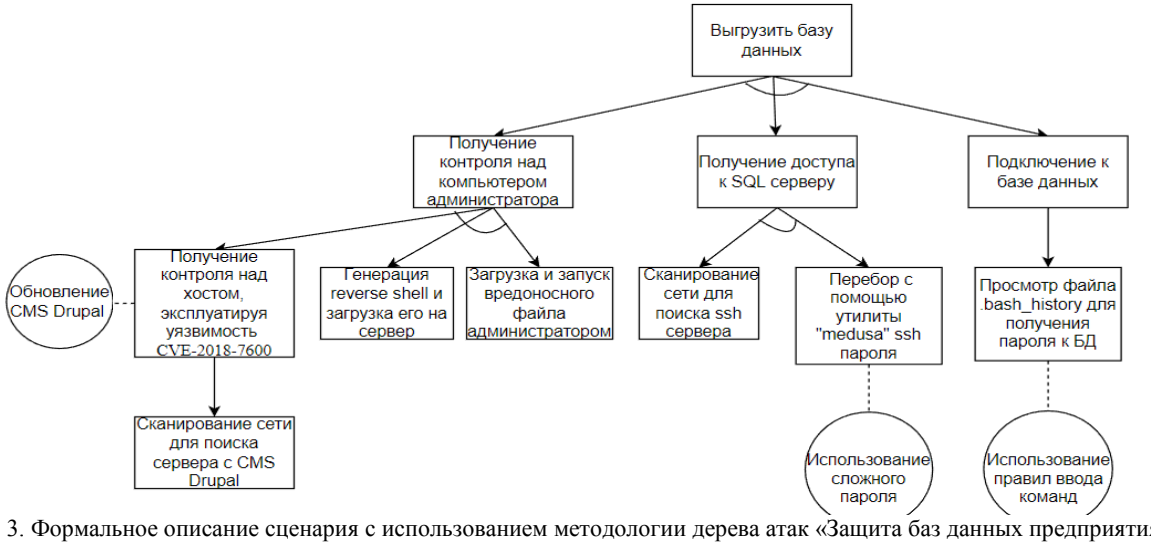


Рис. 3. Формальное описание сценария с использованием методологии дерева атак «Защита баз данных предприятия»

Используя методологию дерева атак, можно снизить вступительный порог до минимума. При использовании данной методологии моделирования решается проблема с громоздкостью и отсутствием контрмер по противодействию.

Учитывая тот факт, что речь идёт про киберполигон, цель которого – обучать новых специалистов, в частности, людей, не имеющих большого опыта в расследовании инцидентов кибербезопасности и закрытии уязвимостей, дерево атак будет наилучшим вариантом как для разработки новых решений, так и для применения таких моделей в процессе обучения.

Заключение

В результате проделанной работы было рассмотрено несколько способов формализации с использованием существующих методологий моделирования. В частности, были рассмотрены методологии моделирования MAL и «Дерево атак». На основе рассмотренных методологий были построены модели сценария «Защита баз данных предприятия», интегрированного в систему киберполигона Ampire.

Были выявлены преимущества и недостатки каждого метода. Был сделан вывод о том, что интегрированная логическая схема справляется со своей задачей не так хорошо, как разработанные альтернативные модели. В свою очередь, разработанные модели можно использовать для организации киберучений на базе киберполигона Ampire, так как такие подходы для процесса обучения будут более эффективными.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках базовой части государственного задания ТУСУРа на 2020–2022 гг. (проект № FEWM-2020-0037).

Литература

1. Информационная безопасность (тренды) [Электронный ресурс]. – Режим доступа: https://www.tadviser.ru/index.php/Статья:Главные_тенденции_в_защите_информации, свободный (дата обращения: 17.09.2022).

2. Киберполигон Ampire [Электронный ресурс]. – Режим доступа: https://amonitoring.ru/product/ampire/, свободный (дата обращения: 17.09.2022).

3. Классификация методов моделирования систем [Электронный ресурс]. – Режим доступа: https://it.rfei.ru/course/~HJ8b/~2./~2.3., свободный (дата обращения: 17.09.2022).

4. Model of Threats to Computer Network Software / A. Novokhrestov, A. Konev, A. Shelupanov // Symmetry. – 2019. – Vol. 11, Iss. 12. – 1506. – 13 p.

5. Модель угроз безопасности информации и ее носителей / А.К. Новохрестов, А.А. Конев, А.А. Шелупанов, Н.С. Егошин // Вестник Иркут. гос. техн. ун-та. – 2017. – Т. 21, № 12. – С. 93–104.

6. Computer network threat modelling / A. Novokhrestov, A. Konev, A. Shelupanov, A. Buymov // IOP Conf. Series: Journal of Physics: Conf. Series. – 2020. – Vol. 1488, 012002. – 6 p.

7. Meta Attack Language The open source platform for creation of cyber threat modeling systems [Электронный ресурс]. – Режим доступа: https://mal-lang.org, свободный (дата обращения: 17.09.2022).

8. Деревья атак и их применение при анализе проблемы безопасности и защищённости программных продуктов [Электронный ресурс]. – Режим доступа: https://www.software-testing.ru/library/testing/security/140, свободный (дата обращения: 17.09.2022).

Егошин Николай Сергеевич
Канд. техн. наук, доцент каф. КИБЭВС ТУСУРа
ORCID: 0000-0003-4770-0701
Эл. почта: ens@fb.tusur.ru

Сермавкин Никита Игоревич
Студент каф. безопасности информационных систем (БИС) ТУСУРа
Эл. почта: iis.vseverske@mail.ru

Репкин Владимир Сергеевич
Студент каф. БИС ТУСУРа

Калякин Антон Дмитриевич
Студент каф. БИС ТУСУРа
Эл. почта: a.d.kalyakin@gmail.com

УДК 519.17

А.Ю. Якимук, С.А. Устинов, Т.П. Лазарев, А.С. Коваленко

Методы формализации описания сценариев кибератак

Проблема оценки качества разрабатываемых систем наиболее актуальна в наши дни. Суть проблемы заключается в том, что какая-либо формализованная методика, которая могла бы реально оценить безопасность информационной системы, отсутствует, из-за этого оценка системы ложится на плечи эксперта, поэтому многое зависит от квалификации эксперта.

Ключевые слова: граф атаки, дерево атаки, метаязык, cyber kill chain.

Чаще всего классификация атак направлена на отдельные факторы моделирования. Типы атак, способность учитывать динамические особенности атак, а также многое другое будут выступать в роли аспектов моделирования. Существуют различные подходы к формализации угроз и атак на основе теории графов, например, описание с помощью «Дерева атак», «Графа атак», модели Cyber Kill Chain, а также описание атак с помощью метаязыка [1–3].

Формальное описание сценариев с помощью «Дерева атак»

Первый метод формализации, который будет рассмотрен в данной статье, – «Дерево атак». Особенностью данного метода является то, что атака будет представлена в виде дерева, основой которого являются узлы, находящиеся ближе всего, т.е. корень. А способами, благодаря которым достигается реализация атаки на выбранную цель, выступают листья. Помимо корневого узла, существует еще два типа узлов, выполняющих операцию «логическое И» и «ИЛИ». Узел, выполняющий логическую операцию «И», подразумевает обособленные шаги для выполнения поставленной цели. Узел, представляющий собой возможные варианты достижения цели, выполняет логическую операцию «ИЛИ» [4].

На рис. 1 показано, как обозначается операция «логическое И». На рис. 2 показано, как обозначается операция «логическое ИЛИ».

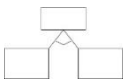


Рис. 1 Операция «логическое И»

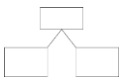


Рис. 2. Операция «логическое ИЛИ»

Для осуществления атаки требуется обойти все дочерние узлы типа И или хотя бы один узел типа ИЛИ.

Для того чтобы сформировать дерево атак, первым делом необходимо определить потенциальные цели атак, после чего все возможные цели атаки становятся корнем своего дерева атаки. Также важно пояснить, что во время создания дерева рассматриваются все возможные атаки на выбранные ранее цели. Данные атаки образуют дочерние узлы «И» и «ИЛИ», при этом все атаки будут рассмотрены как цель и будут сформированы ее дочерние узлы. После чего данный процесс продолжится рекурсивно. В уже сформированном дереве атаки поддеревом выступает сценарий атаки, который включает в узел все его дочерние узлы «И» и как минимум один «ИЛИ». Таким же образом будут определены все поддеревья для дочерних подцелей.

К каждому узлу может быть привязан какой-либо атрибут. Благодаря таким атрибутами есть возможность выявить атаки, которые используют определенные параметры, данные функции могут помочь с выявлением угроз, а также требуемых защитных мер. Также можно использовать разные типы стрелок, соединяющие листья дерева, для учёта особенности нижнего блока (например, вероятность реализации маленькая – пунктирная линия у стрелки). Деревья атак подходят для создания сценариев атак, определения уязвимостей, эксплуатация которых может нанести серьезный ущерб, а также для создания вариантов закрытия уязвимостей [5]. Контрмеры прописываются в кружках (◌).

На рис. 3 показан пример графа, построенного с использованием методологии «Дерева атак».

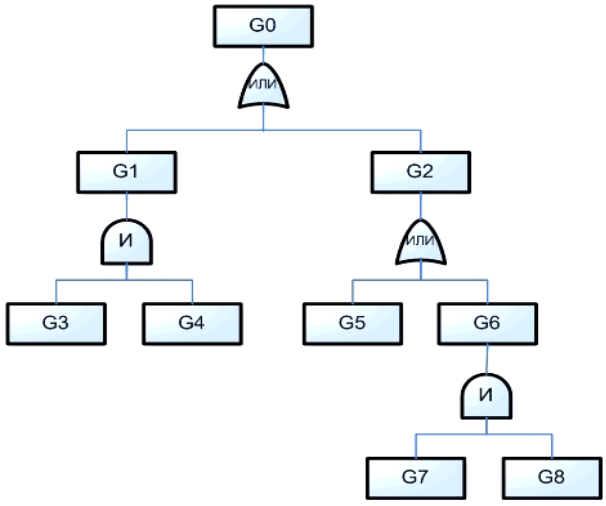


Рис. 3. Пример построения «Дерева атак»

Формальное описание сценариев атак с помощью «Графа атаки»

Метод формализации, основанный на графе атаки, подразумевает под собой перечень потенциальных действий злоумышленников, направленных на эксплуатацию имеющихся уязвимостей. Последовательность таких действий носит название «Путь атаки» [6]. Ниже представлен вариант простейшего «Графа атаки» (рис. 4), в котором действия злоумышленника обозначены как P_1, P_2 ... P_{10}.

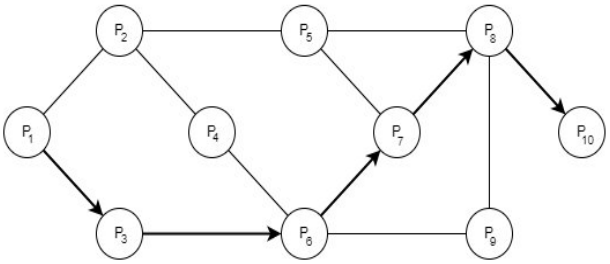


Рис. 4. Пример простейшего «Графа атаки»

Зачастую графы атак используются в контексте анализа защищенности сетей. Данный анализ означает последовательную проверку всех хостов сети на наличие уже известных уязвимостей. В результате после проверки будет получен отчет, в котором будет список обнаруженных уязвимостей, в том числе и рекомендации по их устранению. Но существует иная модель анализа защищенности, которая учитывает «топологию» компьютерной системы, а именно характеристики, свойства и связь объектов в рамках этой системы. Данный анализ называется топологическим, а функционал, на основе которого выполняется работа, называется «топологические сканеры безопасности». В настоящие время происходит постепенное продвижение данной технологии.

Уже созданный граф имеет все возможные варианты достижения злоумышленником своих целей. В результате проверки построенного графа можно получить данные, свидетельствующие, например, о списках успешных атак, которые по той или иной причине не смогла найти система обнаружений вторжений, о соответствии уровня защищенности сети и принятых мер безопасности, о перечне наиболее важных уязвимостей, рекомендации по предотвращению эксплуатации уязвимостей, для которых не предусмотрены обновления, а также определенный набор мер, необходимых для защищенности сети.

Основной проблемой данного метода формализации выступает масштабируемость, это значит, что построить граф атаки для сети, в которой находится большое число хостов и уязвимостей, построить будет крайне проблематично [7].

Метод формализации сценариев кибератак с помощью Cyber Kill Chain

Модель Cyber Kill Chain показывает, какие шаги должны предпринять злоумышленники для достижения своих целей. При использовании данной модели можно наглядно увидеть, что блокировка хакера на каком-либо из этапов атаки приводит к разрыву всей цепочки атаки.

Зачастую для успешной реализации атаки хакерам необходимо пройти всю цепочку шагов, в свою очередь, специалистам информационной безопасности, достаточно заблокировать их на каком-либо этапе, чтобы разорвать цепочку шагов и добиться минимального успеха. Конечная точка – это точка, через которую проходят все атаки, поэтому остановка атаки на данной точке сильно повышает вероятность на противодействие любой кибератаке [8].

На рис. 5 показан сценарий, представленный с помощью Cyber Kill Chain.

Формализация сценария с помощью графа «Meta attack»

MAL (Meta Attack Language) – это языковая структура моделирования угроз, которая сочетает в себе графы вероятностных атак и защиты с объектно ориентированным моделированием, которое, в свою очередь, может использоваться для создания DSL (digital subscriber line) и автоматизации анализа безопасности экземпляров моделей в каждом домене. Также он предоставляет формализм, который позволяет полуавтоматически генерировать, а также эффективно вычислять очень большие графы атак [9].

MAL позволяет экспертам по безопасности кодифицировать знания, относящиеся к конкретной предметной области, чтобы можно было моделировать атаки на системы в интересующей области. Созданные таким образом языки моделирования атак для конкретного домена впоследствии могут повторно использоваться людьми с меньшим опытом в области безопасности для автоматической оценки безопасности конкретных систем в домене [10, 11].

Основной большой круг голубого цвета определяет используемые активы в системе, например, база данных предприятия или компьютер администратора. Переход между активами по шагам злоумышленника происходит через квадраты (□ – логические шаги типа «И»), которые определяют итог атаки, например внедрение вредоносного пакета в систему или получение данных из базы данных предприятия. Сам ход атаки описывается 2 операторами. Первый – это маленький кружок (● – логические шаги типа «ИЛИ»), который показывает то, благодаря чему была выполнена атака злоумышленника, например, права захваченного компьютера администратора. С другой же стороны треугольником (∇) показывается возможный шаг защиты от выполняемой атаки. Относительно каждой успешной атаки в нижних кружках оранжевого цвета показывается используемая система, например, Windows 7, Debian или любая другая ОС.

На рис. 6 показан граф сценария, построенного с помощью языка метаатак (Meta Attack Language).

Преимущества и недостатки рассмотренных методов

К главным преимуществам моделей, в основе которых лежит дерево атаки, относятся адаптируемость и отличная масштабируемость. К недостаткам относятся сложность создания моделей технической атаки, а также нужда в динамическом моделировании [12].

В результате анализа графа атаки будет получен отчет, в котором будет список имеющихся уязвимостей, а также рекомендации по их устранению.

Минусы данной модели аналогичны недостаткам моделей, основанных на методе дерева атаки, а именно трудность моделирования циклических атак и слабая масштабируемость.

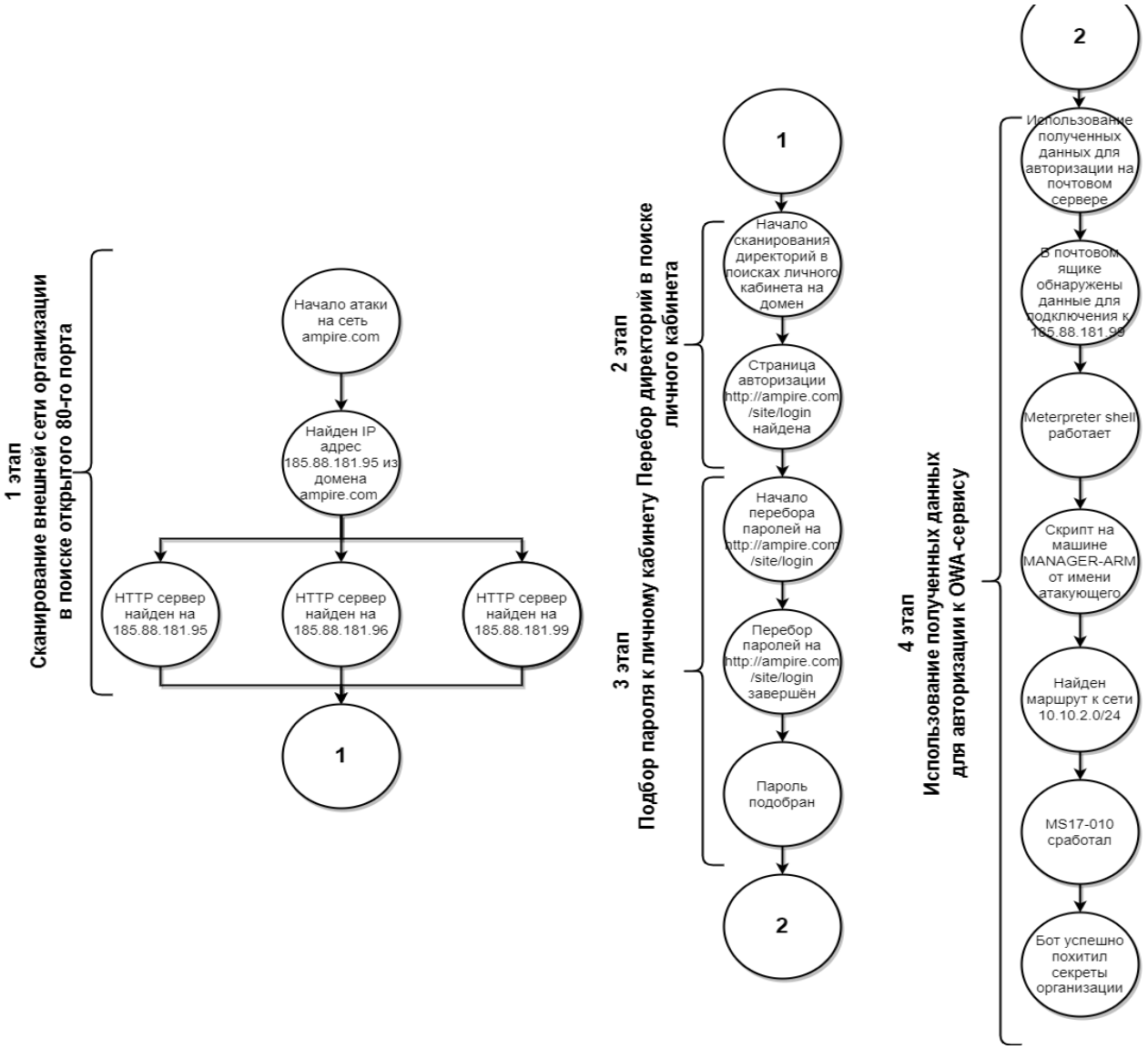


Рис. 5. Сценарий, представленный с помощью Cyber Kill Chain

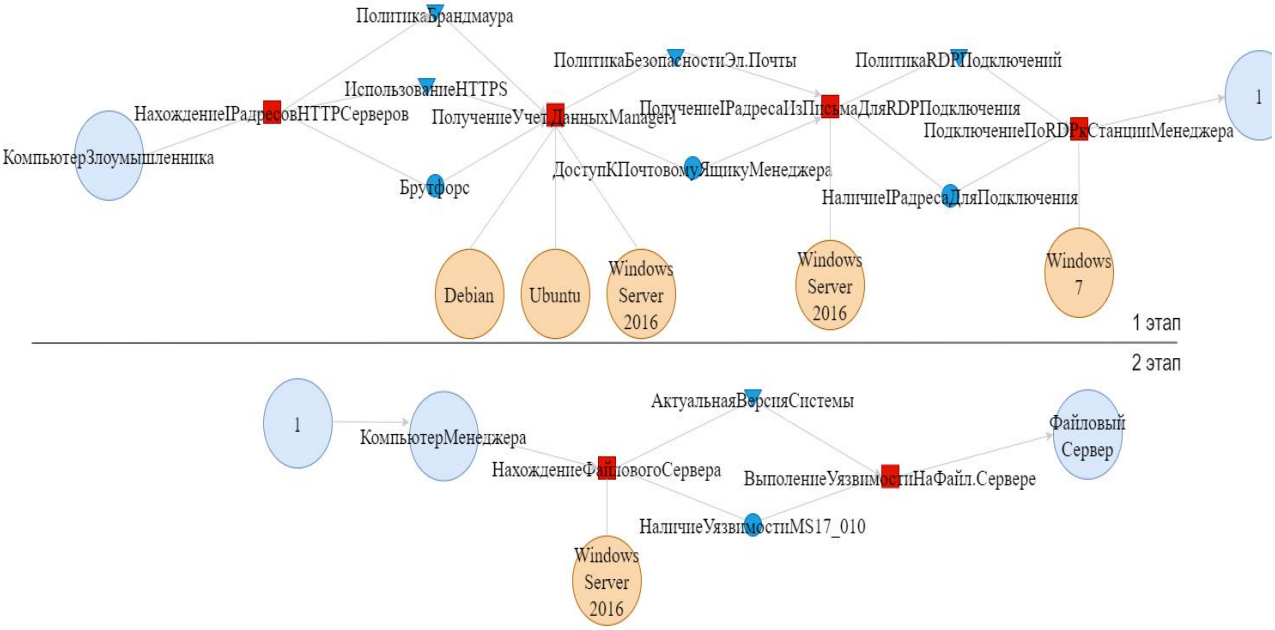


Рис. 6. Сценарий, представленный с помощью языка метаатак (Meta Attack Language)

Cyber Kill Chain, как и большинство способов описания атак, подтверждает тот факт, что ограничение злоумышленников на любой стадии прекращает всю цепочку атаки. Но в то же время с помощью данного метода возможно противодействовать любой кибератаке.

Созданные сценарии атак с помощью метаязыка могут использоваться людьми с меньшим опытом в области безопасности для автоматической оценки безопасности конкретных систем в домене, однако это может снизить квалификацию эксперта ИБ.

Заключение

Подводя итог всему вышесказанному, можно сделать несколько выводов по разобранным методам формального описания. Для наиболее универсального описания подойдут модели, основанные на деревьях атак. Если необходимо последовательное сканирование всех хостов, то подойдет метод графа атак. Cyber Kill Chain подойдет для наиболее автоматизированного описания. В свою очередь, метаязык подойдет скорее для разработчиков сценариев, чем для начинающихся учиться основам ИБ.

Таким образом, предложенные в данной статье методы формального описания атак могут помочь в создании сценариев атак с использованием общей понятийной базы.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ в рамках базовой части государственного задания ТУСУРа на 2020–2022 гг. (проект № FEWM-2020-0037).

Литература

1. Новохрестов А.К., Конев А.А. Многоуровневая модель информационной системы на основе атрибутивных метаграфов // Матер. науч.-техн. конф. «Электронные средства и системы управления». – Томск, 2015. – Ч. 1. – С. 184–188.

2. Model of Threats to Computer Network Software / A. Novokhrestov, A. Konev, A. Shelupanov // Symmetry. – 2019. – Vol. 11, Iss. 12. – 1506. – 13 p.

3. Модель угроз безопасности информации и ее носителей / А.К. Новохрестов, А.А. Конев, А.А. Шелупанов, Н.С. Егошин // Вестник Иркут. гос. техн. ун-та. – 2017. – Т. 21, № 12. – С. 93–104.

4. Деревья атак и их применение при анализе проблемы безопасности и защищённости программных продуктов [Электронный ресурс]. – Режим доступа: https://software-testing.ru/library/23-security-testing/140-2008-10-02-18-15-03, свободный (дата обращения: 28.09.2022).

5. Гаранжа А.В., Губенко Н.Е. Применение метода построения деревьев атак для защиты интернет-магазинов [Электронный ресурс]. – Режим доступа: http://masters.donntu.org/2017/fknt/garanzha/library/article2.htm.%202017, свободный (дата обращения: 10.10.2022).

6. Карпов Д.С. Графовые модели систем защиты информации [Электронный ресурс]. – Режим доступа: https://ppt-online.org/578597, свободный (дата обращения: 15.10.2022).

7. Колегов Д.Н. Проблемы синтеза и анализа графов атак [Электронный ресурс]. – Режим доступа: https://www.securitylab.ru/contest/299868.php, свободный (дата обращения: 8.10.2022).

8. Что такое Cyber-Kill Chain и почему ее надо учитывать в стратегии защиты [Электронный ресурс]. – Режим доступа: https:// https://habr.com/ru/company/panda/blog/327488, свободный (дата обращения: 17.10.2022).

9. A systematic literature review of information sources for threat modeling in the power systems domain [Электронный ресурс]. – Режим доступа: https://link.springer.com/chapter/10.1007/978-3-030-58295-1_4, свободный (дата обращения: 17.10.2022).

10. Accelerating large graph algorithms on the gpu using cuda [Электронный ресурс]. – Режим доступа: https://link.springer.com/chapter/10.1007/978-3-540-77220-0_21, свободный (дата обращения: 15.10.2022).

11. A meta language for threat modeling and attack simulations [Электронный ресурс]. – Режим доступа: https://link.springer.com/chapter/10.1007/978-3-540-77220-0_21, свободный (дата обращения: 20.10.2022).

12. Методы и средства моделирования атак в больших компьютерных сетях: состояние проблемы / Д.И. Котенко, И.В. Котенко, И.Б. Саенко // Тр. СПИИРАН. – 2012. – Вып. 22.

Якимук Алексей Юрьевич
Канд. техн. наук, доцент, каф. КИБЭВС ТУСУРа
Эл. почта: yay@fb.tusur.ru

Устинов Сергей Алексеевич
Студент каф. БИС ТУСУРа
Эл. почта: ustinov0304@gmail.com

Лазарев Тимофей Петрович
Студент каф. БИС ТУСУРа
Эл. почта: tima.lazarev.68@mail.ru

Коваленко Александр Сергеевич
Студент каф. БИС ТУСУРа
Эл. почта: a.s.kovalenko@mail.ru

Секция 16

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В УПРАВЛЕНИИ И ПРИНЯТИИ РЕШЕНИЙ

*Председатель секции – **Сенченко Павел Васильевич**, проректор по УР, доцент каф. АОИ, к.т.н.*

УДК 621

А.Б. Кураков, Р.С. Кульшин, Г.А. Волокитин, С.В. Коновалов

Разработка языка шаблонов кодогенерации для информационной системы автоматизации разработки мобильных приложений

Рассматривается система генерации шаблонов и воспроизведения программного кода. Работа выполнена в рамках программы фонда содействия инновациям «Студенческий стартап» – Система автоматизации разработки мобильных приложений ETICON Core, договор №503ГССС15-L/78798 от 08.09.2022.
Ключевые слова: шаблоны, генерация, воспроизведение, автоматизация, low-code система.

В разработке больших программных продуктов встречается очень много однотипного и шаблонного кода, что приводит к тому, что разработчик программного продукта вынужден тратить больше времени на реализацию продукта из-за постоянного повторения однотипных действий. Из-за этого увеличиваются сроки реализации программного продукта, а написание шаблонного кода зачастую приводит к эмоциональному выгоранию специалистов. Решается проблема написания однотипного и шаблонного кода при разработке мобильных приложений.

Держателями проблемы являются IT-компании. С ростом спроса на разработку мобильных приложений IT-компаниям теперь требуется как можно быстрее и качественнее создавать программный продукт, чтобы увеличивать свою прибыль и количество клиентов. Использование разрабатываемого программного обеспечения позволяет использовать уже готовые гибко настраиваемые шаблоны для кодогенерации. Кодогенерация позволяет избежать мелких ошибок, которые могут впоследствии привести к катастрофическим последствиям и финансовым потерям компании. Также разрабатываемый продукт позволяет ускорить разработку мобильных приложений. Использование генератора кода позволит максимально быстро интегрировать нового члена команды в разработку, который еще не был знаком с уникальным стилем написания кода компании.

ETICON Core

ETICON Core – это No-code система генерации кода для мобильных приложений [1].

Разрабатываемое программное обеспечение позволяет сократить время разработки на 10–15%, предоставляет более 20 различных гибких шаблонов. Имеет более 10 различных функций генерации кода (классы, функции и т.д.). Реализует более 5 шаблонов (MVC, MVVM, BLoC и др.) [2]. Является эффективным инструментом при разработке мобильных приложений. Позволяет сериализовать и десериализовать JSON-объекты в модели данных для обеспечение сетевого взаимодействия. Встроенный алгоритм анализа уже существующих моделей данных в совокупности с использованием рекомендательных алгоритмов позволяет не дублировать уже существующие модели, а переиспользовать или дорабатывать их.

На текущий момент на рынке существуют различные no-code и low-code системы, однако они не предлагают никакой гибкости в использовании шаблонов. Поэтому предполагается, что команде необходимо разработать собственное решение для генерации кода. Основой для разработки будет являться разработанный ранее пакет для языка dart eticon_struct и eticon_extension. Одним из ключевых моментов развития данного ПО является создание пользовательских шаблонов. Именно для этих целей и создается язык шаблонов.

Язык шаблонов

Назначение:

- Создание пользовательских шаблонов программного кода (но язык не ориентирован на ручное написание, для создания требуется GUI).
- Хранение шаблонов.
- Реализация шаблонов.

Функции языка:

- Вставка других шаблонов.
- Декларирование структуры.
- Декларирование переменных.
- Указание полей, вводимых пользователем при реализации.
- Возможность добавлять статический код языка.
- Импорт библиотек с их установкой.
- Импортирование файлов.

Возможно, будут зарезервированные переменные (например, путь до папки).

Существует 2 типа шаблонов: структурные и бесструктурные.

Ключевые слова языка шаблонов:

- name <имя шаблона> – объявляется имя шаблона;
- global: <имя переменной 1, имя переменной 2> – глобальные переменные будут вводиться пользователем, который реализует шаблон;
- struct – объявляет пути где должны быть размещены генерируемые файлы, и задает файловую структуру;
- file <имя файла> – задает название для генерируемого файла;
- import <имя файла/библиотеки> – подключает уже существующие файлы или библиотеки;
- insert <имя шаблона> – добавляет созданный ранее шаблон в код создаваемого шаблона.

Ниже приведен листинг кода, написанный на языке шаблонов.

Листинг 1. Пример шаблона

```
name Kubik

global: USER_STATE, USER_INPUT_1, USER_INPUT_2,
PATH

struct
/screens/$PATH/cubit/cubit.dart
/screen/$PATH/cubit/state.dart

file state.dart
|abstract class| $USER_STATE|State{}|
|class Loading|$USER_STATE| extends $US-
ER_STATE|State{}|
|class Loaded|$USER_STATE| extends $US-
ER_STATE|State{}|

file cubit.dart
import file state.dart
import library block

|class| $USER_INPUT_1 |extends Cubit<$USER_STATE>{|
$USER_INPUT_1|(): super(Loading|$USER_STATE|());|
|void| $USER_INPUT_2|() async{|
insert request.elang
|emit(Loaded|$USER_STATE|);|
|}|
```

Заключение

Гибкость разработанного языка позволяет генерировать код для любого языка программирования с любой структурой проекта, которую каждая команда разработчиков может удобно подстраивать под свои требования. Применение данного языка как часть разрабатываемой информационной системы ETICON Core позволило ускорить разработку мобильных приложений на 5–8%.

Литература

1. Кульшин Р.С. Программное обеспечение кодогенерации и его использование при разработке мобильных приложений / Р.С Кульшин и др. // Матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР», Томск, 18–20 мая 2022 г.: в 3 ч. – Томск: В-Спектр, 2022. – Ч. 2. – 318 с.

2. Разделение бизнес-логики и UI во Flutter с помощью BLoC-архитектуры [Электронный ресурс]. – Режим доступа: https://inostudio.com/blog/articles-develop/razdelenie-biznes-logiki-i-ui-vo-flutter-s-pomoshchyu-bloc-arkhitektury/ (дата обращения: 10.09.2022).

Кураков Андрей Борисович
Студент, ассистент каф. АСУ ТУСУРа
Эл. почта: sniperaft@bk.ru

Кульшин Роман Сергеевич
Студент, ассистент каф. АОИ
М.н.с. лаб. объектно-ориентированного моделирования информационных систем (ЛООМИС) ТУСУРа
Эл. почта: roman.s.kulshin@tusur.ru

Волокитин Геннадий Александрович
Студент, ассистент каф. АОИ
М.н.с. ЛООМИС ТУСУРа
Эл. почта: kolesolek97@gmail.com

Коновалов Степан Вячеславович
Студент каф. АОИ ТУСУРа
Эл. почта: konovalov@eticon.ru

УДК 004.89

Г.А. Волокитин, Р.С. Кульшин, А.А. Сидоров, П.В. Сенченко

Использование решающих деревьев для задачи выбора товара в системах цифровых двойников

Рассматривается применение решающих деревьев для применения в цифровых двойниках, в частности процесс выбора товара покупателем. Описаны процесс выбора и теоретико-информационный критерий.
Ключевые слова: цифровые двойники, ритейл, решающие деревья.

В современном мире использование цифровых двойников становится все более частым явлением. В основном они используются в разработке прототипов реальных физических производственных объектов, используя различные датчики, они отслеживают показатели физических объектов. В настоящее время мощности вычислительных устройств позволяют существенно расширить возможности цифровых двойников и для применения их не только для физических объектов, а вместо датчиков используют наборы данных (dataset). В частности, их используют для решения следующих задач:

1) управление экономической эффективностью цепочки создания стоимости;
2) управление программой цифровой трансформации;
3) управление развитием активов;
4) интегрированное производственно-экономическое планирование на основе нормативов;
5) инвестиционное планирование;
6) управление портфелями проектов;
7) управление инновациями;
8) управление рисками.

Одним из ярких прикладных примеров использования цифровых двойников в отношении нематериальных по сути моделирования объектов является ритейл. Данная сфера стала сильно развиваться в последнее время, а появление большого количества различных онлайн-маркетплейсов очень хорошо этому способствует. Также хорошим толчком к развитию стала пандемия, после которой люди привыкли к покупке товаров с функцией доставки.

В основу цифрового двойника для ритейла входит концепция использования трех моделей:

1. Модель покупателя, которая позволяет на основе данных о покупателях подбирать персональные скидки и рекомендации к покупке товаров. Содержит в себе набор данных о покупателе, и набор функций.

2. Модель продавца или магазина, в которой есть возможность обрабатывать количество товаров, их наличие, распределение продавцов в магазине и отслеживание потока покупателей по времени.

3. Модель товара. На основе данных о товаре можно определить его популярность, в зависимости от нее можно уже варьировать цену и расставлять конкретный товар в зависимости от проходимости стеллажей.

Объединив эти модели, есть возможность контролировать продажи, управлять скидками и ценами, а также использовать возможности своевременной поставки товаров. Это центральный, хотя и потенциально не весь список возможностей данного типа моделей, способствующих улучшению процесса продаж [1].

Также цифровой двойник должен предоставлять функционал принятия решений. Поскольку цифровой двойник является большой системой, состоящей из несколько математических моделей, выбор методов принятия решений является основным и приоритетным. В зависимости от количества функций могут использоваться разные варианты методов, будь то машинное обучение или моделирование.

Поскольку есть задачи, которые необходимо решить, нужно определить метод решения этих задач. В случае использования цифровых двойников в ритейле как пример рассматривается выбор необходимости приобретения товара покупателем. В рамках данной работы планируется использовать решающие деревья для решения задачи выбора товара.

Принцип использования решающих деревьев

Поскольку использование цифрового двойника служит для принятия определенных решений, было решено разобрать и применить принцип решающих деревьев.

Решающее дерево – это решение задачи обучения с учителем, основанное на том, как решает задачи прогнозирования человек. В общем случае – это k-ичное дерево с решающими правилами в нелистовых вершинах и некоторым заключением о целевой функции в листовых вершинах (прогнозом). Решающая правило – некоторая функция от объекта, позволяющее определить, в какую из дочерних вершин нужно поместить рассматриваемый объект. В листовых вершинах могут находиться разные объекты: класс, который нужно присвоить попавшему туда объекту (в задаче классификации), вероятности классов (в задаче классификации), непосредственно значение целевой функции (задача регрессии) [2].

Рассмотрим алгоритм покупки товара покупателем. Для этого рассмотрим критерии покупки товара:

1) необходимость покупки;
2) финансовые возможности;
3) покрытие потребности;
4) отношение к бренду;
5) наличие скидки;
6) наличие повреждений;

7) исправность товара;
8) актуальность товара.

Блок-схема процесса выбора и покупки товара представлена на рис. 1. На схеме видно, что для того, чтобы осуществить покупку товара, необходимо, чтобы товар подходил всем представленным критериям. Поскольку на представленной схеме видно, что ответы могут быть либо положительными, либо отрицательными, можно сделать вывод, что тип решающих деревьев – это бинарное дерево. Разбиение здесь осуществляется по заданному правилу, для которого определяется атрибут. Данный атрибут должен разделить множество наблюдений в узле так, чтобы результирующие подмножества содержали примеры с одинаковыми объектами или были максимально к этому приближены. Наибольшей популярностью пользуется теоретико-информационный критерий, который задается формулой

$$H = -\sum_{i=1}^{n} \frac{N_i}{N} \log\left(\frac{N_i}{N}\right),$$

где n – число объектов в исходном подмножестве, N_i – число примеров i-го класса, N – общее количество примеров в подмножестве [3].

Рис. 1. Блок-схема процесса выбора товара

Данная формула позволяет определить, насколько подмножества неоднородные по представленным в нем объектам. Когда объекты представлены в равных долях и неопределенность классификации максимальная, то и энтропия тоже максимальная. В конкретном примере на основе данных

из модели покупателя формула позволяет определить список товаров, наиболее подходящих покупателю.

Также при помощи формулы появляется возможность классифицировать и сортировать критерии в зависимости от приоритета. Сортировка позволит в первую очередь проверять более точные критерии, от которых больше всего зависит выбор товара. Поскольку несоответветсвие какому-либо критерию не означает отказ от покупки, появляется возможность подобрать другой товар, с другим брендом или в другой комплектации.

Заключение

В данной работе рассматривается применение решающих деревьев для определения процесса выбора товара покупателем в модели покупателя в цифровом двойнике. Выбор товара покупателем был рассмотрен с помощью применения решающего бинарного дерева, в частности, описан пример использования теоретико-информационного критерия.

Работа выполнена в рамках ГЗ «Наука» Минобрнауки РФ. Проект FEWM-2020-0036 «Методическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде».

Литература

1. Использование цифровых двойников в ритейле / Г.А. Волокитин, Р.С. Кульшин, С.В. Коновалов // Научная сессия ТУСУР–2022: матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых, Томск, 18–20 мая 2022 г.: в 3 ч. – Томск: В-Спектр, 2022. – Ч. 2. – 318 с.

2. Синицин Ф. Учебник по ML от ШАД: Решающие деревья [Электронный ресурс]. – URL: https://ml-handbook.ru/chapters/decision_tree/intro] (дата обращения: 18.09.2022).

3. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning. – Springer, 2009.

Волокитин Геннадий Александрович
Студент, ассистент каф. автоматизации обработки информации (АОИ)
М.н.с. лаб. объектно-ориентированного моделирования информационных систем (ЛООМИС) ТУСУРа
Эл. почта: gennadii.volokitin@tusur.ru

Кульшин Роман Сергеевич
Студент, ассистент каф. АОИ
М.н.с. ЛООМИС ТУСУРа
Эл. почта: roman.s.kulshin@tusur.ru

Сидоров Анатолий Анатольевич
Зав. каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: anatolii.a.sidorov@tusur.ru

Сенченко Павел Васильевич
Доцент каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: pvs@tusur.ru

УДК 004.4’2

С.Е. Лопатин, Р.С. Кульшин, Г.А. Волокитин

Разработка системы тестирования мобильных приложений

При разработке мобильного программного обеспечения появляется потребность в его тестировании. Для платформы IOS используется TestFlight – официальный клиент компании Apple. Для платформы Android такого же простого и функционального решения на рынке не представлено. Из-за этого разработчики испытывают проблемы с безопасностью и простотой процесса тестирования мобильных приложений. В данной статье рассматривается возможность создания системы тестирования мобильных приложений, которая решала бы возникающие у разработчиков проблемы.

Ключевые слова: мобильные приложения, тестирование, безопасность.

Рынок мобильных приложений на платформе Android за 2020 г. создал доход в $38,6 млрд (примерно $1200 каждую секунду), и эта цифра растет с каждым годом. Появление новых мобильных игр и мобильных приложений раз за разом сотрясает рынок программ для телефонов, а новый разрабатываемый функционал привычных приложений способен возродить интерес к казалось бы забытым нами программам.

В то же время устройства Android и Apple с каждым годом все больше начинают быть похожими друг на друга. Однако в некоторых аспектах Android приложения до сих пор не могут приблизиться к конкурентам из Apple. Безопасность и конфиденциальность данных – частая проблема в смартфонах на операционной системе Google, как, например, уязвимости модулей в машинных кодах, уязвимости механизмов межкомпонентного взаимодействия, уязвимости модификаций и компонентов производителей устройств и т.д. С проблемами конфиденциальности данных могут столкнуться не только пользователи Android-устройств, но и разработчики программного обеспечения под эту платформу. В процессе разработки многие компании тестируют свои мобильные приложения функциональным (ручным) методом. Это позволяет тестировщикам получать доступ к apk-файлам приложения с возможностью использования в личных целях, декомпиляции приложения и нарушения конфиденциальности данных, являющихся собственностью компании-разработчика.

В компании Apple данную проблему решает служба бета-тестирования для приложений iOS TestFlight, доступная через AppStore Connect. Каждая сборка доступна для тестирования в течение 90 дней со дня ее загрузки разработчиком. Оставшийся срок тестирования отображается под названием приложения в TestFlight. TestFlight каждый раз оповещает тестировщика о появлении новой сборки вместе с указаниями целей тестирования. В TestFlight 3 или новее можно также включить автоматические обновления, чтобы последние бета-версии сборок устанавливались автоматически. По окончании периода тестирования бета-версия сборки становится недоступной.

Принцип работы

Работа приложения распределяется на 2 платформы: веб-приложение и мобильное приложение.

Веб-приложение необходимо для разработчиков, чтобы они имели возможность загружать новые версии сборки реализуемой программы на сервер, указывая, какую часть приложения необходимо протестировать, приглашать в команду тестировщиков новых людей и создавать проект приложения. Также в проекте приложения будет доступен список всех тестировщиков и их отчетов, написанных для каждой отдельной версии приложения. Разработчик имеет возможность указать, какая часть приложения нуждается в тестировании.

Разработчик входит или регистрируется в сервисе. Далее он может создать новый проект приложения либо же войти в существующие проекты. Для создания нового проекта достаточно загрузить первую версию для бета-тестирования. Впоследствии данные о проекте приложения можно будет изменить.

Следующим шагом для разработчика является приглашение участников в бета-тестирование проекта. Тестировщику в этом случае придет push-уведомление о приглашении в проект на телефон.

Мобильное приложение необходимо для тестировщиков, чтобы они могли принимать, просматривать и скачивать новые сборки реализуемого программного обеспечения, давать оценки и оставлять отчет об ошибках. Приложение оповещает тестировщиков о появлении новых версий приложения через push-уведомления, сохраняет последние сборки только определенное количество времени и исключает возможность передачи пользователю доступа к apk-файлам приложений, доступных для тестирования (рис. 1).

После получения тестировщиком доступа к проекту приложения он имеет возможность скачать последнюю доступную его версию. Впоследствии он может создавать форму, в которой можно указать тип, причину и сценарий ошибки, прикрепить фото или видео момента сбоя.

Разработчику предоставляются проанализированные и сформированные данные о количестве потраченного времени каждым из тестировщиков, типе устройства, а также отчет об ошибках пользователей по итогам тестирования о каждой части, необходимой для тестирования.

Разработчику доступен полный список всех версий тестируемых приложений, в то время как

тестировщик имеет доступ только к последней его версии. При необходимости разработчик может дать доступ к конкретной прошлой версии приложения конкретным тестировщикам, указав период времени действия доступа (рис. 2).

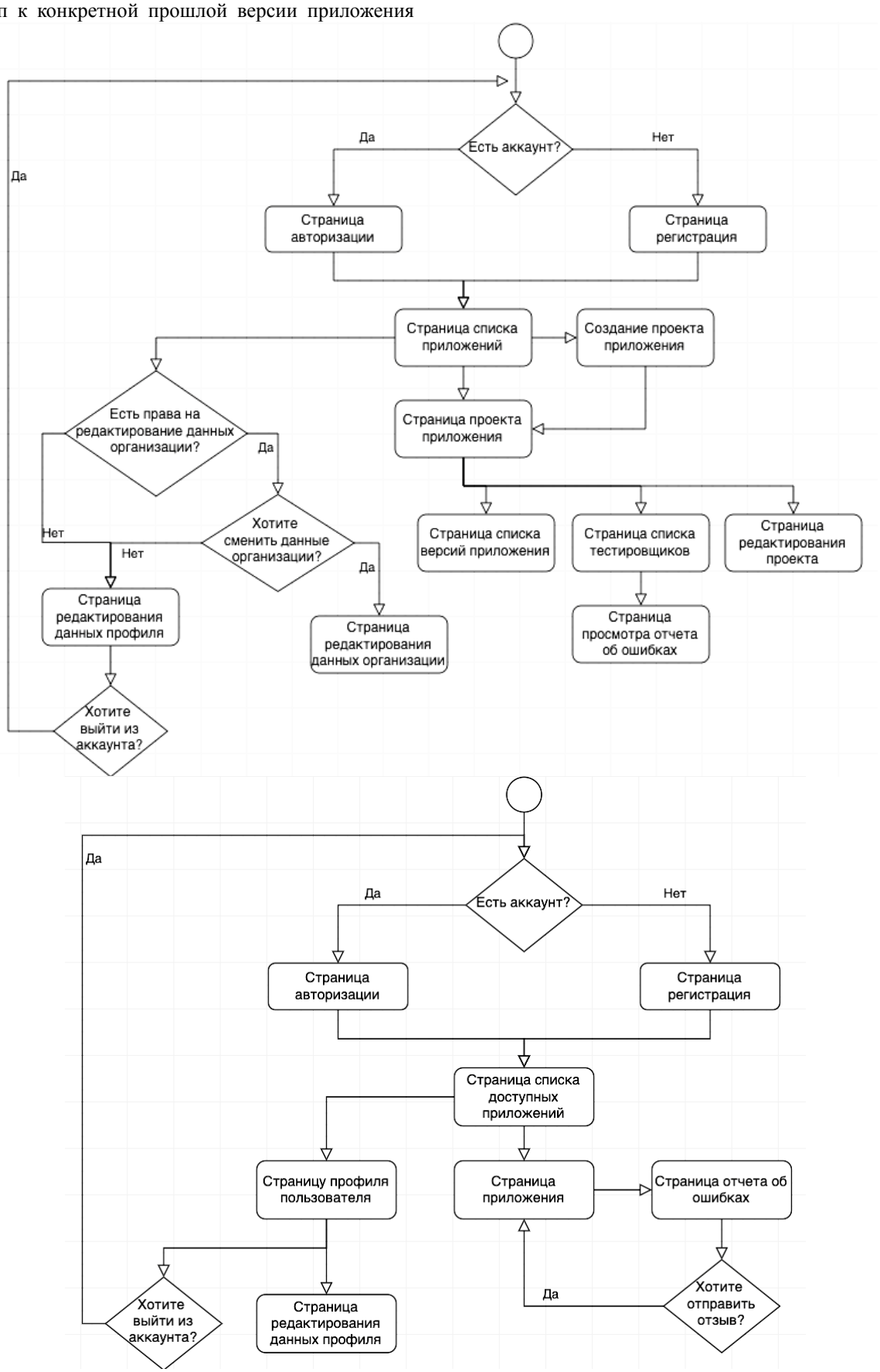


Рис. 1. Диаграммы активностей для тестировщика

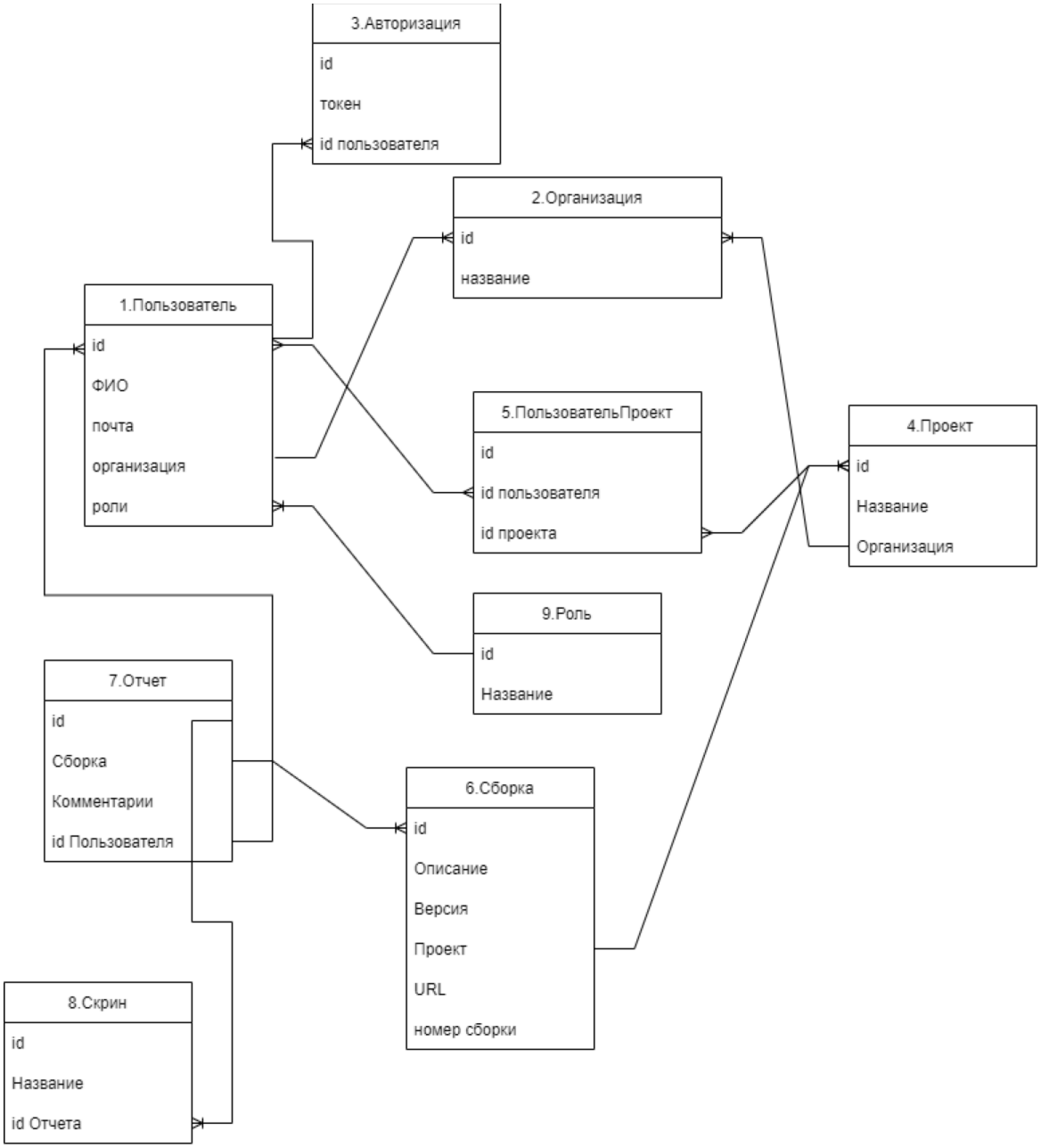


Рис. 2. Схема реализуемой базы данных

Стек разработки

Для разработки проекта необходимо написать мобильное приложение, веб-приложение, базу данных и REST API. Под каждую из этих задач выбран свой стек разработки.

Для веб-приложения лучшим решением является использование JavaScript, этот язык полностью интегрирован с HTML и способен полностью управлять веб-страницей. Разработка сайта с JavaScript является одной из самых быстрых и функционально предсказуемых. JavaScript используется в качестве клиентского языка программирования на 98,0% всех веб-сайтов.

Для разработки мобильного приложения хорошо подойдёт фреймворк Flutter на языке Dart. Это кроссплатформенное решение представляет все нужные функции для быстрой и качественной разработки мобильного приложения, такие как Hot Reload/Hot Restart и функционал Null Safety для избавления кода от непредвиденных ошибок с нулевыми переменными. В будущем выбор этого фреймворка может помочь с расширением функционала реализуемой службы бета-тестирования при работе с платформами Windows, macOS и Linux.

База данных будет располагаться на SQLite, а REST API реализуется на платформе для использования JavaScript на стороне сервера Node.js. Такой выбор обусловлен простотой в использовании и реализации.

Сравнение с аналогами

Для сравнения реализуемого продукта с аналогами была создана таблица с выделенными конкурентами.

Во всех найденных конкурентных приложениях отсутствует удобный и интуитивно понятный интерфейс, способный повлиять на желание использования продукта потенциальными клиентами.

Также следует отметить тот факт, что часть из представленных сервисов либо не поддерживается разработчиками и, как следует из этого, устаревает. Другая же часть приложений не поддерживается на территории Российской Федерации.

Сравнение с конкурентами

Сервис	Удобство использования	Доступ в РФ	Дизайн	Поддержка
HockeyApp (App Center SDK)	–	+	–	+
Apphance	–	–	–	+
DeployGate	–	+	–	+
UberTesters	+	+	–	–
AppBlade	–	–	–	–
Разрабатываемая система	+	+	+	+

Только один из найденных конкурентов имеет схожий удобный функционал, напоминающий реализуемый нами сервис, однако он не соответствует другим критериям, выписанным в таблицу.

Следует заметить, что некоторая часть из представленных приложений используется для автоматизированного тестирования, а не ручного. Следовательно, такие сервисы только отчасти можно назвать конкурирующими.

Первое время наш сервис тоже будет иметь недостатки, такие как ограниченность платформ, под которые доступен сервис, ограниченность памяти в базе данных для одной организации, урезанный функционал. Все эти проблемы вытекают из необходимости делать минимально жизнеспособный функционал продукта для быстрого внедрения в общее пользование. Впоследствии наша команда планирует избавиться от этих недостатков как можно скорее.

Заключение

В данной статье были рассмотрены актуальность и возможность создания системы тестирования мобильных приложений для платформы Android. Были выявлены основные моменты функционала и конкурентноспособные качества продукта.

Проект реализуется в рамках ГПО «Система тестирования мобильных приложений» АОИ-2206.

Литература

1. Usage statistics of JavaScript as client-side programming language on websites [Электронный ресурс]. – https://w3techs.com/technologies/details/cp-javascript

2. Тестирование приложений с помощью TestFlight [Электронный ресурс]. – https://testflight.apple.com/

3. 12 замен TestFlight: платформы мобильного бета-тестирования [Электронный ресурс]. – https://apptractor.ru/test/beta-testing/12-zamen-testflight-platformyi-mobilnogo-beta-testirovaniya.html

4. Проблемы безопасности Android-приложений: классификация и анализ [Электронный ресурс]. – https://habr.com/ru/company/pt/blog/332904/

5. Гид по ручному тестированию приложений: преимущества, этапы и методологии [Электронный ресурс]. – https://habr.com/ru/company/skillbox/blog/418889/

Лопатин Степан Евгеньевич
Студент каф. АОИ ТУСУРа
Эл. почта: gabblegaffe@gmail.com

Кульшин Роман Сергеевич
Студент, ассистент каф. АОИ
М.н.с. ЛООМИС ТУСУРа
Эл. почта: roman.s.kulshin@tusur.ru

Волокитин Геннадий Александрович
Студент, ассистент каф. АОИ
М.н.с. ЛООМИС ТУСУРа
Эл. почта: gennadii.volokitin@tusur.ru

УДК 004.4

Р.С. Кульшин, Г.А. Волокитин, А.А. Сидоров, П.В. Сенченко

Программное обеспечение для исследования алгоритмов рекомендательных систем

Рассматривается программное обеспечение для исследования алгоритмов рекомендательных систем. Описаны инструменты использование при разработке данного программного обеспечения, функциональные возможности, а также планы по развитию данного ПО.

Ключевые слова: рекомендация, программное обеспечение, набор данных, исследование.

В связи с увеличением объемов интернет-продаж и бурным ростом стриминговых сервисов, увеличился интерес к формированию персонализированных рекомендаций [1]. В основе персональной рекомендации находится рекомендательный алгоритм.

В сложившейся ситуации на помощь пользователю приходят информационные технологии. Огромные корпорации и маленькие компании повсеместно начинают внедрять рекомендательные системы. Рекомендательные системы используются для повышения качества предлагаемых товаров пользователю, тем самым увеличивая прибыль компании. Наиболее качественные рекомендательные системы формируют именно персонализированные рекомендации, которые ориентированы на каждого отдельно взятого пользователя.

Высокая потребность компаний в рекомендательных системах привела к быстрому развитию и большой заинтересованности в алгоритмах формирования пользовательских рекомендаций.

Основы технологий формирования рекомендаций появились достаточно давно, в момент появления машинного обучения. Именно тогда были заложены основные математические принципы и алгоритмы. С тех пор произошло немало изменений, появилось большое количество гибридных подходов к формированию рекомендаций.

На текущий момент одними из наиболее распространенных являются алгоритмы, которые относятся к группе коллаборативной фильтрации [2]. Создание рекомендательного алгоритма является трудоемким процессом и требует его глубокой проработки, так как при формировании рекомендации учитывается множество параметров. Именно для задачи исследования и отладки рекомендательных алгоритмов было разработано программное обеспечение (ПО), представленное в данной работе.

Разработанное программное обеспечение, в первую очередь, ориентировано на исследование алгоритмов формирования рекомендаций, которые относятся к группе коллаборативной фильтрации.

Метод коллаборативной фильтрации, а также его гибриды и модификации наиболее распространены в области формирования персонализированных рекомендаций. Принцип работы метода основывается на предположении о консервативности его предпочтений (если пользователи одинаково оценили определенный товар, то с достаточно большой вероятностью пользователи схожим образом оценят и другие товары со схожими характеристиками).

Разработка программного обеспечения

Программное обеспечение для исследования алгоритмов рекомендательных систем разработано на языке – Python 3.9 [3]. Python – это язык программирования высокого уровня общего назначения с строгой динамической типизацией и автоматическим управлением памятью, основная его цель – это повышение эффективности разработчика, качества и читаемости кода, а также обеспечение переносимости написанных на нём программ.

Для реализации графического интерфейса пользователя использовался набор расширения для графического фреймворка Qt – PyQt5. PyQt является плагином расширения языка Python и имеет тот же функционал, что и стандартная библиотека Qt [4]. Средой для разработки графического интерфейса стал Qt Designer. Qt Designer – свободная кроссплатформенная среда для разработки графических интерфейсов пользователя для программ, использующих библиотеку Qt. Входит в состав Qt framework. Разработанный интерфейс сохраняется в файл с расширением ui, который подключается к создаваемой программе с помощью специальных методов библиотеки Qt.

Для поддержки возможности производить вычисления на графических процессорах использовалась библиотека CuPy. Это библиотека с открытым исходным кодом для вычислений с ускорением GPU на языке программирования Python. CuPy использует библиотеки инструментария CUDA, включая cuBLAS, cuRAND, cuSOLVER, cuSPARSE, cuFFT, cuDNN и NCCL, чтобы в полной мере использовать архитектуру GPU.

Для выполнения математических вычислений использовались следующие библиотеки. NumPy – это фундаментальный пакет для научных вычислений на языке программирования Python. Pandas – это быстрый, мощный, гибкий и простой в использовании инструмент анализа и обработки данных с открытым исходным кодом, построенный поверх языка программирования Python. Si-kit learn – бесплатная библиотека программного обеспечения для машинного обучения для языка программирования Python. Она включает различные алгоритмы классификации, регрессии и кластеризации, включая методы опорных векторов, случайные леса, повышение

градиента, k-средние и DBSCAN и предназначена для взаимодействия с числовыми и научными библиотеками Python numpy и scipy.

Также для масштабирования вычислений наиболее распространенных математических библиотек для языка Python (numpy, pandas, sci-kit lean) использовался унифицированный фреймворк на основе тензоров для крупномасштабных вычислений – mars.

Python является интерпретируемым языком программирования, и эта особенность использовалась для реализации расширяемости функционала. Конечному пользователю достаточно добавить свой рекомендательный алгоритм, написанный по заданным правилам, и он будет автоматически интегрирован в программу.

Интерфейс программного обеспечения

Интерфейс программного обеспечения представлен на рис. 1.

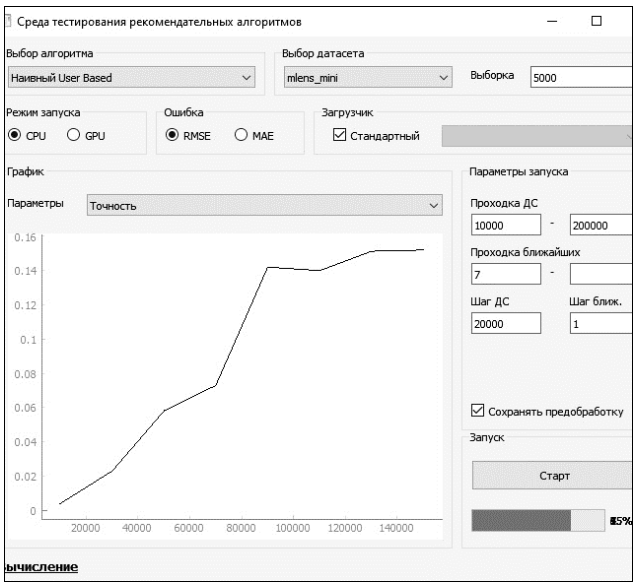


Рис. 1. Интерфейс ПО

Первым разделом интерфейса разработанного ПО является выбор алгоритма. Данный блок представлен выпадающим списком и отображает все доступные алгоритмы для использования. Алгоритмы описываются в файле и при запуске программы она автоматически получает все доступные методы.

Следующим элементом идет выбор набора данных. В данном элементе в выпадающем списке доступны все наборы данных, которые находятся в папке datasets в форматах csv, json и xml. А также поле ввода выборки – количество элементов набора данных, которые будут использоваться при исследовании.

Далее идет режим запуска. Этот элемент является переключением из двух режимов: вычисления с использованием CPU (центральный процессор) и вычисления с использованием GPU (графический процессор), если у какого-то алгоритма н реализовано вычисление с использованием GPU, данный переключатель блокируется. Реализация вычисления того или иного метода на GPU является проблематичной, так как требует векторизацию функций. Если векторизация не применялась, то результативность вычисления на GPU сильно снижается, приближаясь к CPU.

Затем идет выбор метода вычисления ошибки из RMSE (среднеквадратичное отклонение) и MAE (средняя абсолютная ошибка).

Следующим элементом идет загрузчик. Загрузчик – это класс-преобразователь входных наборов данных в табличную структуру данных – DataFrame. Также задачей загрузчика является разбиение входных данных на два набора: формирования рекомендаций и тестирования. Стандартный загрузчик использует три упорядоченных параметра: пользователь, объект, оценка. Если требуется работа с более сложными структурами данных или же их предобработка, имеется возможность написать свой загрузчик, и он автоматически будет доступен при запуске программы.

Далее идет настройка параметров запуска. Первым параметром является диапазон изменения набора данных, а также шаг изменения. Это требуется для того, чтобы оценить поведение алгоритма при изменении величины набора данных. Вторым параметров является диапазон изменения наиболее похожих пользователей, а также шаг изменения. Этот параметр необходим, чтобы оценить поведение алгоритма при изменении области схожести пользователя с другими пользователями. Последним параметром является чек-бокс сохранения данных, которые мы получили в ходе проведения исследования в файле формата .json. Данная функция необходима для восстановления результатов эксперимента.

Потом идет элемент графического отображение результатов эксперимента. В основе данного элемента лежит график, на котором изменения отображаются в реальном времени в зависимости от итерации. График имеет возможность масштабирования и скроллинга. Также в данном элементе имеется возможность выбора отображаемого параметра. На данный момент их два: время, затраченное на формирование рекомендации, и точность сформированной рекомендации.

Заключительным элементом пользовательского интерфейса является панель запуска и остановки эксперимента с панелью прогресса выполнения.

Заключение

В данной работе представлено программное обеспечение для исследования алгоритмов рекомендательных систем. В будущем функциональные возможности данного ПО будут расширены. Также рассматривается возможность реализовать клиент-серверную архитектуру для проведения исследований на более мощных вычислительных кластерах.

Работа выполнена в рамках ГЗ Наука Минобрнауки РФ. Проект FEWM-2020-0036 «Методическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде».

Литература

1. Vibhor Kant, Kamal K. Bharadwaj. Enhancing Recommendation Quality of Content-based Filtering through Collaborative Predictions and Fuzzy Similarity Measures // International Conference on Modeling, Optimization and Computing.

2. Xiaoyuan Su, Taghi M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques A Survey of Collaborative Filtering Techniques // Hindawi Publishing Corporation, Advances in Artificial Intelligence archive, USA. – 2009. – P. 1–19.

3. Официальная документация языка Python [Электронный ресурс]. – URL: https://www.python.com (дата обращения: 18.09.2022).

4. Официальная документация библиотеки PyQt [Электронный ресурс]. – URL: https://doc.qt.io/qtforpython (дата обращения: 18.09.2022).

Кульшин Роман Сергеевич
Студент, ассистент каф. АОИ, м.н.с. ЛООМИС ТУСУРа
Эл. почта: roman.s.kulshin@tusur.ru

Волокитин Геннадий Александрович
Студент, ассистент каф. АОИ, м.н.с. ЛООМИС ТУСУРа
Эл. почта: gennadii.volokitin@tusur.ru

Сидоров Анатолий Анатольевич
Зав. каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: anatolii.a.sidorov@tusur.ru

Сенченко Павел Васильевич
Доцент каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: pvs@tusur.ru

УДК 004.02

Р.С. Кульшин, И.Д. Тикшаев, А.А. Сидоров, П.В. Сенченко

Использование скоринговой модели оценки значимости действий пользователя для формирования персонализированных рекомендаций

Рассматривается использование скоринговой модели оценки значимости действия пользователя для формирования персонализированных рекомендаций. Описаны основные принципы построения скоринговой модели с променением логистической регрессии.

Ключевые слова: рекомендация, скоринг, логистическая регрессия, персонализация, коллаборативная фильтрация.

Вскоре после изобретения Всемирной паутины появилась система рекомендаций, а связанные с ней технологии были широко изучены и применены как научными кругами, так и промышленностью. В настоящее время рекомендательная система стала одним из самых успешных веб-приложений, ежедневно обслуживая миллиарды людей, рекомендуя различные виды контента, включая новостные ленты, видео, товары электронной коммерции, музыку, фильмы, книги, игры, друзей, работу и т.д.

Для обеспечения работоспособности рекомендательных систем необходимо, чтобы одновременно реализовывалось несколько процессов. Нужно понимать, как собирать данные о пользователях и как их интерпретировать. Также нужно владеть различными алгоритмами рекомендательных систем, чтобы для каждой ситуации можно было выбрать наиболее подходящее решение.

В основе любой рекомендательной системы лежит пользователь и объект [1]. На основании интересов пользователя рекомендательный алгоритм может предполагать те объекты, которые заинтересуют пользователя.

В данной работе рассматривается применение скоринговых моделей для метода коллаборативной фильтрации. Коллаборативная фильтрация, а также ее гибриды и модификации наиболее распространена в области формирования персонализированных рекомендаций. Принцип работы метода основывается на предположении о консервативности его предпочтений (если пользователи одинаково оценили определенный товар, то с достаточно большой вероятностью оценят и другие товары со схожими характеристиками).

Коллаборативная фильтрация (CF) и фильтрация на основе контента (CBF) – два основных метода, используемых в рекомендательных системах. Системы CF работают, сначала собирая пользовательские предпочтения для элементов в данной предметной области. Затем системы используют собранные данные для поиска пользователей с похожими профилями и используют их рейтинги для прогнозирования элементов, которые могут заинтересовать конкретного пользователя. CBF – это альтернативный метод, который исходит из области поиска информации. Системы на основе контента полагаются на описание контента элементов (например, заголовок, автор, текстовое описание, параметры объекта и т.д.), чтобы найти такие объекты, похожие на элементы, которые интересуют пользователя. CBF, в основном, используется в предметных областях, где доступны характеристики и описание объектов.

В данной работе рассматривается применение скоринговой модели для формирования глубокого представления о заинтересованности пользователя объектом.

Сбор данных для дальнейшего использования в рекомендательных системах

Данные о пользователях и их поведении стоят в основе любой рекомендательной системы. Правильность их сборки и качество выборки напрямую будут влиять на результат работы рекомендательного алгоритма [2].

Чем больше данных о пользователе и его поведении можно собрать, тем более точным будет результат предсказания рекомендации при условии соблюдения качества этих данных. Но есть и ряд проблем, связанных с этим. Во-первых, увеличение объема собираемых данных потребует большего количества места в хранилище данных. Другая проблема заключается в повышении требований к вычислительным ресурсам. Для достижения максимальной эффективности требуется сохранять баланс между объёмом собираемых данных и желаемым результатом.

Скоринговая модель

Скоринговая модель – это эвристическая модель рейтингов, которая, основываясь на предположении, что различные объекты со схожими показателями ведут себя одинаково, позволяет классифицировать эти объекты по группам. Область применения скоринга довольно-таки широка – это и страховые системы, маркетинговые платформы, операторы сотовой связи, но традиционной сферой применения считается банковская [3]. Для того чтобы снизить риски и минимизировать издержки, банки прибегают к использованию нескольких основных видов кредитного скоринга.

Application-скоринг. Самый популярный и простой вид скоринга. С помощью специальной балль-

ной системы позволяет оценить кредитоспособность заемщика.

Collection-скоринг. Данный вид позволяет найти оптимальный вариант работы с просроченной задолженностью и используется только к клиентам с подмоченной кредитной историей.

Fraud-скоринг. Специальный инструмент для распознания мошенников.

Behavioral-скоринг. Скоринговая модель, позволяющая спрогнозировать финансовое благополучие заемщика, и уже на основе полученных данных подобрать лучшую программу кредитования.

Скоринговая модель поможет определить отношение определенного пользователя к определенному объекту, опираясь на его действия в информационной системе. Из гибкого набора данных формируется предсказание о том, входит ли данный товар в зону интересов пользователя.

Скоринговую модель можно разработать на основе одного из двух существующих методов: первый осуществляется через анализ статистических данных, второй – на экспертной оценке. Зачастую используется первый метод, так как он более надежный и точный, но для него необходимо иметь желательно полную базу данных. Из-за того, что метод экспертной оценки строится на опыте специалиста (эксперта), то в таком случае возникают проблемы, связанные с выявлением факторов, сложностью разработки и неполнотой множества правил.

Применение скоринговой модели для оценки действий пользователя

В классических разновидностях алгоритмов коллаборативной фильтрации за основу берется только один параметр. Как правило этим параметром является рейтинговая оценка, проставленная пользователем определенному объекту. Именно введение дополнительных критериев рекомендации является затруднением при использовании алгоритмов коллаборативной фильтрации. Используя скоринговую модель, можно осуществлять свертку данных разных действий пользователя на целевой странице и определять его заинтересованность по отношению к объекту. После этого именно результат работы скоринговой модели ставится в основу вычисления алгоритма коллаборативной фильтрации.

Данная предобработка расширит количество критериев, которые будут учитываться при формировании персонализированной рекомендации определенному пользователю без изменения самого рекомендательного алгоритма.

Для построения скоринговой модели потребуется собрать изначальный набор данных о действиях пользователей на целевой странице объекта. Для примера можно рассмотреть возможные действия пользователя на целевой странице товара в «Интернет-магазине DNS», которые могут нести ценность для оценки заинтересованности.

Список действий:

1) покупка товара;
2) добавление товара в корзину;
3) добавление товара в «избранное»;
4) переход на целевую страницу;
5) просмотр подробного описания;
6) выставление рейтинговой оценки товару;
7) написание отзыва товару;
8) просмотр отзыва товару;
9) добавление товара в сравнение;
10) нажатие на кнопку поделиться в социальных сетях.

Для решения данной задачи будет использоваться метод логистической регрессии.

Логистическая регрессия – это подвид множественной регрессии, который позволяет установить зависимость между несколькими независимыми переменными и зависимой. В большинстве случаев результат логического анализа на основе величинных переменных принимает не точное значение (0 или 1), а значение в промежутке от 0 до 1 [4]. Поэтому полученные значения в результате выполнения логистической регрессии можно понимать как вероятность того, что данный результат может быть отнесен к какому-либо из классов.

Например, если результат, полученный в ходе логистической регрессии, близок к единице, то это говорит о том, что данный объект входит в зону интересов определенного пользователя, а значит, может быть рекомендован и другим пользователям, которые похожи на него. Результат, который показал значение ближе к нулю, соответственно, не входит в область интересов пользователя. Значения, которые получились в середине интервала, не дают нам полного понимания об отношении пользователя к данному объекту, но все равно могут использоваться в алгоритме коллаборативной фильтрации.

Заключение

В данной работе предлагается использование скоринговой модели с применением логистической регрессии. В дальнейшем планируется реализация скоринговой модели для оценки значимости действий пользователей с дальнейшим использованием в качестве свертки данных для алгоритма коллаборативной фильтрации.

Работа выполнена в рамках ГЗ «Наука» Минобрнауки РФ. Проект FEWM-2020-0036 «Методическое и инструментальное обеспечение принятия решений в задачах управления социально-экономическими системами и процессами в гетерогенной информационной среде».

Литература

1. Vibhor Kant, Kamal K. Bharadwaj. Enhancing Recommendation Quality of Content-based Filtering through Collaborative Predictions and Fuzzy Similarity Measures // Optimization and Computing: International Conference on Modeling.

2. Xiaoyuan Su, Taghi M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques A Survey of Collaborative Filtering Techniques // Hindawi Publishing Corporation, Advances in Artificial Intelligence archive, USA. – 2009. – P. 1–19.

3. Что такое кредитный скоринг? / Портал знаний StatSoft [Электронный ресурс]. – Режим доступа: http://statsoft.ru/solutions/tasks/scoring/ свободный.

4. Tolles J., Meurer W.J. Logistic Regression Relating Patient Characteristics to Outcomes // JAMA. – 2016. – Vol. 316 (5). – P. 533–4. DOI:10.1001/jama.2016.7653. ISSN 0098-7484. OCLC 6823603312. PMID 27483067.

Кульшин Роман Сергеевич
Студент, ассистент каф. АОИ, м.н.с. ЛООМИС ТУСУРа
Эл. почта: roman.s.kulshin@tusur.ru

Тикшаев Иван Денисович
Ассистент каф. АОИ ТУСУРа

Сидоров Анатолий Анатольевич
Зав. каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: anatolii.a.sidorov@tusur.ru

Сенченко Павел Васильевич
Доцент каф. АОИ, вед. н.с. ЛООМИС ТУСУРа
Эл. почта: pvs@tusur.ru

Секция 17

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБУЧЕНИИ

Председатели секции – ***Дмитриев Вячеслав Михайлович****, профессор каф. КСУП, д.т.н.;*
Ганджа Тарас Викторович*, профессор каф. КСУП, д.т.н.*

УДК 004.031.42

В.К. Шабалин, Д.А. Чебыкин, Д.Д. Пимонов, И.В. Климов, С.А. Краснов

Структура словаря веб-приложения для изучения японского языка

Рассматривается разработка мобильного и веб-приложения для изучения японского языка. На текущем этапе разработки был реализован прототип приложения, включающий тестовые тренировочные модули.
Ключевые слова: японский язык, веб-приложение, JavaScript, Golang, JSON, MySQL, веб-сокеты.

В современном мире знание иностранного языка является неоспоримым преимуществом. Понимание иностранных языков позволяет получать информацию из различных источников в различных сферах жизни. В наши дни существует огромное количество способов изучения иностранных языков. Для их эффективного изучения необходимо одновременно текстовое, графическое и звуковое восприятие информации, что могут предложить некоторые приложения.

На ресурсах Google Play [https://play.google.com] и в appstore можно найти достаточно большое количество приложений для изучения японского языка. Достаточно подробный список можно найти на сайте [https://lhlib.ru/bestjapanese35].

Целью данного проекта является разработка веб-приложения для комплексного изучения японского языка.

Задачами проекта являются: разработка и реализация структуры базы данных, разработка серверной части приложения, разработка клиентской мобильной части приложения и заполнение базы данных учебно-практическим материалом.

Несмотря на это многообразие приложений и информационных ресурсов, ни один из них не является самодостаточным, вследствие чего у изучающих язык возникают проблемы с полноценным изучением языка. В отечественных аналогах этот язык в целом недостаточно представлен [1], и ресурса, предоставляющего комплексный подход изучения всех аспектов языка для русскоговорящего пользователя на данный момент, не существует, но при этом востребованность такого ресурса присутствует. На предыдущих этапах проекта был проведён конкурентный анализ, выполнены макеты для страниц авторизации и личного профиля и экспериментально исследованы варианты способов анализа речи для оценки произношения [2].

На ранних стадиях текущего этапа нами были рассмотрены варианты создания структуры базы данных через информационно-поисковый тезаурус для максимально гибкого и быстрого чтения информации базы данных.

Реализация информационно-поискового тезауруса требует наличия опыта в проектировании сложных баз данных и больших трудозатрат, поэтому нами была выбрана упрощенная реализация многоязыкового приложения, позволяющего читать, запрашивать и редактировать данные для разных языков. На рис. 1 представлена упрощённая структура базы данных.

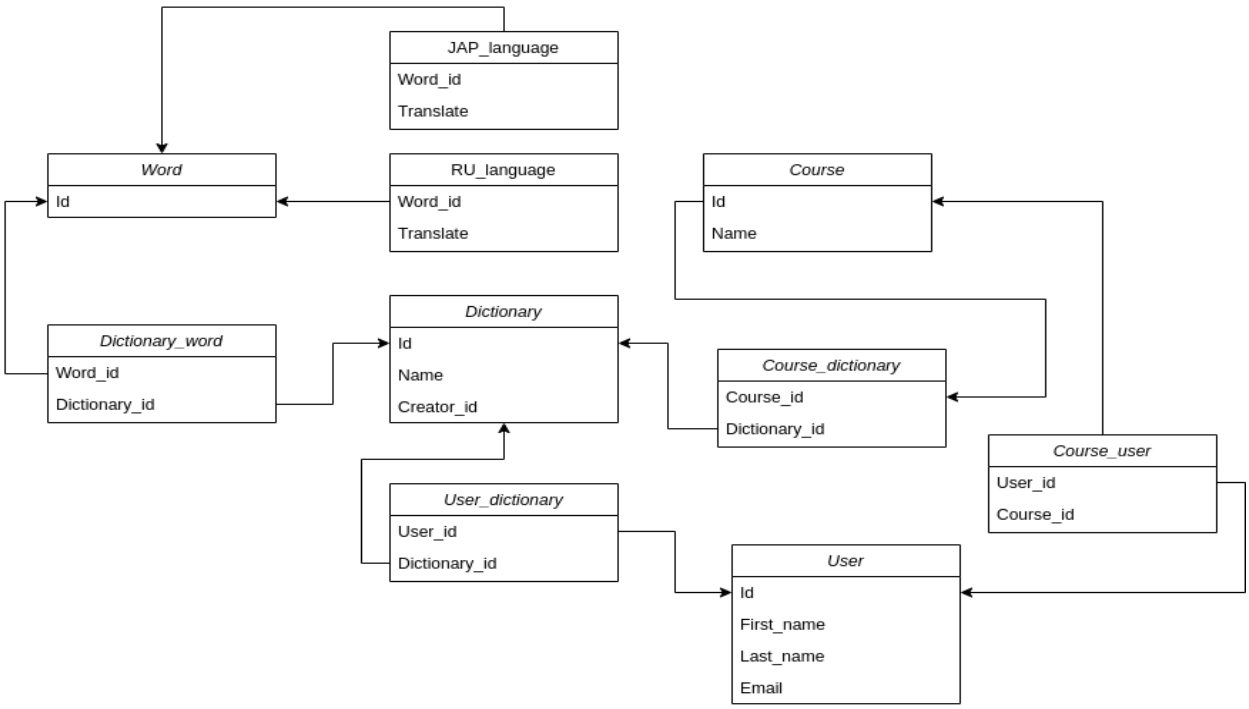


Рис. 1. Упрощённая структура базы данных

При продолжении разработки основной уклон был сделан на проработку базы данных системы. Структура базы данных должна поддерживать запросы слов на разных языках и давать возможность встраивать эти слова в словари.

Также рассматривается возможность расширить словарь для контекстного перевода всевозможных лингвистических конструкций. В нём будут храниться данные не только о дословном переводе слов, но и о всевозможных вариантах их перевода в зависимости от контекста. Это позволит переводить полноценные предложения на ходу, при этом не обозначая перевод для каждого отдельного предложения, что упростит процесс написания уроков для преподавателей и обычных пользователей и улучшит степень понимания обучающимися читаемого ими текста. Но у этой концепции расширения словаря есть и минусы – она значительно сложнее обычного дословного словаря, и при небольшом количестве уроков она будет приносить лишь неоправданное усложнение конструкции сервиса. Также у этого подхода есть ограничения, поскольку невозможно заранее предусмотреть все возможные контексты и неизбежно придётся в определённых ситуациях вводить контекстные переводы вручную. Вследствие вышесказанного наша команда не приняла окончательное решение о реализации этого расширения словаря.

Ещё одним вариантом решения этой проблемы может быть обращение к сторонним сервисам. Наш сервис сконцентрирован на изучении языка, а не предоставлении перевода, и для упрощения работы и освобождения ресурсов для решения других задач имеет смысл использовать сторонние сервисы контекстного перевода.

Словари в нашей системе могут быть двух видов. На рис. 2 представлен пример отображения словаря в мобильном приложении.

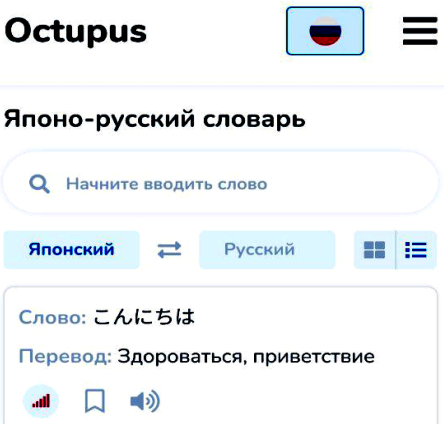


Рис. 2. Отображение словаря

Системный словарь – это единый словарь для всех пользователей, изменения в который могут вносить исключительно администраторы системы и который является основой для формирования пользовательских словарей. В него будут заноситься новые слова по рекомендации от экспертов и в случае, если большое количество пользователей посчитает это необходимым.

Пользовательские словари – это словари, формируемые из системных и администрируемые самими пользователями. Пользователи могут добавлять, изменять, получать или удалять данные в таких словарях. Такие словари предназначены для того, чтобы пользователи могли хранить там слова, которые они на данный момент изучают, что упростит процесс повторения и повысит эффективность освоения материала.

Мини-игры – интерактивные модули, предлагающие уникальный пользовательский опыт для учеников. Пользователи системы могут встраивать в другие модули разные игры, такие как «Тетрис» или «Крестики-нолики», предоставляющие закрепление учебного материала в игровой форме.

Тренировки – это модули, предназначенные для изучения отдельных аспектов языка в интерактивной форме.

Учебники – это сборники правил и различных учебных материалов, создаваемые пользователями системы. В данные сущности можно встраивать различные словари, мини-игры и медиафайлы для более полного погружения студентов в учебный процесс.

Пользователи системы могут создавать курсы, в которых они будут проводить занятия и вести учет успеваемости. В курсы могут встраиваться словари, мини-игры, учебники, медиафайлы в виде оформленных виджетов, которые позволят пользователям собирать курс как конструктор. Такой подход создания курсов позволяет снизить нагрузку на разработчиков и упростить создание учебных материалов для преподавателей.

Клиентская часть системы реализуется с помощью следующих технологий: HTML, препроцессор SCSS, язык программирования TypeScript, фреймворк Angular 14, библиотека Angular Material, RXJS и MomentJS. Для создания мобильной версии приложения используется язык программирования TypeScript и фреймворк React Native.

Для реализации серверной части приложения используется язык программирования Golang. В качестве системы управления базами данных используется PostreSQL. Для передачи данных по сети используется текстовый формат данных JSON [3].

Заключение

Результатом работы является прототип мобильного приложения, который включает в себя тестовые тренажёры, а также спроектированная структура базы данных. В текущем семестре мы планируем реализовать базу данных на серверной части проекта.

Литература

1. Семь лучших приложений для изучения иностранных языков [Электронный ресурс]. – Режим доступа: https://www.klerk.ru/boss/articles/394728/, свободный (дата обращения: 22.02.2022).

2. Шабалин В.К. Веб-приложение для изучения японского языка / В.К. Шабалин, Д.А. Чебыкин, Д.Д. Пи-

монов, И.В. Климов, С.А. Краснов // Научная сессия ТУСУР–2022: матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых, Томск, 18–20 мая 2022 г.: в 3 ч. – Томск: В-Спектр, 2022. – Ч. 2. – С. 24–26.

3. Клиент-серверная архитектура [Электронный ресурс]. – Режим доступа: https://developer.mozilla.org/ru/docs/Learn/Server-side/First_steps/Client-Server_overview, свободный (дата обращения: 23.02.2022).

Шабалин Валерий Константинович
Студент каф. АСУ ТУСУРа
Эл. почта: shabalinvalera.ts@gmail.com

Чебыкин Дмитрий Алексеевич
Студент каф. АСУ ТУСУРа
Эл. почта: chebykin.d.439-2@e.tusur.ru

Пимонов Даниил Дмитриевич
Студент каф. АСУ ТУСУРа
Эл. почта: pimonov.daniil@bk.ru

Климов Иван Васильевич
Студент каф. АСУ ТУСУРа
Эл. почта: climov.van@yandex.ru

Краснов Сергей Александрович
Студент каф. АСУ ТУСУРа
Эл. почта: awm2503@gmail.com

УДК 004.94

А.С. Болденков, Т.В. Ганджа

Применение дистанционных виртуальных лабораторий в обучении

Рассматривается применение дистанционных виртуальных лабораторий в процессе обучения студентов и сотрудников предприятий. Поставлены проблема и задача по анализу применения виртуальных инструментов для обучения.

Ключевые слова: дистанционная виртуальная лаборатория, виртуальный прибор, обучение, СМ МАРС.

На сегодняшний день активно используются программные инструменты для обучения студентов техническим специальностям. Такие инструменты позволяют погрузить ученика в область, в которой ему предстоит работать в будущем. Применяются они для построения различных моделей электрических цепей, технологических процессов, бизнес-процессов и т.д.

Научным коллективом под управлением профессора В.М. Дмитриева был разработан программный продукт среда моделирования МАРС (СМ МАРС), позволяющий создавать модели реальных физических процессов, проводить реально-виртуальные и виртуальные эксперименты, выполнять лабораторные работы и создавать виртуальные приборы [1, 2]. ПО хорошо зарекомендовало себя как хороший инструмент в процессе обучения, однако есть и свои недостатки:

1. Установка на рабочий ПК.

Для работы с ПО пользователь должен установить СМ МАРС на компьютер, на котором планируется проводить моделирование, т.е. весь процесс происходит локально без сетевого взаимодействия, все результаты сохраняются в единый файл на рабочем ПК.

2. Один пользователь.

Так как на текущий момент СМ МАРС не подразумевает никакого сетевого взаимодействия, то весь процесс работы и просмотра результатов доступен с одного ПК, на котором выполнялась работа.

Для решения указанных недостатков необходимо разработать систему с клиент-серверной архитектурой с возможностью многопользовательского использования со следующими характеристиками:

- Расчет моделей в сетевом вычислительном кластере (облаке).
- Наличие удобного web-интерфейса для работы с моделями и их конфигурирования.
- Многопользовательский доступ к процессу моделирования и анализу вычислений.
- Динамическое изменение параметров модели при расчетах.
- Параллельный запуск вычислений моделей.
- Программный интерфейс взаимодействия (API) для различных прикладных тренажеров и приложений.
- Просмотр результатов измерений в реальном времени.

Для передачи данных предлагается использовать популярный в настоящее время протокол WebSocket. Для начала стоит дать определение нашему протоколу. WebSocket – это независимый протокол, основанный на протоколе TCP и предназначенный для обмена сообщениями между браузером и веб-сервером в режиме реального времени. Таким образом, изначально синхронный протокол HTTP, построенный на модели «запрос – ответ», становится полностью асинхронным и симметричным. При использовании WebSocket'a нет клиента и сервера с фиксированными ролями, а есть два равноправных участника обмена данными.

На рис. 1 представлена функциональная схема сетевой работы СМ МАРС, на которой изображена клиент-серверная архитектура системы.

В данной схеме за роль сервера отвечает непосредственно СМ МАРС, главной задачей которого является обеспечивать обмен данными с клиентским приложением. В качестве базы данных (далее – БД) может быть использована любая реляционная БД. Доступ должен быть как к удаленным БД, так и к БД, использующим файлы в качестве хранилищ данных (MS Access, Oracle Database, MySQL, PostgreSQL, MS SQL и др.), т.е. данные, которые хранятся на локальном компьютере [3].

Для начала передачи данных клиент делает запрос, который обрабатывается и направляется в виде отдельного сообщения. Канал должен быть двойным, чтобы обмен данными происходил в обе стороны. Все сообщения будут передаваться в текстовом формате JSON, основанном на JavaScript. Основными преимуществами данного формата являются:

- компактность;
- легкость чтения (как человеком, так и компьютером);
- легкость преобразования в структуру данных для большинства языков программирования.

После получения запрос обрабатывается СМ МАРС и для завершения установления связи отправляется ответ. Далее МАРС направляет запрашиваемые данные обратно клиенту. С этого момента любая из сторон может отправлять данные в любое время. Таким образом, клиент будет отображать результаты моделирования прямо на экране пользователя. Такой подход позволяет подключаться нескольким пользователям для получения необходимых данных в реальном времени.

Для работы без доступа к серверу система должна предусматривать работу в offline-формате. То есть пользователь сможет локально проводить моделирование и эксперименты с использованием тех компонентов, которые включены в веб-тренажёр.

Потенциал дистанционной виртуальной лаборатории достаточно высок и востребован, так как поможет проводить обучение студентов и сотрудников на более качественном уровне, закрепляя теоретические знания на практических примерах, отражающих реальные процессы.

Основная суть работы с виртуальными лабораториями заключается в эмуляции действий, которые предстоит выполнять студенту в реальной лаборатории, т.е. установку сопротивления, измерение напряжения и т.д. Далее пользователь может проанализировать полученные результаты, провести дополнительные эксперименты с изменёнными значениями, сохранение измерений и вычислений с возможностью повторного их использования в дальнейшем [4].

На рис. 2 отображена схема процесса обучения с использованием виртуальных тренажеров.

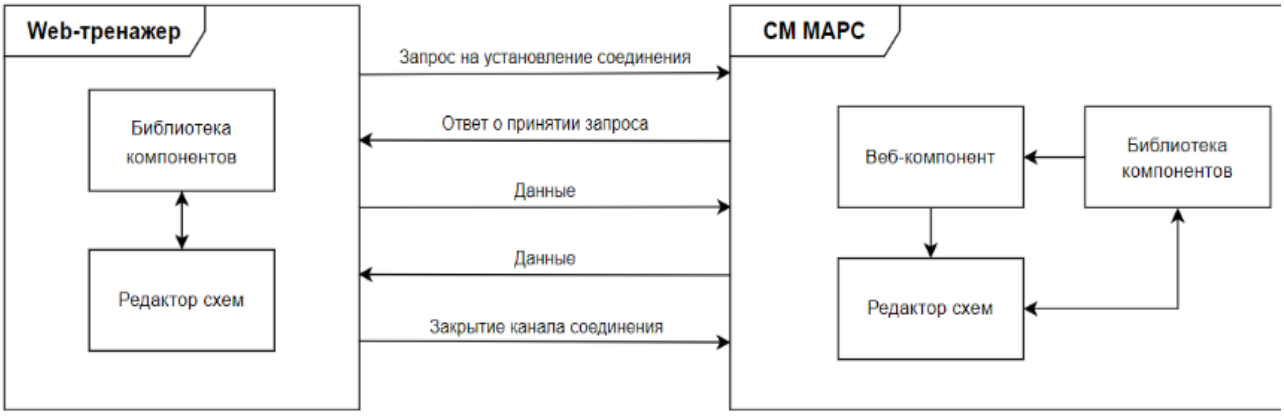


Рис. 1. Структурно-функциональная схема

Рис. 2. Процесс обучения

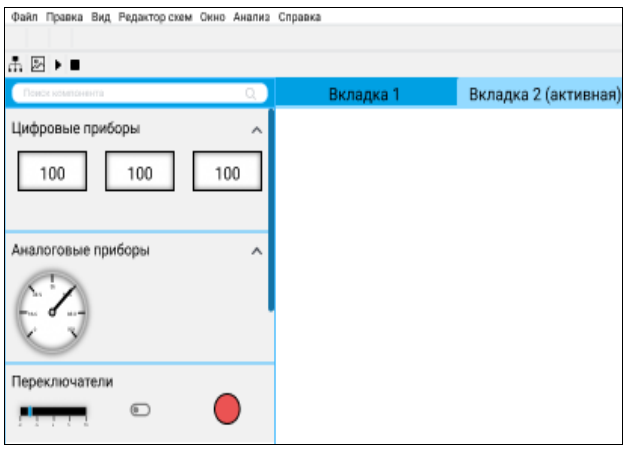


Рис. 3. Графический интерфейс

Далее рассмотрим, как может выглядеть web-интерфейс такой лаборатории. На рис. 3 представлена графический интерфейс пользователя дистанционной виртуальной лаборатории.

Сверху находится основная панель инструментов, позволяющая переключать слои, запускать и останавливать эксперимент и т.д.

Слева располагается дерево компонентов, которое хранит в себе всю базу созданных компонентов, из которых строится необходимая схема. База может постоянно пополняться, что расширяет круг дисциплин, в которых может применяться данный инструмент в процессе обучения.

Рассмотрим принцип построения схемы. Для начала необходимо выложить все необходимые для

эксперимента компоненты на рабочую область, просто перетащив их. Далее необходимо переключиться на логический слой, на котором автоматически добавились компоненты, но в схематичном представлении. На логическом слое следует правильно выстроить потоки передачи данных. В нашем случае на рабочей области находится компонент «Генератор», который будет генерировать заданное пользователем числовое значение и отправлять в другие подключенные компоненты. Для отображения отправляемых значений к генератору на логическом слое подключается компонент «Стрелочник» (рис. 4).

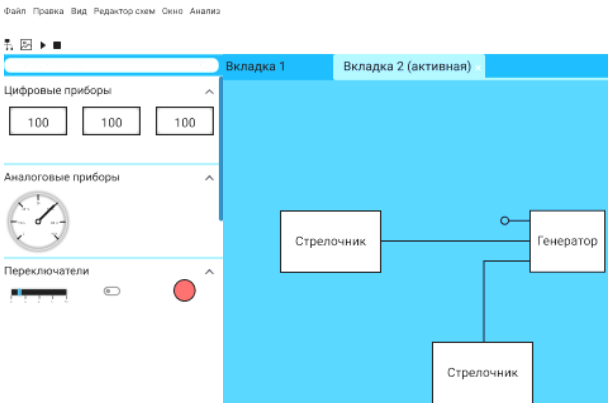


Рис. 4. Потоки данных

После соединения всех потоков данных на логическом слое следует переключиться обратно на визуальный слой. На визуальном слое пользователь располагает компоненты в удобном для себя формате. Затем он выставляет необходимые значения параметров и запускает эксперимент с помощью кнопки «Запуск» (рис. 5).

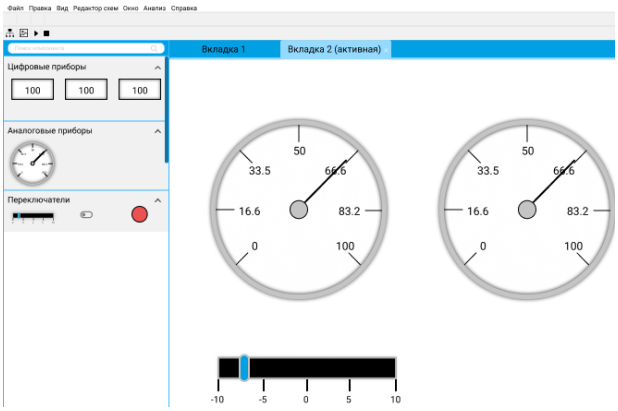


Рис. 5. Визуальный слой

В настоящее время на рынке отсутствует интерактивная дистанционная система, которая бы позволяла проводить практические и лабораторные работы, находясь в любом месте, без установки какого-либо ПО, что является большим недостатком, так как современная тенденция показывает, что многие люди хотят заниматься самообразованием. Для учебных заведений данная система помогла бы решить ряд вопросов по эффективности обучения студентов заочного обучения, студентов, у которых нет возможности посещать занятия по каким-либо причинам, например, по болезни, а также в эпидемиологических условиях, когда учебные заведения вынуждены перенести обучение в дистанционный формат [5, 6].

Использование таких технологий в сфере обучения поможет значительно сократить финансовые издержки, например, оборудование отдельной лаборатории или стенда для проведения занятий у студентов. Виртуальные лаборатории могут заменить большинство физических стендов, необходимо только заполнить библиотеку всеми необходимыми компонентами, с помощью которых будет составляться рабочая схема.

Такие лаборатории также удобны в случаях, когда эксперимент тяжело повторим или неповторим вовсе, ненагляден и т.д. В таких случаях наличие виртуальных лабораторий/тренажёров является необходимостью.

Литература

1. МАРС – среда моделирования технических устройств и систем / В.М. Дмитриев, А.В. Шутенков, Т.Н. Зайченко, Т.В. Ганджа. – Томск: В-Спектр, 2011. – 278 с.

2. Дмитриев В.М. Редактор виртуальных инструментов и приборов / В.М. Дмитриев, Т.В. Ганджа, Т.Ю. Коротина // Приборы и системы. Управление. Контроль. Диагностика. – 2009. – № 6. – С. 19–24.

3. Ганджа Т.В. Архитектура многопользовательской среды моделирования / Т.В. Ганджа, В.М. Дмитриев, Т.Н. Зайченко, А.С. Букреев, А.М. Малышенко, А.А. Шилин // Доклады ТУСУР. – 2019. – № 4. – С. 44–49.

4. Дмитриев В.М. Компьютерное моделирование визуальных интерфейсов виртуальных инструментов и приборов / В.М. Дмитриев, Т.В. Ганджа, В.В. Ганджа, С.А. Панов // Научная визуализация. – 2016. – Т. 8, № 3. – С. 111–131.

5. Груздева М.Л., Смирнова Ж.В. Результаты внедрения модели управления самостоятельной работой обучающихся в образовательный процесс вуза // Вестник Мининского ун-та. – 2017. – № 1. – С. 12–23.

6. Костылев Д.С., Костылева Е.А., Кутепова Л.И. Организация информационной среды в системе дополнительного профессионального образования в условиях дистанционного обучения // Перспективы науки. – 2015. – № 4 (67). – С. 23–25.

Болденков Алексей Сергеевич
Аспирант каф. компьютерных систем в управлении и проектировании (КСУП) ТУСУРа
Эл. почта: Nimfador10@gmail.com

Ганджа Тарас Викторович
Д-р техн. наук, профессор каф. КСУП ТУСУРа
Эл. почта: gandgatv@gmail.com

УДК 331.36

Т.Н. Мосунова, В.А. Хамзина, Е.С. Костюк

Формирование процесса разработки корпоративных обучающих электронных тренажеров для сотрудников

Рассматривается процесс разработки корпоративных обучающих электронных тренажеров для сотрудников строительного магазина. Проведен анализ шагов по разработке тренажеров и выявлен порядок действий для разработчиков с сопутствующими выводами.

Ключевые слова: электронные тренажеры, корпоративное обучение, обучение сотрудников.

Рынок труда стремительно трансформируется, даже внутри одной сферы профессиональной деятельности со временем значительно изменяется спектр необходимых знаний, умений и навыков. Внедрение новых технологий и использование научно-технических достижений в профессиональной деятельности приводит к необходимости внедрения корпоративного обучения.

Кадры являются главной ценностью компании, поскольку без грамотных и компетентных сотрудников, их уровня подготовки и мотивации даже наличие инновационных технологий, современного оборудования и эффективной системы управления не гарантирует высокой результативности. Этот фактор мотивирует крупные компании вкладывать в обучение и развитие персонала от 1 до 5% годового оборота [1].

В современных условиях бизнес, который заинтересован в масштабировании, нуждается в эффективной централизованной системе управления знаниями [2]. Пандемия и вынужденная самоизоляция позволили по-новому взглянуть на обучение сотрудников и осознать удобство использования систем дистанционного обучения. Корпоративное обучение может быть направлено на изучение различных областей деятельности сотрудника, начиная с его основных функций и заканчивая обязанностями при возникновении опасных ситуаций.

На текущий момент многие российские организации рассматривают расходы, связанные с обучением персонала, как первостепенные и приоритетные, поскольку вложения в образование сотрудников являются инвестицией в будущее компании. Корпоративное обучение направлено не только на удовлетворение текущих потребностей компании в компетентных сотрудниках, но и на долгосрочное повышение качества подготовки сотрудников в будущем.

Особое внимание в организациях должно быть уделено выявлению потребности сотрудников в обучении, которая возникает в ответ на актуальные проблемы или может быть связана с реализацией планов корпорации [3].

Процесс разработки таких обучающих тренажеров должен учитывать специфику предприятия и тематику обучения, однако можно выделить правила и подходы, этапы в процессе разработки, которые могут быть применены к процессе разработки корпоративного обучения в любой сфере компании. Рассмотрим процесс разработки обучающих тренажеров с помощью ПО iSpring Suite на примере трех тренажеров. Один из них был разработан как общий корпоративный тренажер, предназначенный для всех сотрудников, а два других предназначены для конкретных категорий сотрудников – технических консультантов и кассиров.

iSpring Suite – это конструктор онлайн-курсов на основе презентаций Microsort PowerPoint. Надстройка позволяет создавать онлайн-курсы, тесты и тренажеры, монтировать видео, писать интерактивные статьи и превращать документы в электронные учебники [4].

Процесс разработки электронных обучающих тренажеров начинается задолго до того, как создаются интерактивности и появляется визуальная составляющая тренажера.

Первым этапом разработки является получение договоренности с заказчиком. В рассмотренном процессе заказчик тренажеров берет на себя ответственность за актуальность и теоретическое наполнение тренажера. По результатам проведенного первого этапа была выявлена необходимость выделения дополнительного этапа по получению теоретической информации. Для того чтобы обучающий контент был понятным любому проходящему тренажер, данные от заказчика проходят анализ и переработку, а также осуществляется поиск дополнительной необходимой информации.

Результаты этого этапа важно также согласовывать с заказчиком, поскольку специфичный контент в изначальной подаче может быть непонятным не только для будущих обучающихся, но и для разработчиков.

Так, на основании проведенной работы над тремя обучающими тренажерами были выявлены следующие шаги, необходимые в рамках первого этапа разработки:

1) анализ полученных исходных материалов от заказчика;

2) поиск дополнительных материалов;

3) уточнение информации, характерной для конкретной компании (филиала), для которой разрабатывается тренажер;

4) внесение исправлений в общий теоретический материал;

5) согласование общего теоретического материала, который будет дан в тренажере, с заказчиком.

Пункты 3–5 могут при необходимости быть произведены повторно.

При проведении данного этапа обнаружилась необходимость в приведении уточнений и введении разработчиков в основы профессиональной деятельности (консультанта, кассира), чтобы минимизировать в курсе сложные профессиональные термины и иметь возможность объяснить процесс так, чтобы его понял любой человек без специальной подготовки.

Срок данного этапа составляет 3–5 дней.

По данному этапу были сформированы некоторые выводы:

- получаемые исходные данные от заказчика могут иметь недостатки методического характера, которые могут быть исправлены разработчиками в процессе анализа и обработки данных;
- получаемые данные могут быть неполными, поэтому необходимы погружение разработчика в тему и критический взгляд на переданные компанией материалы.

Вторым этапом разработки тренажера является формирование плана и сценария. План позволяет повторно проанализировать имеющуюся информацию с точки зрения ее подачи в тренажере. Формы плана могут быть различными, поскольку план служит как внутренний документ для разработчика и не согласуется в таком виде с заказчиком. Чтобы тренажер был более понятным для обучающихся, необходимо, чтобы переходы от одного блока контента к другому были логически обоснованными.

Для этого разрабатывается сценарий тренажера. На этапе разработки сценария выбираются основные персонажи, которые будут участвовать в подаче информации. Персонажами могут выступать люди, животные и даже предметы, а также повествование может идти от лица электронного тренажера. На этапе написания сценария устанавливаются обобщенные отношения между обучающимся и персонажами, выбирается настроение подачи контента и подбираются формулировки для связи теоретических блоков в единый тренажер.

Сценарий является первой полноценной версией разрабатываемого тренажера, поэтому при его формировании важно учитывать, для какой целевой аудитории готовится обучающий тренажер, какова будет подача материала.

Также на данном этапе разрабатываются оценочные мероприятия в обобщенном или конкретизированном виде. Срок данного этапа составляет около 7 дней непрерывной работы разработчиков.

Результатом данного этапа является согласование первой версии сценария тренажера с заказчиком. Сценарий представляет собой последовательность экранов, которые будет видеть обучающийся при прохождении тренажера. Согласуемый сценарий уже отдаленно напоминает финальный результат, поскольку включает практически финальное текстовое наполнение.

По результатам данного этапа были сделаны следующие выводы:

- на этапе разработки сценария могут выявляться пробелы в теоретическом наполнении, поэтому на этапе согласования важно повторно задать вопросы по ним. Для полного погружения провести разработчикам экскурсию по рабочему месту сотрудников, для которых разрабатывался тренажер, что позволит приблизить обучение к реальному;
- при согласовании сценария важно подробно описать главного героя (героев), чтобы в случае несогласия заказчика иметь возможность доработать его (их) сразу и в дальнейшем вносить минимальные корректировки.

Когда предварительное согласование по сценарию завершено и стороны пришли к согласию, продолжается творческая деятельность разработчиков. Третий этап разработки можно определить как создание контента, он включает в себя следующую деятельность разработчиков:

- выбор стиля оформления тренажера (общая цветовая гамма, стиль элементов навигации, шрифт и т.д.);
- разработка визуального наполнения тренажера (рисунки, графики, предметы);
- отрисовка героев в ситуациях, обозначенных сценарием;
- оформление тестовых вопросов;
- продумывание и оформление интерактивных заданий и креативной подачи контента;
- настройка оценивания тестов;
- разработка дополнительных элементов.

Данный этап является самым трудозатратным для разработчиков, поэтому требуется нормировать время работы и грамотно его распределять, чтобы придерживаться установленных договором сроков. На данном этапе требуется внутреннее тестирование курса командой разработчиков, чтобы выявить возможные несоответствия с исходным теоретическим материалом, дефекты подачи информации или технические моменты, не позволяющие проходить курс. Так, например, при использовании плеера с ограниченной навигацией и хотя бы одной неправильно функционирующей кнопкой обучающийся может лишиться возможности дойти курс до конца. По результатам данного этапа были сделаны следующие выводы:

- на этапе тестирования и апробации необходимо выбирать плеер (проигрыватель тренажера, в котором он просматривается до публикации в системе обучения компании) с максимальными возможностями, чтобы ускорить процесс разработки и иметь возможность переходить по тренажеру вне зависимости от работы созданной разработчиком системы навигации;
- необходимо проведение внутренней проверки несколькими разработчиками, чтобы убедиться, что контент и управление тренажером будут понятны любому обучающемуся;
- необходимы исчерпывающие пояснения для разработчика, который отвечает за создание иллюстраций, поскольку некоторые процессы обладают

специфическими особенностями, которые необходимо корректно отразить в визуальной составляющей тренажера;

• вариации интерактивных экранов, предоставляемых iSpring Suite, ограничены, поэтому в некоторых местах целесообразно делать анимацию для тренажера средствами PowerPoint.

После внутреннего тестирования тренажер может быть направлен на согласование с заказчиком. К данному моменту тренажер должен быть финальной версией, то есть таким, каким его увидел бы обучающийся.

В зависимости от объема тренажера и сложности его элементов этап может занимать от семи дней.

При разработке на этом этапе были выявлены следующие замечания:

• для удобства согласования и проверки тренажера удобно предоставлять заказчику информацию для прохождения тестов;

• для удобства внесения исправлений в тренажер возможно разработать шаблон, который заказчик смог бы заполнить при обнаружении недочетов.

При разработке корпоративных обучающих тренажеров на примере первого тренажера были обнаружены недостатки в выстроенном процессе коммуникации и разработки, и на проектах следующих тренажеров были проведены работы по их минимизации. В дальнейшем планируется проведение еще одного этапа разработки, а именно сбора обратной связи как от заказчиков, так и от их сотрудников, прошедших обучение по разработанным тренажерам.

По результатам проведенной работы можно сделать следующие выводы:

1) разработка корпоративного обучения требует глубокой вовлеченности разработчиков, поскольку требуется оказывать не только услуги по разработке, но и услуги методического характера, чтобы сделать обучение понятным и полноценным. Чтобы разработчик был способен вносить коррективы в подачу материала, необходимо его сопровождение специалистами от компании, обладающими необходимыми навыками;

2) при разработке корпоративных обучений важно проводить процесс поэтапно и согласовывать каждое значительное изменение с заказчиком обучения, поскольку это может в дальнейшем помочь сократить время разработки;

3) на финальных этапах разработки корпоративных тренажеров необходимо комплексное тестирование с привлечением дополнительных разработчиков и экспертов со стороны заказчика, чтобы убедиться, что специфичный контент будет понятен обучающемуся без специальной предварительной подготовки (если иное не подразумевается тренажером).

Таким образом, был выстроен процесс разработки электронных тренажеров, учитывающий профессиональную специфику, а также современные тенденции, и были выявлены несколько этапов разработки, которые могут служить основой и планом действий для разработки корпоративных тренажеров любой компании.

Литература

1. Попов В.В. Инвестирование в человеческие ресурсы в условиях кризиса: преимущества корпоративного обучения // Современные тенденции в экономике и управлении: новый взгляд. – 2016. – № 45. – С. 143–149.

2. Коргина О.А. Корпоративное обучение как эффективный инструмент управления развитием персонала в организации // Вестник академии знаний. – 2021. – № 42 (1). – С. 191–196.

3. Захарова Д.С. Поддержание потребности в обучении персонала в системе корпоративного обучения в организации // Экономика, бизнес, инновации: сборник статей VIII Междунар. науч.-практ. конф.: в 2 ч. – Пенза, 2019. – С. 39–43.

4. iSpring Suite 11. Быстрый конструктор онлайн-курсов [Электронный ресурс]. – Режим доступа: https://www.ispring.ru/ispring-suite, свободный (дата обращения: 19.09.2022).

Мосунова Татьяна Николаевна
Студентка каф. КСУП ТУСУРа
Эл. почта: mosunova2013@mail.ru

Хамзина Венера Александровна
Студентка каф. КСУП ТУСУРа
Эл. почта: Santee220@gmail.com

Костюк Екатерина Сергеевна
Студентка каф. КСУП ТУСУРа
Эл. почта: katerina22kos@gmail.com

УДК 378.4: 004.42

А.А. Захарова, А.М. Аверьянова, К.Д. Глухих

Проблема автоматизации проведения нормоконтроля студенческих работ в вузах

Рассматривается актуальность создания автоматизированной системы управления процессом проведения нормоконтроля студенческих работ. Проведён анализ объёма проверяемых работ на примере направления «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем». Представлено описание функционала автоматизированной системы, обеспечивающей проверку студенческих работ на нормоконтроль и выявление ошибок в сданных работах. Автоматизированная система позволит повысить эффективность проверки студенческих работ и сократить временные затраты на проведение нормоконтроля в вузах.

Ключевые слова: нормоконтроль, учебная документация, автоматизированная система.

В настоящее время почти во всех сферах деятельности применяется проверка документации на соответствие межгосударственным, национальным и международным стандартам. На официальном сайте статистики представлено большое количество разнообразных стандартов: межгосударственных стандартов насчитывается 29 620, национальных 18 940, а международных – более 50 000 [1, 2].

Нормоконтроль – это проверка документации на соответствие определенным нормам, стандартам и правилам, которые устанавливаются нормативными документами. При его проведении оценивается правильность изложения и оформления работ с целью повышения качества документации.

Целью данной работы является показать актуальность разработки программного обеспечения для автоматизации процесса проверки учебной документации в процессе нормоконтроля.

К задачам можно отнести:

1. Выявить особенности процесса нормоконтроля в вузе и требования к проверке учебных работ в вузе.

2. Провести анализ трудозатрат на осуществление процесса нормоконтроля на примере одного из направлений обучения в ТУСУРе.

3. Рассмотреть существующие аналоги.

4. Представить основные функции автоматизированной системы для проведения нормоконтроля учебной документации.

Нормоконтроль учебной документации

Проверка учебных работ в вузе на нормоконтроль осуществляется нормоконтролером или другим ответственным лицом. Зачастую функции нормоконтролера возлагаются на преподавателей дисциплин, руководителей практик или выпускных квалификационных работ.

Проверка документации является ресурсозатратным процессом, требующим от нормоконтролера повторения однотипных операций по проверке соблюдения правил оформления заголовков, стилей, интервалов в тексте и других структурных элементов. Рассмотрим процесс проведения нормоконтроля на примере учебной документации. Нормоконтроль в учебной документации применяется для того, чтобы повысить качество подготовки выпускников, а также повысить качество и эффективность студенческих разработок [3].

К проверяемой учебной документации относятся:

– тематические рефераты;

– отчеты по лабораторным работам;

– курсовые работы / проекты;

– отчеты по всем видам практик, а именно: учебная, производственная, преддипломная;

– отчеты по результатам группового проектного обучения (ГПО);

– выпускные квалификационные работы бакалавров (бакалаврские работы);

– выпускные квалификационные работы специалистов (дипломные проекты или дипломные работы);

– выпускные квалификационные работы магистров (магистерские диссертации) [4].

Данные документы содержат в себе следующие разделы:

– титульный лист.

– задание;

– реферат (необязательный раздел учебной документации, кроме ВКР);

– оглавление / содержание;

– введение;

– основная часть;

– заключение;

– список использованных источников;

– приложения (необязательный раздел).

В целом весь документ проверяется на удобство чтения, наличие и правильность нумерации страниц, корректность размеров полей, логичность и грамотность изложения, затем проверяется каждый раздел по отдельности. В табл. 1 представлены проверяемые разделы учебной документации и уточняются требования проверки.

Схема проведения нормоконтроля учебной документации представлена на рис. 1, а общее содержание основных этапов и их выполнения в табл. 2 [5].

Помимо проверки перечисленной выше учебной документации, проверяются ещё и статьи, написанные студентами за время обучения.

Таблица 1

Проверяемые разделы и требования к их проверке

Раздел документа	Требования проверки
Титульный лист	Наличие необходимых подписей, правильность указанных дат и оформления
Задание	Наличие необходимых подписей, корректность дат и правильность заполнения
Реферат (при наличии)	Правильность оформления и составления реферата
Оглавление / содержание	Соответствие оглавления / содержания тексту работы и правильность его оформления
Введение	Корректность поставленных целей и задач работы
Основная часть	Правильность оформления рисунков, таблиц, формул, их нумерации и ссылок в тексте. Правильность сокращения слов и словосочетаний, а также корректное использование сокращений. Соответствие содержанию, а также проверка на выполнение поставленных целей и задач, описанных в задании
Заключение	Корректность сделанных выводов по работе
Список использованных источников	Правильность оформления использованных источников. Наличие и правильность оформления ссылок на источники в тексте работы
Приложения	Правильность оформления приложения. Корректность нумерации таблиц, формул, иллюстраций, а также ссылок на них

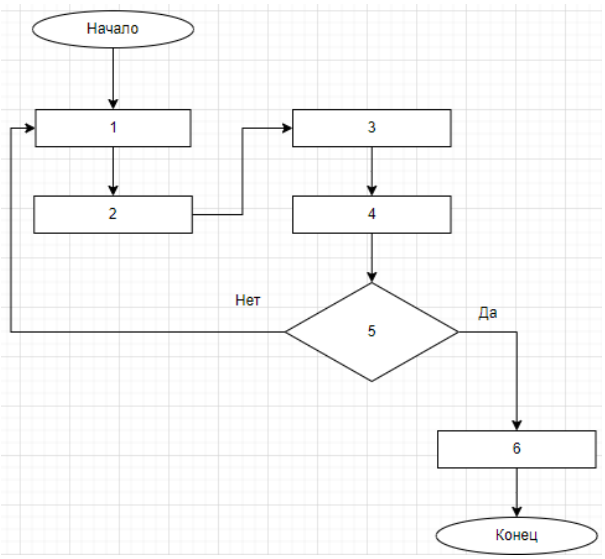


Рис. 1. Схема проведения нормоконтроля учебной документации

Таблица 2

Этапы процесса нормоконтроля учебной документации

Номер этапа и его название	Результат выполнения этапа
1	2
1.Разработка / доработка учебной документации студентом	Полный комплект учебной документации
2. Сдача / отправка учебной документации на нормоконтроль	Выявление соответствий или несоответствий в заявленном комплекте документации
3. Проведение нормоконтроля	Проверенный документ, в котором отображены несоответствия (в случае если они присутствуют) и даны рекомендации по их исправлению
4. Запись результатов проведения нормоконтроля	Оформление перечня замечаний и предложений
5. Решение о прохождении нормоконтроля	Решение о пересмотре учебной документации или ее соответствии требованиям
6. Отметка о сдаче учебной документации	Сданная учебная документация

Продолжение табл. 2

Анализ объёма проверяемых работ по направлению «Информатика и вычислительная техника», профиль «Программное обеспечение средств вычислительной техники и автоматизированных систем», 2018 г. набора

Всего проверяется пять видов учебной документации:

– лабораторные работы;

– курсовые работы;

– отчеты по практикам (учебная, производственная и преддипломная);

– отчеты по проектам группового проектного обучения (ГПО);

– выпускные квалификационные работы.

На рис. 2 представлены средние трудозатраты на проверку каждого вида учебной документации.

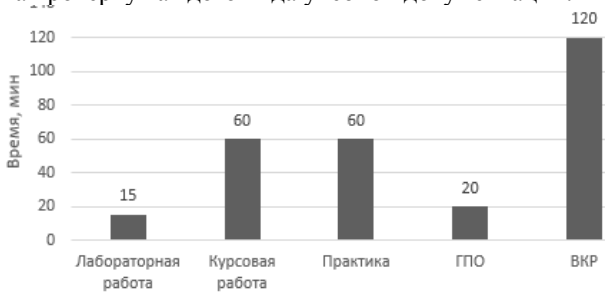


Рис. 2. Трудозатраты на проверку одной работы

Каждый студент по одной дисциплине выполняет в среднем пять лабораторных работ, в потоке обучается около 75 студентов. По одной дисциплине на одного студента преподаватель затрачивает на проверку нормоконтроля 1 ч 15 мин, что составляет на всех студентов 93 ч 45 мин. За весь курс бакалавриата студент выполняет 140 лабораторных работ по двадцати восьми дисциплинам, а проверка всех лабораторных работ занимает 2 625 ч.

Так же студент за весь период обучения выполняет обязательно две курсовые работы, на проверку одной курсовой работы затрачивается 1 ч, проверка работ всех студентов по двум дисциплинам занимает 150 ч.

За время обучения бакалавриата студенты обязательно проходят три практики: учебная практика, производственная и преддипломная практика, после каждой практики студенты пишут отчёт, примерное

затрачиваемое время на проверку отчётов составляет 225 ч.

Студенты выполняют проекты ГПО, в данной дисциплине также необходимо в конце каждого семестра оформлять отчёт по ГОСТу. Один проект выполняется 4 семестра. На кафедре АСУ реализуется 12 проектов. Отчет может выполняться несколькими студентами. Общее время на проверку составляет 48 ч.

Финальным завершением обучения каждого студента является написание выпускной квалификационной работы, это самый большой по объёму вид работы. На данный вид отчёта отводится на проверку каждой работы 2 ч, в среднем сотрудник затрачивает на проверку всех работ 150 ч.

В табл. 3 представлено итоговое количество времени, потраченного на проверку работ одного потока студентов по каждому виду учебной документации.

Таблица 3

Трудозатраты на проверку работ одного потока студентов

Вид работы	Среднее количество работ, выполняемых студентом (группой ГПО), шт.	Время на проверку одной работы, ч	Среднее количество студентов в потоке, чел./количество групп ГПО, шт.	Итоговое время, ч
Лабораторная	140	0,25	75	2 625
Курсовая	2	1,0	75	150
Практика	3	1,0	75	225
ГПО	4	1,0	12	48
ВКР	1	2,0	75	150
ИТОГО	10 998	–	–	3 198

В среднем за время обучения студентов по анализируемому учебному плану проверяется 10998 работ, затрачиваемое время на проверку составляет примерно 3 198 ч.

В табл. 3 представлено итоговое время, если студент сдал учебную документацию с первого раза, но стоит отметить, что большинство студентов оформляет документацию неправильно (в среднем студенты сдают работы примерно с третьего раза), что значительно увеличивает время на проверку работ.

Разработка автоматизированной системы

В настоящее время с помощью различных программ и систем можно улучшить и автоматизировать любой процесс для предоставления более комфортных условий качества жизни. Ранее показано, что на проведение нормоконтроля тратится большое количество времени, поэтому нами было принято решение реализовать автоматизированную систему управления процессом проведения нормоконтроля студенческих работ.

Целями создания автоматизированной системы являются снижение ресурсоемкости процесса нормоконтроля учебной документации, разгрузка преподавателей и других нормоконтролеров, повышение качества учебной документации [6].

На данный момент аналогов разрабатываемого программного продукта на рынке не существует. Были попытки разработки программы с помощью макросов, аналог разрабатывался с помощью Visual Basic for Application (VBA). В данном аналоге был разработан функционал проверки наименования шрифта, размера шрифта, выравнивания и начертания символов с первого по последний абзац. К минусам аналога стоит отнести, что нет удобного пользовательского интерфейса, а также проверки правильности установленных в тексте абзацных отступов, интервалов, размеров полей, оформления таблиц, списков, формул и изображений [7].

Предполагаемая система будет представлена в виде веб-приложения, основным языком программирования планируется использовать C#. Данный язык был выбран, потому что он является надежным, масштабируемым и достаточно простым, а также помогает разработчикам устранить проблему утечки памяти и облегчает поиск ошибок в коде.

Реализуемая программа должна будет проверять оформление документа по определенному загруженному стандарту, а также будет реализована функция редактирования документа. На данный момент нет библиотек, которые могли бы выполнять данные функции, поэтому будет реализована библиотека на net.Framework. Данная библиотека будет предоставлять возможность проверять:

– стиль текста: шрифт, размер шрифта, цвет текста, выравнивание текста, межстрочный интервал, отступы и др.;

– оформление рисунков;

– оформление таблиц: границы, отступы и т.д.;

– нумерацию страниц;

– формат сносок.

А также все эти элементы будут редактироваться в автоматическом режиме.

Заключение

В заключение хотелось бы отметить, что нормоконтроль является важной частью в проверке учебной документации, а также при проверке другой документации в различных сферах деятельности.

Анализ показывает, что на нормоконтроль уходит большое количество времени, в связи с этим программное обеспечение по автоматизации проверки учебной документации будет полезным и востребованным, учитывая, что аналогов практически нет.

После завершения разработки планируется протестировать данную систему и внедрить на кафедре АСУ, а затем и на других кафедрах ТУСУРа. В дальнейшем предполагается выпуск рыночного продукта для различных сфер деятельности.

Литература

1. Стандарты и регламенты [Электронный ресурс]. – Режим доступа: https://www.rst.gov.ru/portal/gost/home/standarts, свободный (дата обращения: 18.09.2022).

2. Перечень стандартов ИСО [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Перечень_стандартов_ИСО#ISO_50001 свободный (дата обращения: 18.09.2022).

3. Требование к качеству учебной документации. Нормоконтроль [Электронный ресурс]. – Режим доступа: http://kei.ulstu.ru/assets/files/pdf/studentam/met_normokontr_2016.pdf, свободный (дата обращения: 18.09.2022).

4. Образовательный стандарт вуза ОС ТУСУР 01–2021. Работы студенческие по направлениям подготовки и специальностям технического профиля. Общие требования и правила оформления от 25.11.2021 [Электронный ресурс]. – Режим доступа: https://regulations.tusur.ru/storage/150499/ОС_ТУСУР_01-2021_(согласовано).pdf?1642386839, свободный (дата обращения: 18.09.2022).

5. Буйвал С.С., Барабанова И.А. Анализ процесса нормоконтроля технической документации и оценка качества выполнения документов // Сб. научных статей 4-й Всерос. конф.: Юго-Западный гос. ун-т. Курск. – 2022. – С. 15–19.

6. Что такое автоматизированная система [Электронный ресурс]. – Режим доступа: https://dic.academic.ru/dic.nsf/ruwiki/16398 свободный (дата обращения: 18.09.2022).

7. Зобнина В.Е., Вихорь Н.А. Автоматизация проверки форматирования текстового документа // Избр. доклады 65-й юбилейной университетской науч.-техн. конф. (УНТК–2019). – Томск: Том. гос. архитектурно-строительный ун-т, 2019. – С. 880–881.

Захарова Александра Александровна
Д-р техн. наук, доцент, проф. каф. АСУ ТУСУРа
ORCID: 0000-0002-2379-8698
Эл. почта: zacharovaa@mail.ru

Аверьянова Анна Михайловна
Ассистент каф. АСУ ТУСУРа
Эл. почта: averjyanova-anna@mail.ru

Глухих Ксения Дмитриевна
Студентка каф. АСУ ТУСУРа
Эл. почта: ksenia2801@mail.ru

УДК 37.03

Ф.А. Кун, Р.С. Герасимов, А.В. Цымбал, Д.Ю. Заигрин

Оценка эффективности достижений студентов

Рассматривается система активизации участия студентов ТУСУРа в общественных, научных, спортивных и иных мероприятиях с помощью информационной системы оценки достижений обучающихся, формирующей лояльность по отношению к активностям вуза.

Ключевые слова: оценка эффективности, система оценки, студенческий актив.

Формирование лояльности студентов по отношению к вузу является приоритетным направлением в деятельности любой образовательной организации. В ТУСУРе реализуется ряд мероприятий, направленных на повышение вовлеченности студентов в деятельность вуза, однако эти мероприятия не дают ожидаемых результатов, реализуются не системно.

В ТУСУРе имеется общая оценка деятельности студентов, в которую входит учебная, научная, личная (различные кружки) и общественная деятельность. При этом учебная и научная деятельность имеет более прозрачную, четкую систему оценивания, достижения в этих сферах записываются в портфолио и помогают выпускникамв создании резюме.

Учет достижений в личной и общественной деятельности, а также формирование четкой, прозрачной системы оценивания достижений по этим направлениям приведет к повышению мотивации студентов участвовать в общественной и культурной жизни вуза. В совокупности с учебной деятельностью это дает возможность всесторонне развивать личность студента, что плодотворно скажется на его жизненной позиции.

Идея проекта

Данный проект предполагает как разработку балльной оценки достижений студентов по общественной, научной, спортивной и иным сферам деятельности, так и возможность добавить свои достижения в портфолио.

Оценка достижений предполагает учет всех видов достижений и начисление баллов/кредитов за участие в каждом мероприятии. Для учета планируется привлечение структурных подразделений ТУСУРа, ответственных за взаимодействие со студентами в рамках отдельных направлений деятельности. Учет достижений возможен через дополнительный модуль в личном кабинете студента «Мои показатели». По результатам учебного года студент может конвертировать накопленные баллы/кредиты и получить материальное вознаграждение. Средства для выплаты материального вознаграждения могут быть предусмотрены в стипендиальном и иных фондах ТУСУРа, предназначенных для стимулирования активности студентов в учебной и внеучебной деятельности.

Отметим, что при составлении резюме для работодателя, помимо учебной и научной деятельности, будут включаться достижения личной и общественной деятельности, что позволит работодателю узнать не только данные профессиональных навыков, но и коммуникативные компетенции.

Это позволит повысить заинтересованность студентов в участии в общественной и культурной жизни вуза.

Основные задачи проекта на этапах реализации

1. Проанализировать нормативную правовую базу ТУСУРа в части мотивации студентов.
2. Проанализировать запросы и потребности заинтересованных сторон в реализации проекта.
3. Разработать модель оценки достижений студентов.
4. Разработать информационную систему модели оценки достижений студентов.
5. Разработать и апробировать макет мобильного приложения для учета достижений студента в цифровое портфолио.
6. Провести опытную эксплуатацию модели, доработку модели по результатам тестирования.
7. Внедрить модель.

Модель оценивания студентов

Модель представляет собой взаимосвязь между критериями и их значимостью в числовом выражении.

Ниже дана таблица с критериями для мероприятий (табл. 1).

В данной таблице первый столбец представляет собой перечень областей, которым принадлежат критерии и их числовая значимость. Второй столбец отвечает за весовой коэффициент областей, в сумме весовые коэффициенты всех областей дают единицу, что является условием работы модели. Третий столбец отвечает за наличие критериев, при выполнении которых студент получает числовую оценку своей деятельности, эквивалентную полученному за мероприятие количеству баллов. А четвертый столбец является числовой значимостью критериев, описанных в третьем столбце.

Аналогичное описание для таблицы успеваемости и посещаемости студента, представленной ниже (табл. 2).

Функционирование модели описывается по следующим формулам:

$$M(j) = 1/N * b * aj * kj, \quad (1)$$

$$Y(i) = 1/L * \Sigma(\alpha i * Pk), \quad (2)$$

$$Б(i, j) = g1 * M(j) + g2 * Y(i), \quad (3)$$

где N – количество мероприятий; L – количество предметов; a, k, b, g, Pk – вес, M – средний балл за участие в мероприятиях; Y – средний вклад в учёбу; $Б$ – общий балл за общественную и учебную жизнь.

Таблица 1

Демонстрация модели оценивания студентов

	Весомость (ai)	Критерии оценивания	Значимость критерии (k)
Уровень мероприятия	0,4	1. Внутри кафедры	0,2
		2. Внутри факультета	0,4
		3. Внутри вуза	0,6
		4. Городское	0,8
		5. Региональное	1
		6. Всероссийские	1,5
		7. Международные	2
Вид деятельности	0,3	1. Умственная	1
		2. Физическая	0,7
		3. Обывательская	0,2
		4. Волонтерская	0,3
Участие студента в организации мероприятия	0,2	1. Организатор	1
		2. Пиарщик	0,5
		3. Регистратор	0,2
		4. Волонтер	0,2
		5. Контролер	0,2
		6. Иная поддержка мероприятия	0,1
		7. Участник	0,1
Продолжительность мероприятия	0,1	1. Меньше часа	0,1
		2. От часа до двух	0,3
		3. От двух часов до трех	0,4
		4. Три и более	0,5

Таблица 2

Оценивание успеваемости и посещаемости студента

Успеваемость	0,6	1. Отличник	1
		2. Отлично / хорошо	0,75
		3. Хорошо	0,5
		4. Удовлетворительно / хорошо	0,3
		5. Удовлетворительно	0,25
Посещаемость	0,4	Выше 90%	1
		70–90%	0,75
		50–70%	0,5
		До 50%	0,25

Среда реализации

IntelliJ IDEA – это ориентированная на язык Java среда разработки от JetBrains. Каждая часть данной платформы была создана для того, чтобы повысить продуктивность разработки. Умный редактор кода в сочетании с приятным для глаза дизайном делают разработку эффективной и приятной [1].

Плюсы

– Доступна для систем Windows, macOS и Linux.

– Умное автодополнение.

– Встроенная отладка.

– Встроенная интеграция с системами контроля версий.

– Интеграция с инструментами сборки, такими как Apache Maven, Gradle и Webpack.

– Поддержка плагинов.

Минусы

– Бесплатная версия не содержит всех инструментов для разработки.

– Высокая стоимость платных версий.

Visual Studio Code. Данная среда разработки включает в себя частичный функционал IDE. Это программа, в которой есть редактор кода и ряд других механизмов, позволяющих его анализировать. Чаще всего этот инструмент представляют, когда речь заходит о IDE для web-разработки, так как эта среда имеет необходимый функционал для решения всех задач. По статистике выбора бесплатного HTML редактора именно Visual Studio code занимает первое место [2].

Плюсы

– Включает в себя большую часть функционала IDE.

– Наличие IntelliSense (механизм автозаполнения).

– Большой спектр расширений и дополнений.

– Кроссплатформенность – существуют версии для разных ОС.

– Интегрирован с Git.

– Открытый исходный код приложения.

– Приложение распространяется бесплатно.

Минусы

– Достаточно большое время запуска приложения.

– Поиск по проектам осуществляется медленно.

Toast editor. Редактор пользовательского интерфейса поддерживающий режим Markdown* и режим WYSIWYG* [3].

Плюсы

– Возможность редактирования Markdown и видеть изменения в реальном времени.

– Синхронная прокрутка Markdown и изображения, не нужно прокручивать отдельно каждый блок.

– Есть подсветка синтаксиса, возможность сразу проверять неработающий Markdown.

– Поддержка различных форматов данных, включая таблицы, блоки с программным кодом и ссылки.

– Возможность подключения дополнительных модулей, таких как графики и схемы.

*Markdown** – облегчённый язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций.

*WYSIWYG** – свойство прикладных программ или веб-интерфейсов, в которых содержание отображается в процессе редактирования и выглядит максимально близко похожим на конечную продукцию.

Минусы

– Нет поддержки ранних версий браузера Internet Explorer, поддерживаются версии 11+.

– Отсутствует русификация интерфейса.

– Сложность конвертации для использования в мобильных приложениях.

Из всех рассмотренных вариантов редакторов наиболее подходящим на наш взгляд является *Visual Studio Code*, потому что большое количество разработчиков используют данное ПО, а также данный продукт обладает массой преимуществ по сравнению с современными аналогами.

Заключение

По результатам деятельности в проекте было разработано математическое обоснование модели оценивания эффективности студентов. Выявлена подходящая среда разработки для создания модуля. Модуль будет спроектирован на языке Ruby с использованием базы данных postgresql.

Литература

1. IntelliJ IDEA [Электронный ресурс]. – Режим доступа: https://www.jetbrains.com/ru-ru/idea/ (дата обращения: 16.09.2022).

2. Visual Studio Code [Электронный ресурс]. – Режим доступа: https://code.visualstudio.com/ (дата обращения: 16.09.2022).

3. Toast Editor [Электронный ресурс]. – Режим доступа: https://ui.toast.com/tui-editor (дата обращения: 16.09.2022).

Кун Филипп Александрович
Студент каф. АСУ ТУСУРа
Эл. почта: filipp.kun@gmail.com

Герасимов Роман Сергеевич
Студент каф. АСУ ТУСУРа
Эл. почта: gerrom1212@mail.ru

Цымбал Антон Вадимович
Студент каф. АСУ ТУСУРа
Эл. почта: toxarus286@gmail.com

Заигрин Денис Юрьевич
Студент каф. АСУ ТУСУРа
Эл. почта: zadin002@mail.ru

УДК 338.001.36

В.А. Семиглазов, Н.А. Романенко, В.А. Хамзина, Т.Н. Мосунова

Подбор инструментов для онлайн-продвижения курсов и оценки затрат по их применению для реализации проекта

Рассматриваются базовые современные инструменты для публикации рекламных материалов на интернет-площадках, а также показатели, влияющие на потенциальную эффективность продвижения объявлений на онлайн-сайтах. В процессе производится оценка затрат на универсальные методы продвижения в рассматриваемых ресурсах.

Ключевые слова: маркетинг, методы продвижения, онлайн-курсы.

С ростом конкуренции на рынке дополнительного профессионального образования важно учитывать выигрышные инструменты и методы продвижения собственного продукта. В связи с цифровизацией множества процессов создателям онлайн-курсов необходимо делать упор на преимущества собственного товара и продвигать его на подходящих каналах сбыта, способствующих эффективно вводить продукт на рынок, стимулировать активный спрос и продажи, также формировать целевую аудиторию лояльных клиентов. Реализация данных целей формируется в процессе планирования верных маркетинговых коммуникаций.

Интернет на сегодня является наиболее действенной площадкой и средством продвижения онлайн-курсов. Так, за счет большого количества потенциальных клиентов во Всемирной сети существует множество различных интеграций рекламы.

Однако несмотря на прогрессивные методы продвижения продуктов, в мире все еще существуют архаичные виды рекламы, такие как телевизионная, реклама в газетах и журналах, реклама на радио и т.д. Несмотря на устарелость данных видов, каждый имеет место быть. Поскольку это обусловлено сформированной целевой аудиторией, реклама на данных площадках предусмотрена для клиентов старше пятидесяти лет. На данный момент практически для любого бизнеса интернет-реклама служит одним из самых эффективных способов продвижения товаров и услуг с точки зрения соотношения затрат и результатов.

Как правило, проблема привлечения клиентов в онлайн-обучении заключается не только в выборе платформы для размещения рекламного объявления, но и в самих клиентах, оценивающих ценовую политику, известность лекторов, конкретный результат обучения и т.д. Подобные факторы формируют специфику продвижения дополнительного профессионального образования в сфере онлайн, построенной из ряда задач, при выполнении которых повышается уровень потока клиентов.

Для продвижения онлайн-курса «Конкурентоспособность в цифрах» задачи можно разделить на:

1. *Подбор площадки по наличию в ней целевой аудитории.* Целевая аудитория – люди, объединенные общими признаками или ради общей цели или задачи. Если сегментировать потенциальных клиентов по таким критериям, как пол, возраст, профессия, то выясняется, что портрет целевой аудитории программы ДПО выглядит следующим образом: мужчины и женщины в возрасте от 20 лет, начинающий предприниматель, сотрудник компании, в которой не хватает специалиста по оценке качества товаров.

В статье Т.Н. Мосуновой, В.А. Хамзиной, Е.С. Костюк, Н.А. Романенко «Подбор маркетинговых средств продвижения онлайн-курсов по программам ДПО» было проведено сравнение и выбраны наиболее подходящие инструменты продвижения в области дополнительного образования [1]. Данный выбор был основан на анализе возрастного состава ЦА, возможности оплаты онлайн-курсов, количества пользователей (более 100 тыс.), выгоды / задачи пользователей.

2. *Осуществление SMO-продвижения.* SMO (Social media optimization) – это комплекс мер, направленных на повышение количества упоминаний продукта и ссылок на сайт в социальных сетях. Смысл метода состоит в оптимизации собственного сайта под социальные медиа, т.е. под посетителей. SMO направлен на улучшение дизайна сайта в направлении современных тенденций, на создание интуитивно понятного интерфейса, на совершенствование навигации. Все приведенные аспекты легли в основу метода, за счет которого анализируется и совершенствуется страница с контентом, представленным потенциальным клиентам. В ходе применения сайта или визитки, основанной на SMO-продвижении, повышается привлекательность ресурса. В идеале это значит, что пользователь должен задержаться на сайте, превратиться в клиента и поделиться контентом с другими людьми [2].

3. *Осуществление SMM-продвижения.* SMM (Social media marketing) – плана достижения бизнес-целей компании при помощи социальных сетей. Стратегия поможет определить цели присутствия в общественных медиа, пути их достижения, расставить приоритеты, а главное – спрогнозировать потенциальный эффект.

На этапе создания рекламы важно включить актуальные ресурсы, использующиеся при создании контента и его продвижении в социальных сетях.

Для создания рекламы важно также учитывать следующие аспекты:

• визуальный контент должен обладать уникальностью;

• нужно использовать популярные фото- и видеоматериалы;

• необходимо делать акцент на продукте;

• важно включать и учитывать мнение экспертов.

Оценивая текущее состояние на рынке предложений по численности пользователей, актуальности социальных сетей, активности, можно выделить некоторые площадки для продвижения продукта.

«ВКонтакте» – крупнейшая рекламная площадка Рунета. В конце 2021 г. ежемесячная аудитория соцсети составляла 72,5 млн пользователей, а ежедневная – 47 млн, в марте 2022 г. дневная аудитория соцсети выросла минимум на 4 млн человек. Любому бизнесу уместно и полезно размещать рекламы на просторах социальной сети «ВКонтакте» [3].

Осуществление рекламы возможно в нескольких интеграциях:

• *Таргетированная реклама* – это объявления, которые показываются пользователям социальной сети в правой части страницы и в новостной ленте. Объявления могут состоять из заголовка, изображения, короткого текстового описания и различных медиаматериалов. При настройке рекламы добавляют портрет потенциального покупателя, поэтому подобные объявления видят пользователи, которые соответствуют описанию целевой аудитории рекламодателя.

• *Размещение клипов.* В сутки короткие видео на платформе смотрят 350 млн раз. На текущий момент возрастает количество клипов с обучающим наполнением.

В ходе использования рекламы возможна низкая эффективность, которая по первой и важной причине обусловлена неправильной настройкой таргетинга, т.е. ошибочным выбором целевой аудитории. На основе некорректного выбора условий кликабельность и активность заинтересованных пользователей будет стремиться к нулю. Вторая причина может заключаться в самом наполнении рекламного объявления: не отображается суть, неинтересный заголовок, плохой дизайн и т.д. [4].

Важной составляющей эффективности контент-маркетинга выступает воронка продаж AARRR.

Acquisition – этап привлечения. На этом этапе используем всем привычные форматы: активно предлагаются бонусы для новых подписчиков, публикуется полезный контент о продукте/услугах, акции и конкурсы. Об эффективности контента на этом этапе можно судить по достижению поставленных целей: например, сколько подписчиков удалось привлечь в сообщество и рассылку. Для привлечения используются акции и скидки, конкурсы, бонусы за подписку и полезный контент о продукте, на данном этапе об эффективности можно судить лишь по достижению поставленных целей: например, по количеству привлеченных подписчиков.

Для понимания результативности действий на каждом этапе стратегии необходимо анализировать полученные результаты.

Например, эффективность рекламы для привлечения новой аудитории можно анализировать по стоимости подписчика. В идеале чем она ниже – тем лучше.

Activation и Retention – этап активации и удержания. В процессе адаптации к специфике соцсетей происходит объединение двух этапов в один, чтобы удерживать аудиторию в соцсетях при условии, что контент полезен и интересен.

На этом этапе используем следующие виды контента: демонстрация продукта, его описание, которые мотивируют потребителя взаимодействовать.

Revenue – этап «желание заплатить». На этом этапе нужно использовать контент, рассказывающий о конкретных товарах и услугах, описывающий условия покупки, доставки, помогающий сделать выбор в пользу данного товара. В стратегии задействуются скидки, специальные предложения

Referral – этап «желание рекомендовать». Доверие пользователей соцсетей в большей степени склонно к людям, что знают, чем к брендам, поэтому нужно обязательно работать над тем, чтобы пользователи рекомендовали вас своим подписчикам [5].

Контекстная реклама – вид рекламы, которая появляется исходя из контекста запроса пользователя. Отличие контекстной рекламы от таргетированной состоит в том, что в первом случае объявления показываются на основе введенного поискового запроса, во втором случае объявления видят только те, которые соответствуют заданной целевой аудитории.

Считается, что два лидирующих сервиса контекстной рекламы разделили сферы влияния: Google AdWords – зарубежный рынок, Яндекс.Директ – только российский. Это сервисы с помощью которого можно создавать и настраивать контекстную и медийную рекламу разных форматов: от текстовых сообщений до видеороликов.

Первые четыре объявления в блоке – рекламные. Каждый участник аукциона, выбравший для показа определенные ключевые слова, сделал свою ставку. Алгоритм рассчитал ставки всех участников для поискового запроса, рассчитал качество их объявлений, кликабельность и расположил их в порядке убывания получившегося рейтинга.

Эффективность контекстной рекламы на примере «Яндекс Директ» можно рассчитывать по трем выполненным категориям:

• качественно составленный сайт. Разница в конверсии между проработанным и «сделанным на коленках» сайтом может превышать в сотни раз. При сравнении выбор клиента в большинстве случаев будет в сторону качественного, приятного сайта;

• качественное коммерческое предложение, в свою очередь которое не заставит клиента сомневаться. Для этого необходимо построить канал взаимодействия с потребителем;

• навыки в верной настройке контекстной рекламы, т.е. привлечение специалиста [6].

Mail-маркетинг. По данным за 2021 г. покупку программ ДПО производили: 38,05% – слушатели

при поддержке работодателей, 34,9% – государство, 19,27% – участники программ, 7,78% – работодатели. Из представленной статистики можно отразить принцип подходящего для компаний маркетингового хода – mail-маркетинг. Данный тип маркетинга основан на рассылке электронных писем, благодаря которым можно отправлять письма по онлайн-обучению в крупные инновационные компании, заинтересованные в повышении квалификации сотрудников в сфере конкурентоспособности инновационных товаров.

Модель эффективности метода использует такие характеристики параметров, как:

- *открываемость*. Для увеличения количества открываемости писем необходимо создавать «кричащую» тему, удалять неактивные адреса, создавать релевантные ссылки по сегментации получателей, оптимизировать время отправки под клиента;
- *кликабельность*. Необходимо оптимизировать письма под мобильные версии, добавлять активные глаголы в письмо и т.д.;
- *коэффициент конверсии*. Приоритетная метрика, эффективность которой увеличивается за счет персонализации, выбора контента клиентом, грамотного акцентирования на текст [8].

Крауд-маркетинг – это ход коммуникации с целевой аудиторией на просторах интернета для повышения интереса к определенному продукту, бренду.

Крауд-маркетинг рассчитан на действенное использование внешних возможностей, например, комментарии с активной ссылкой, отзывы на сайтах нацелены на увеличение трафика потенциальных клиентов, его узнаваемости.

Эффективность метода включает не только нахождение тематической площадки, но и актуальность темы на момент создания рекламы, создание естественного, цепляющего текста. Важно помнить, что массовое появление одного текста во всех площадках приведет покупателя к мысли о навязчивой рекламе.

Выбор оптимального варианта размещения рекламы онлайн-курсов предусматривает подход к оценке затрат, наличие целевой аудитории, также платежеспособность потенциальных покупателей. Так, например, сравнение показателей представлено в таблице.

Таким образом, изучив площадки для размещения интернет-рекламы, возможно без труда выбрать наиболее оптимальный метод продвижения на основе сравниваемых показателей, в итоге в соответствии с выбранным типом рекламы возможно сформировать качественный маркетинговый план. На данный момент mail-маркетинг наименее выигрышный, поскольку рассылка – это большой труд, требующий определенного количества времени для освоения. Помимо этого, проблемой данного метода является спам. В групповом проекте «Разработка программ ДПО по профилю кафедры» создается курс, на примере которого возможно провести тест продвижения на выгодной по всем перечисленным критериям площадке.

Оценка затрат на применение инструментов, заинтересованность аудитории, платежеспособность

Тип рекламы	Оценка затрат	Заинтересованность аудитории	Платежеспособность
ВКонтакте	Можно установить лимит на день: от 1 руб. за клик, от 50 руб за 1000 просмотров	x	x
Яндекс Директ	Цена одного клика устанавливается рекламодателем самостоятельно, с учетом того, что один клик может стоить от 0,01 до 50 у.е.	x	x
Google AdWords		x	x
Mail-маркетинг	Для эффективности работы рассылки необходимо найти специалиста в этой области, з/п от 80 000 руб. / мес		x
Крауд-маркетинг	Временные затраты / затраты на копирайтера, от 50 руб./1000 символов	x	

В результате проделанной работы были подобраны инструменты для онлайн-продвижения курсов, а также была проведена оценка затрат по их применению для реализации проекта. В таблице представлены наиболее подходящие площадки – «Вконтакте», «Яндекс Директ» «Google AdWords» – как по ценовой политике, так и по методам привлечения клиентов. По представленным данным «Вконтакте» – актуальная социальная сеть, которой пользуется способная, обучаемая молодая аудитория, которая в результате может стать основным из потребителей из-за удобного и актуального для обучения онлайн-формата. Контекстная реклама, в свою очередь, «Яндекс Директ» и «Google AdWords» удобны для привлечения уже заинтересованной аудитории, готовой платить. При правильной подаче продукта и при эффективных рекламных действиях на перечисленных площадках возможно сформировать лояльность потребителей, тем самым обеспечив высокую узнаваемость и коммерциализацию курсов.

Литература

1. Мосунова Т.Н. Подбор маркетинговых средств продвижения онлайн-курсов по программам ДПО / Т.Н. Мосунова, В.А. Хамзина, Е.С. Костюк, Н.А. Романенко // Научная сессия ТУСУР–2022: матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых, Томск, 18–20 мая 2022 г.: в 3 ч. – Томск: В-Спектр, 2021. – Ч. 2. – С. 151–155.

2. SMO: как привлекать посетителей на сайт через соцсети [Электронный ресурс]. – Режим доступа: https://checkroi.ru/blog/smo-chto-eto-takoe/ http://www.ssl.stu.neva.ru/psw/crypto.html, свободный (дата обращения: 02.10.2022).

3. ВКонтакте подвела итоги первого квартала 2022 г.: количество российских пользователей в месяц – 73,4 млн,

ежедневных просмотров, VK Видео – 2,45 млрд [Электронный ресурс]. – Режим доступа: https://vk.com/press/q1-2022-results

4. Морозова А.А. Вербальный и иконический компоненты таргетированной рекламы в социальной сети «ВКонтакте» // Знак: проблемное поле медиаобразования. – 2020. № 1. – С. 58–62.

5. Продвижение по методу AARRR [Электронный ресурс]. – Режим доступа: https://tilda.education/articles-aarrr-metrics

6. Гилев Н.А. О контекстной рекламе // Молодой ученый. – 2010. – Т. 1, № 5 (16). – С. 167–169.

7. Правительство представило статистику дополнительного профобучения россиян [Электронный ресурс]. – Режим доступа: https://skillbox.ru/media/education/pravitelstvo-predstavilo-statistiku-dopolnitelnogo-profobucheniya-rossiyan/

Семиглазов Вадим Анатольевич
Канд. техн. наук, доцент каф. управления инновациями (УИ) ТУСУРа
Эл. почта: vadim.tusur@mail.ru

Романенко Наталья Александровна
Студентка каф. УИ ТУСУРа
Эл. почта: natalya.romanenko2002@gmail.com

Хамзина Венера Александровна
Студентка каф. УИ ТУСУРа
Эл. почта: venera_18.10@mail.ru

Мосунова Татьяна Николаевна
Студентка каф. УИ ТУСУРа
Эл. почта: mosunova2013@mail.ru

Секция 18

ВИДЕОИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ЦИФРОВОЕ ТЕЛЕВИДЕНИЕ

Председатели секции – ***Курячий Михаил Иванович****, доцент каф. ТУ, к.т.н.,*
Капустин Вячеслав Валериевич*, доцент каф. ТУ, к.т.н.*

УДК 004.932.4

С.А. Забуга, В.В. Капустин, А.Н. Шлыкович

Оценка эффективности методов повышения кадровой частоты видеопотока

Проведено сравнение эффективности методов временной интерполяции кадров с помощью вычисления оптического потока, а также с применением свёрточной нейронной сети.

Ключевые слова: временная интерполяция кадров, сверточная нейронная сеть, перцептивные потери, оптический поток.

Методы, решающие задачу повышения кадровой частоты видеопотока, уже долгое время встречаются в научных публикациях. В связи с широким распространением данной проблемы, начало набирать популярность использование искусственных нейронных сетей в работах, направленных на изучение компьютерного зрения. Традиционные решения для этой задачи были предложены в виде двухэтапных методов, состоящих из расчета оптического потока, такими как метод Лукаса–Канаде с мультискейллингом и без, метод Фарнербэка, а также метод SimpleFlow, и последующей интерполяции. Данные методы дают возможность оценить перемещение пикселей между кадрами и, исходя из этого, предсказывать их координаты [1–4].

Архитектура сверточной нейронной сети временной интерполяции кадров

Нейронная сеть (НС) – это математическая модель из искусственных нейронов, объединённых между собой, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма. Общность нейронов, соединенных друг с другом параллельно или последовательно, в зависимости от задачи сети, создает собой сеть, которую называют нейронной [5].

Нейрон в нейронной сети является единичным элементом, в способности которого входит вычисление выходного сигнала из объединённых входных векторов. Последовательность действий единичного элемента нейронной сети состоит из: принятия векторов от предыдущих нейронов, объединения входных векторов, вычисления выходного вектора, передачи выходного вектора следующим нейронным связям. Различают многослойные и однослойные нейронные сети. В многослойных нейронных сетях происходит объединение единичных объектов нейронных сетей, таких как нейрон, в один слой. Подобный слой объединяет нейроны с неделимыми входными векторами (рис. 1) [6].

Входными данными нейронной сети являются два изображения, обозначим их I_0 и I_1, между которыми вставляется промежуточный(-ые) кадр(-ы) $\{I_{t\,i}\}^{N}_{i=1}$, где $t_i \in (0, 1)$ [7]. Обозначим реальные промежуточные кадры как $\{\hat{I}_{t\,i}\}^{N}_{i=1}$.

Функцией потерь в данной нейронной сети является линейная комбинация четырех условий:

$$l = \lambda_r l_r + \lambda_p l_p + \lambda_w l_w + \lambda_s l_s. \quad (1)$$

Значения коэффициентов были установлены эмпирическим путем и равны λ_r = 0,8, λ_p = 0,005, λ_w = 0,4, λ_s = 1.

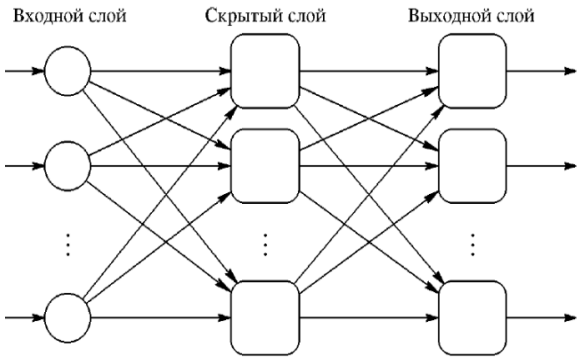


Рис. 1. Схематическое представление многослойной нейронной сети

Потери воссоздания l_r. Оценивают, насколько хорошо было произведено воссоздание промежуточных кадров:

$$l_r = \frac{1}{N}\sum_{i=1}^{N}\left\|\hat{I}_{t_i} - I_{t_i}\right\|_1. \quad (2)$$

Перцептивные потери l_p. Необходимы для сохранения деталей прогнозов и увеличения четкости интерполированных кадров [2]

$$l_p = \frac{1}{N}\sum_{i=1}^{N}\left\|\varphi(\hat{I}_t) - \varphi(I_t)\right\|_2. \quad (3)$$

Потери при варпинге l_w. Оценивают качество вычисленного оптического потока

$$l_w = \left\|I_0 - g(I_1, F_{0\to1})\right\|_1 + \left\|I_1 - g(I_0, F_{1\to0})\right\|_1 + \\ + \frac{1}{N}\sum_{i=1}^{N}\left\|I_{t_i} - g(I_0, \hat{F}_{t_i\to0})\right\|_1 + \frac{1}{N}\sum_{i=1}^{N}\left\|I_{t_i} - g(I_1, \hat{F}_{t_i\to1})\right\|_1. \quad (4)$$

Для вычисления светового потока и CNN используется архитектура U-Net, представленная на рис. 2. U-Net – это свёрточная нейронная сеть, состоящая из кодера и декодера со сквозными соединениями между слоями.

Сравнение результатов работы алгоритмов временной интерполяции кадров

Для оценки качества интерполированных кадров были выбраны следующие объективные метрики: PSNR и SSIM. Для исследования использовалось

исходное видео с разрешением 1920 на 1080 пикселей, с частой 60 кадров в секунду. Кадры исходного видеопотока были прорежены в 2 раза до частоты 30 кадров в секунду. После применения нейронной сети и методов интерполяции с использованием оптического потока кадровая частота видеопотока была восстановлена, а получившиеся промежуточные кадры сравнивались с исходными.

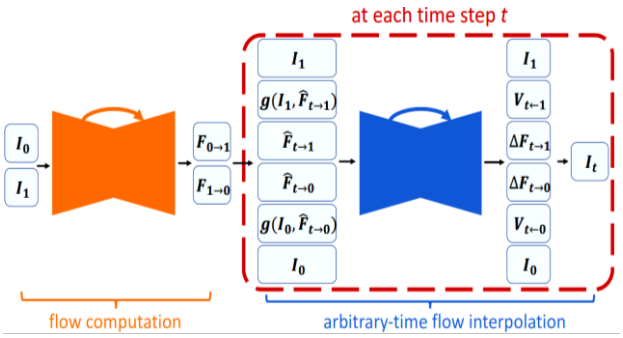
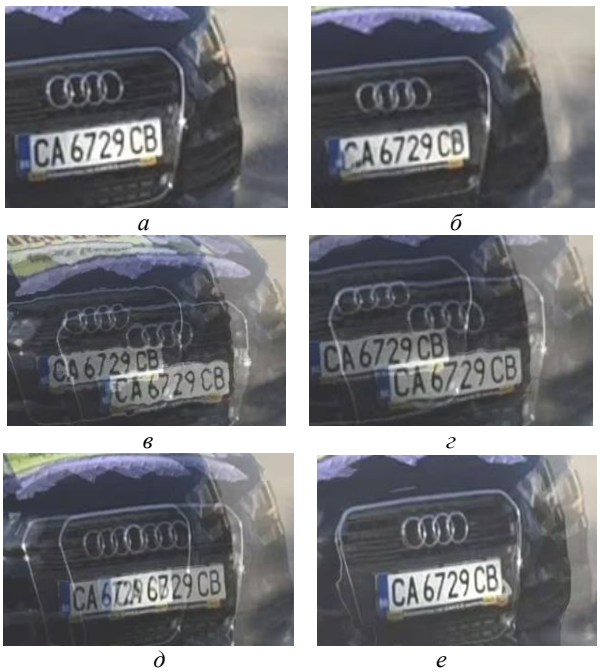


Рис. 2. Архитектура нейронной сети

Обучение нейросети и вычисление оптического потока проводились на одном компьютере. Основные параметры: процессор AMD Ryzen 3700x с тактовой частотой 4450 МГц, 8 ядер, 16 потоков кэш 1-го уровня 512 КБ, кэш 2-го уровня 4 МБ, кэш 3-го уровня 32 МБ, SDD-диск с интерфейсом SATA; видеокарта NVIDIA RTX 2060 6ГБ; оперативная память (RAM): 16 ГБ, частота оперативной памяти 3333 МГц. На рис. 3 приведены исходный и интерполированные кадры для сравнения.



а *б* *в* *г* *д* *е*

Рис. 3. Реальный кадр (*а*), интерполированный нейронной сетью (б), метод Лукаса–Канада без мультискейллинга (*в*), метод Лукаса–Канада с мультискейллингом (*г*), метод Фарнербэка (*д*), метод SimpleFlow (*е*)

Для интерполированных кадров можно заметить такой артефакт, как размытие и двоение объекта в области быстрого движения точек. Визуально сравнивая методы интерполяции, можно заметить, что нейросеть лучше справляется с быстродвижущимися объектами, при этом не деформируя их.

Подобные результаты достигаются благодаря хорошо подобранному датасету видеофайлов с большим разрешением и количеством кадров. Поскольку вносимые искажения не так значительны, можно говорить о развитии технологии сверточных нейронных сетей в задачах интерполяции кадров видеопотока.

В методе Лукаса–Канаде достигаются самые большие искажения в промежуточном кадре, такие как гостинг, алиасинг и размытие. Также наблюдается понижение качества изображения. Все это следствия применения стандартной реализации метода Лукаса–Канаде без мультискейллинга, т.е. без применения пирамиды изображений. Метод очень чувствителен к быстродвижущимся объектам, так как при использовании ряда Тейлора мы находим изменение значения функции в конечной окрестности точки, а производная дает представление о бесконечно малой окрестности.

При рассмотрении промежуточных интерполированных кадров на рис. 3 можно заметить превосходства метода расчета оптического потока Фарнербэка, над методом Лукасом–Канаде. Оптический поток в данном методе больше приближен к реальности и не имеет потерю качества на границах объекта. Подобные результаты достигаются за счет компенсации шумов при вычислении, используется предположение, что в окрестности исследуемой точки у всех точек одинаковый сдвиг, а также не меняется интенсивность.

В таблице приведены результаты по времени синтеза промежуточных кадров рассматриваемыми методами.

Сравнение времени синтеза промежуточных кадров

Методы интерполяции	Среднее время обработки кадра, с	Способ вычисления
Лукас–Канаде без мультискейллинга	0,56	CPU
Лукас–Канаде с мультискейллингом	1,34	CPU
Фарнербэк	2,13	CPU
SimpleFlow	1,53	CPU
CNN	2,5	GPU

На рис. 4 изображена зависимость PSNR каждого метода интерполяции от номера промежуточного кадра. Исходя из графика, можно заметить, что сверточная нейронная сеть обладает в среднем более высокими показателями PSNR относительно других методов.

На рис. 5 изображена зависимость SSIM каждого метода интерполяции от номера промежуточного кадра. Так же, как в рис. 4, можно заметить, что сверточная нейронная сеть обладает в среднем более большими показателями SSIM относительно других методов.

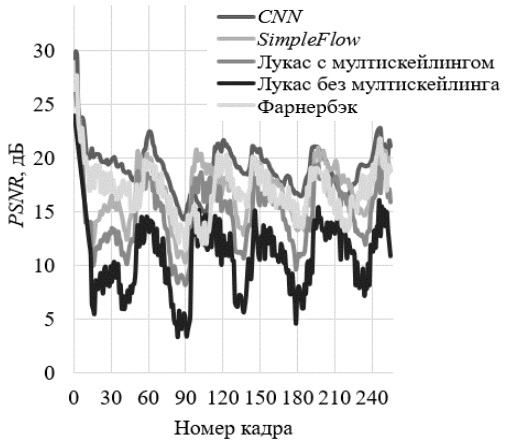


Рис. 4. График изменения PSNR от номера кадра

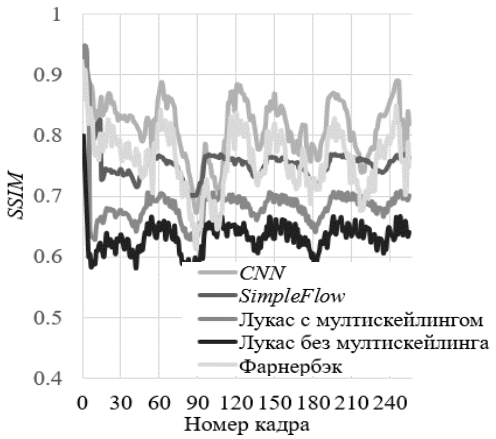


Рис. 5. График изменения SSIM от номера кадра

Периодическое изменение амплитуды можно объяснить, периодичностью проезда автомобилей перед видеокамерой, так как методы оптического потока плохо справляются с большими изменениями в координатах точек.

По результатам видно, что метод SimpleFlow имеет оптимальные характеристики относительно качества, точности и скорости синтеза кадра среди стандартных методов расчета оптического потока.

Показатели метода вычисления оптического потока SimpleFlow обладают большей стабильностью по сравнению с другими стандартными методами, а также в сравнении с искусственной сверточной нейронной сетью, а в некоторых кадрах, например, с 100-го по 120-й, даже превосходят нейронную сеть. Из этого следует, что стандартные методы, в некоторых случаях могут превосходить методы, использующие нейронные сети.

Заключение

Анализ современных подходов к решению задачи временной интерполяции видеопотока показал, что несмотря на факт существования относительно быстрых и качественных алгоритмов, таких как световой поток, постоянно появляются новые методы. В последние годы все чаще для решения этой задачи используются искусственные нейронные сети. В большинстве исследований используются различные реализации моделей сверточных нейронных сетей.

Анализируя результаты, был сделан вывод, что использование сверточной нейронной сети позволяет добиться более лучшего качества промежуточных кадров и меньших искажений при прочих равных условиях. Также даже при использовании графического ускорителя с относительно большой мощностью расчет интерполированного кадра с применением сверточной нейронной сети занимает почти в два раза больше времени в сравнении со стандартными методами. Это большой недостаток нейронных сетей, так как, чтобы приблизиться ко времени обычных методов, нужно иметь большие вычислительные мощности.

Рассматривая стандартные методы, можно увидеть, что более новый метод SimpleFlow, исходя из данных, приведённых в данной работе, обладает лучшей способностью к предсказанию перемещения точек. Кроме того, он не имеет недостатка в виде невозможности детектирования или ошибочного детектирования быстродвижущихся объектов.

Литература

1. Bouguet J.Y. Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm // Intel Corporation Microprocessor Research Labs. – 2007. – PP. 1–9.

2. Fleet D.J., Weiss Y. Optical Flow Estimation // ICCV. – 2005. – PP. 9–20.

3. SimpleFlow: A Non-iterative, Sublinear Optical Flow Algorithm / M. Tao, J. Bai, P.t Kohli, S. Paris // Computer Graphics forum.– 2015. – PP. 345–353.

4. Гафаров Ф.М., Галимянов А.Ф. Искусственные нейронные сети и приложения: учеб. пособие. – Казань: Изд-во Казан. ун-та, 2018. – С. 5–23.

5. Video frame interpolation via adaptive separable convolution / S. Niklaus, L. Mai, F. Liu // ICCV. – 2017. – PP. 4–5.

6. Video frame synthesis using deep voxel flow / Z. Liu, R. Yeh, X. Tang, Y. Liu, A. Agarwala // ICCV. – 2017. – PP. 1–10.

7. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition // CoRR. – 2014. – PP. 15–17.

Забуга Сергей Александрович
Магистрант каф. ТУ ТУСУРа
Эл. почта: sergeizabuga@gmail.com

Капустин Вячеслав Валериевич
Канд. техн. наук, доцент каф. ТУ ТУСУР
Эл. почта: vk@tu.tusur.ru

Шлыкович Андрей Николаевич
Магистрант каф. ТУ ТУСУРа
Эл. почта: iksis23@bk.ru

УДК 004.855.5

В.С. Чебанов

Трансферное обучение модели SSD-MobileNet с использованием микрокомпьютера Jetson Nano

Рассматривается трансферное обучение ранее обученной модели SSD-MobileNet на определенную выборку классов, используя микрокомпьютер Nvidia Jetson Nano. Приведены результаты обнаружения полученной модели при различном количестве эпох обучения.

Ключевые слова: трансферное обучение, SSD-MobileNet, Nvidia Jetson Nano, ImageNet, PyTorch.

В современном мире использование нейросетей все больше проникает в нашу повседневность. С помощью алгоритмов глубокого машинного обучения телефон может быть разблокирован с помощью сканирования лица его владельца, он учитывает различные факторы, чтобы систему не могла обмануть просто фотография человека или его закрытые глаза. Используются нейросети для обработки изображений можно повышать разрешение, резкость, а также выполнять постобработку снятых кадров самими смартфонами.

Еще одним примером служит проект Facebook под названием DeepFace – алгоритм распознавания лиц на фотографиях, точность которого достигает 97%, независимо от качества фото, степени освещенности и расположения человека на снимке [1].

Рассматривая применения нейросетей, стоит также отметить отечественный пример. Актуальным и бытовым примером может послужить подборка плейлистов, которые могут быть интересны слушателю, а основывается подборка на уже добавленных аудиозаписях пользователя. Не менее актуальна работа алгоритмов и нейросетей на видеохостингах, где, анализируя информацию, получаемую от пользователя, формируются подборки видеороликов, которые могут быть интересны пользователю.

Если же рассматривать сферу алгоритмов автоматического распознавания и обнаружения транспортных средств, то стоит отметить, что предложений присутствует огромное количество, а также предъявляются довольно низкие требования для использования этих алгоритмов. Сейчас можно использовать готовые алгоритмы в интернет-ресурсах. Есть слабые стороны обнаружения транспортных средств, что сравнимо с распознаванием людей, где наибольшей проблемой является распознавание детей, которых алгоритмы обнаруживают и определяют с наибольшим количеством ошибок.

Для алгоритмов распознавания транспортных средств слабым местом являются скутеры, байки, мотоциклы. Если рассматривать Индию, то можно отметить огромный поток именно двухколесных транспортных средств, поэтому эта узконаправленная сфера является актуальной и требует решений.

Трансферное обучение

Изначально стоит разобраться, что собой представляет трансферное обучение, в чем заключается его смысл и какими оно обладает преимуществами.

Трансферное обучение (Transfer Learning) – подраздел машинного обучения, целью которого является применение знаний, полученных при решении одной задачи, для решения другой задачи.

Примером использования можно представить использование знаний, полученных при обучении на классификации пользовательских статей Википедии, чтобы в дальнейшем, используя накопленные знания и методы, применить это для классификации медицинских текстов. Или же использовать обученную модель на большое количество классов и применить трансферное обучение для узконаправленной области. Использование трансферного обучения позволяет добиться более высокой точности правильного определения объекта и повышения быстродействия.

Нейронные сети обладают рядом уникальных свойств, которые делают их мощным инструментом для создания систем управления: способностью к обучению на примерах и обобщению данных, способностью адаптироваться к изменению свойств объекта управления и внешней среды, пригодностью для синтеза нелинейных регуляторов, высокой устойчивостью к повреждениям своих элементов в силу изначально заложенного в нейросетевую архитектуру параллелизма [2].

Стоит отметить огромную конкуренцию в области искусственного интеллекта, она присутствует, начиная от разработки алгоритмов, до качества исходных данных и их выборке, что обусловлено тем, что успешные модели и алгоритмы основаны на контролируемом обучении, которое требует поддержки больших массивов данных.

Развитие искусственного интеллекта имеет тенденцию соответствовать точным требованиям приложений, не требуя больших объемов данных. Таким образом, обучение а малых базах данных становится новой тенденцией.

Методы обучения по небольшим данным, представленные миграционным обучением и обучением с подкреплением, могут лучше отражать искусственный интеллект для успешного трансферного обучения [3].

Трансферное обучение модели SSD-MobileNet

Трансферное обучение было выполнено на сетевой архитектуре SSD-Mobilenet, используя PyTorch и набор данных Open Images.

Классы, использованные для обучения, были выбраны из 600 классов изображений, представлен-

ных в наборе данных Open Image. Выборка состояла из 5 классов, а именно: Bicycle, Bicycle wheel, Motorcycle, Vehicle, Vehicle registration plate.

Модель была обучена на 5 классов, выборка изображений составила 6 726 изображений, из них 3 862 были распределены на мотоциклы как наиболее приоритетный класс. Обучение было произведено на микрокомпьютере Jetson Nano с использованием библиотеки Hello AI World, набора данных Open Image, обучающих сценариев PyTorch, а также Docker Container. Для большей точности необходимо большее количество изображений для обучения, однако микрокомпьютер Jetson Nano не подходит для этого, так как при обучении больше, чем на семи-восьми тысячах изображений, происходит отключение системы.

Использование контейнера обусловлено тем, что контейнеры – такой способ упаковки, где приложение и его зависимости помещаются в единый образ. Для запуска образа используется изолированная среда, которая, в свою очередь, не имеет никакого влияния на основную операционную систему. При использовании контейнеров полезным становится то, что отделение приложения от инфраструктуры позволяет разработчику не заострять внимание на окружении приложения, настройках и зависимостях. Одним из главных достоинств единого образа является то, что он может быть запущен на других системах, сохраняя все настройки и зависимости.

Docker – это свободно распространяемая платформа для развертывания и эксплуатации приложений. Целью Docker является более быстрый и простой запуск приложений, а также легкий перенос с одного хоста в другой. Он позволяет быстрее тестировать и быстрее разрабатывать приложение. Docker – это контейнерная виртуализация, т.е. легковесная платформа виртуализации, которая умеет запускать различные программы, написанные на разных языках программирования и на разных технологиях [4].

Изначально стоит разграничить различия в понятиях контейнеризации и виртуализации. Если рассматривать виртуализацию, то она представляет собой будто бы отдельный компьютер, который имеет свою виртуальную операционную систему и оборудование. В свою очередь, при контейнеризации запуск виртуальной среды происходит прямо из ядра операционной системы. Таким образом, можно сделать заключение, что контейнер находится лишь в той операционной системе, где основная, а не создает виртуальную, как это происходит в виртуализации. Использование контейнеризации также позволяет сэкономить ресурсы в силу того, что не виртуализирует еще одну систему и оборудование.

Базовая модель уже была предварительно обучена на другом наборе данных (PASCAL VOC), поэтому не нужно обучать SSD-Mobilenet с нуля, что заняло бы гораздо больше времени. Вместо этого было использовано трансферное обучение для обнаружения новых классов объектов.

Обучение происходило в течение 26 эпох, на рис. 1 представлен график зависимости количества ошибок от номера эпохи.

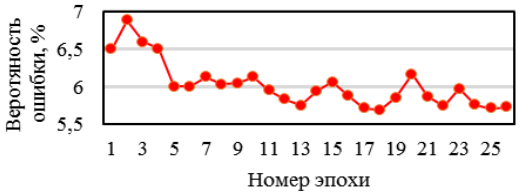


Рис. 1. Зависимость вероятности ошибки распознавания от номера эпохи

Исследования распознавания мотоциклов, как приоритетного класса проводились на 10, 18 и 26 эпохах, как имеющей наименьшую вероятность ошибки.

Рассмотрим распознавание первого изображения на 10, 26 и 18-й эпохе. Стоит отметить, что все изображения обрабатываются в среднем со скоростью 30 кадров с секунду.

В результате модель допустила ошибку и на одном изображении с мотоциклом выделила сразу два класса. Хоть и большую вероятность имеет класс мотоцикла, но результат все равно является ошибочным (рис. 2).

Рис. 2. Результат работы модели на 10-й эпохе

Определение класса на изображении проведено успешно, вероятность составляет 58,3% (рис. 3).

Рис. 3. Результат работы модели на 18-й эпохе

Вероятность обнаружения составляет 64,6%, мотоцикл распознан верно (рис. 4).

Рис. 4. Результат работы модели на 26-й эпохе

Из-за малого количества эпох обучения происходит ошибка в обнаружении класса и мотоциклу присваивается класс велосипеда (рис. 5).

Рис. 5. Результат работы модели на 10-й эпохе

Вероятность обнаружения составляет 53,0% и присвоен правильный класс, обнаружение произошло успешно (рис. 6).

Рис. 6. Результат работы модели на 18-й эпохе

При максимальной эпохе обучения достигается наибольшая вероятность обнаружения мотоцикла, и она составляет 80,9%, а также размеры мотоцикла, выделенные квадратом, соответствуют действительности (рис. 7).

Рис. 7. Результат работы модели на 26-й эпохе

Для большей объективности оценки работы модели исследуем обнаружение мотоцикла при расположении мотоцикла анфас.

При малом количестве эпох обучения происходит неправильное присвоение класса (рис. 8).

Рис. 8. Результат работы модели на 10-й эпохе

Вероятность обнаружения составляет 74,6%, однако мотоцикл так же распознается, как велосипед (рис. 9).

Рис. 9. Результат работы модели на 18-й эпохе

При 26-й эпохе достигается наибольшая вероятность распознавания и составляет 54,5% (рис. 10).

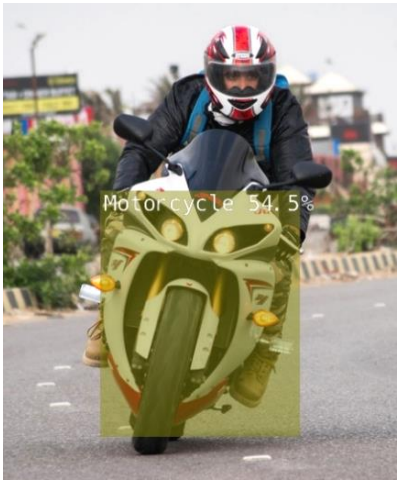


Рис. 10. Результат работы модели на 26-й эпохе

Таким образом, можно сделать вывод, что наилучшие результаты достигаются при максимальном количестве эпох, несмотря на то, что на восемнадцатой эпохе достигнута наименьшая вероятность ошибки распознавания. Наибольшие трудности распознавания происходят при изображениях, где мотоцикл находится анфас. На двадцать шестой эпохе вероятность правильного распознавания варьируется от 54,5 до 80,9%, что можно считать удовлетворительным в силу того, что рекомендуемое количество эпох для обучения составляет 100, что сложно реализуемо на микрокомпьютере Jetson Nano, и необходимо новое решение для обучения. Время, затрачиваемое на обработку, в среднем составляет 30 мс, при разрешении 1 024 на 683 и 1 024 на 779 пикселей, что больше минимально необходимых 24 кадров в секунду для распознавания в реальном времени.

Литература

1. Рахимов К.С. Исследование и реализация технологий распознавания лиц в условиях больших данных // Вестник магистратуры: научный журнал. – 2020. – Ч. 1. – С. 48–49.

2. Нейро-нечеткие методы в интеллектуальных системах обработки и анализа многомерной информации / Т.В. Абрамова, Е.В. Ваганова, С.В. Горбачев, В.И. Сырямкин, М.В. Сырямкин. – Томск: Изд-во ТГУ, 2014. – 442 с.

3. Что такое трансферное обучение? Анализ методов и инструментов реализации трансферного обучения [Электронный ресурс]. – URL: https://russianblogs.com/article/37401375362 (дата обращения: 14.09.2022).

4. Васильев П.А. Развертывание сервера с помощью технологии Docker // Вестник науки и образования. – 2016. – № 12 (24). – С. 25.

Чебанов Виктор Сергеевич
Студент каф. ТУ ТУСУРа
Эл. почта: victorchebanov1@gmail.com

УДК 004.9’31

М.Е. Сукотнова, Г.Е. Эдель, В.В. Капустин

Идентификация дорожных знаков с использованием нейронных сетей

Рассмотрен метод распознавания дорожных знаков с использованием построения сверточной нейронной сети (CNN), также представлены описание топологии сети, принцип ее работы и основные достоинства. Выполнена оценка применения рассмотренного метода на основе полученных зависимостей количества эпох от вероятности определения нужного класса.

Ключевые слова: дорожные знаки, распознавание, вероятность, нейронные сети, сверточные нейронные сети, CNN, алгоритм, эпоха.

Дорожные знаки – ключевой элемент путевой инфраструктуры. Они предоставляют основную информацию о ситуации на дороге для корректировки поведения водителя во время передвижения для соблюдения дорожных правил. С появлением беспилотных автомобилей возникла необходимость не только в формировании единых баз данных дорожных знаков, но также в создании таких интеллектуальных приложений, как автономное вождение, системы помощи водителю, мобильное картографирование и т.д. [1–4].

На сегодняшний день существует множество способов для программного распознавания дорожных знаков.

Распознавание в системах компьютерного зрения – сложный вычислительный процесс, при котором система, на основе обучающей выборки, создает модель, способную соотнести входящее в нее изображение к нужному классу [5–7]. При этом сам процесс распознавания будет меняться в зависимости от выбранного разработчиком способа – он может быть основан как на использовании классических методов, так и на использовании нейронных сетей, которые сегодня считаются одним из самых популярных методов для решения задач классификации, распознавания, локализации и прочих сложных вычислительных компьютерно-технических задач.

В данной статье распознавание происходило с помощью создания нейронной сети. Нейронная сеть – это метод компьютерного зрения, при котором система обрабатывает входящие в нее данные с помощью взаимосвязанных узлов или нейронов. При таком методе система учится на ошибках, возникающих в процессе, постоянно обучаясь на выделенной выборке. Основными достоинствами нейронных сетей является высокая скорость обработки и возможность обучения на любом наборе данных [8]. Нейронные сети делятся на множество видов, каждые из которых имеют свою структуру и способ обучения. В данной статье речь пойдет о сверточной нейронной сети (далее – CNN), основа работы которой – процесс свертки.

Сверточная нейронная сеть является одним из самых популярных видов нейронных сетей, поскольку они инвариантны к изменениям освещения, угла поворота, размеров изображения и прочим возможным видам искажения. Подобная инвариантность достигается за счет трех архитектурных идей:

1. Локальные рецепторные поля, которые обеспечивают локальность двумерных связностей нейронов.

2. Синаптические коэффициенты, которые отвечают за детектирование образов в произвольном (выбранном) участке, уменьшая число весовых коэффициентов.

3. Достаточно точная и выстроенная иерархическая организация слоев с пространственными подвыборками [9].

CNN состоит из нескольких слоев: сверточные слои, в которых происходит операция свертки; субдискретизирующие (подвыборочные) слои, призванные вдвое уменьшать размер карты признаков (матрицы свертки); полносвязный слой (многослойный персептрон), отвечающий за процесс классификации, вычисляя сумму всех входных данных с учетом их весов, при этом оптимизируя и улучшая качество распознавания; и выходной слой, который выполняет функцию распознавания и отвечает за результат работы нейронной сети (выдает «знак» и «незнак»).

Создание модели

Процесс распознавания состоит из нескольких этапов:

– формирование базы данных;
– предобработка изображений;
– построение нейронной сети;
– процесс обучения;
– процесс распознавания.

На рис. 1 представлен алгоритм создания нейронной сети для распознавания дорожных знаков.

Рис. 1. Алгоритм создания нейронной сети

Для данной работы было выбрано три класса дорожных знаков – класс 0 («Остановка запреще-

на»), класс 1 («Осторожно, дети») и класс 2 («Искусственная неровность»). На рис. 2 представлено наглядное разделение изображений на классы.

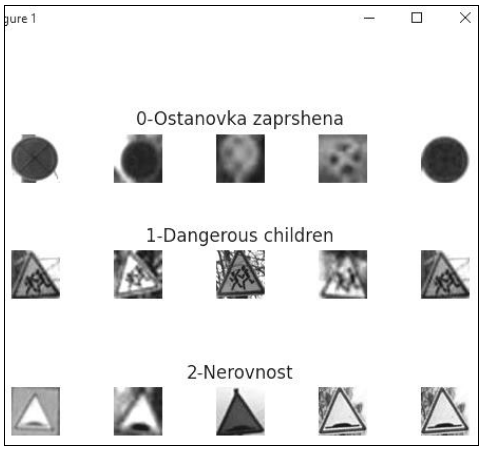


Рис. 2. Пример сформированного класса

Стоит отметить, что нейронная сеть не ограничивает разработчика в количестве классов, однако чем больше классов, тем больше время обучения. В данном случае, чтобы обучить 3 класса, в каждом из которых находилось по 200 изображений, потребовалось 30 мин. На рис. 3 представлен пример одного из сформированных классов.

Рис. 3. Пример сформированного класса

Выбранные изображения были взяты из открытой базы данных дорожных российских знаков. Каждый класс, на котором обучалась система, содержал в себе по 200 изображений необходимого знака при различном освещении, углах поворота, размерах и т.д.

Для получения достоверных результатов необходимо, чтобы помещенные в классы изображения имели единый размер, формат и яркость [10]. Для этого на этапе предобработки изображения эквализируются для стандартизации контрастности, нормализируются (значения от 0 до 1) и масштабируются посредством увеличения, растягивания и вращения. На рис. 4 представлен процесс предобработки, а на рис. 5 – ее результат.

Рис. 4. Процесс предобработки

Рис. 5. Результат предобработки

Таким образом, изображение, представленное на рис. 5, является эталонным и будет использоваться нейронной сетью в качестве обучающего элемента, а также подаваться в готовую обученную модель.

Используя пакет данных sklearn, реализуется разделение данных для обучения и тестирования – т.е. половина входных изображений используется системой в качестве меток (принадлежность к классам), а другая половина – в качестве обучающих элементов.

После процесса предобработки начинается построение CNN. Сверточная нейронная сеть в данном случае состоит из двух сверточных слоев (используется принцип «сканирующего окна» для нахождения признаков объектов), двух подвыборочных слоев, осуществляющих уменьшение размерности матрицы свертки, и выходного слоя. Таким образом, построенная нейронная сеть имеет цельную архитектуру.

Обучение происходило на 30 эпохах (поколениях). Эпоха обучения – определенное количество итераций, которые проходит система. Чем выше число таких итераций, тем больше время обучения. Для реализации задач распознавания рекомендуется использовать 20–30 эпох для обучения. Для доказательства этого, на рис. 6 представлен график точностей, полученный в ходе распознавания. Данный график отражает зависимость точности результатов распознавания от числа эпох.

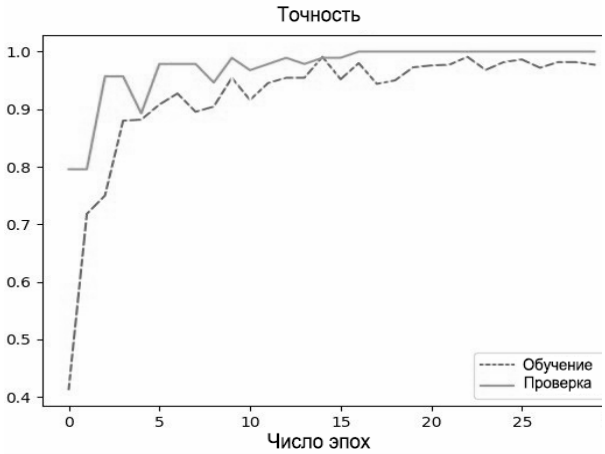


Рис. 6. Зависимость полученной точности от числа эпох

Из рис. 6 видно, что точность, полученная в результате проверки, выше точности, полученной в ходе обучения. К тому же значения конечной точности более стабильны, чем значения, полученные в ходе обучения.

Также для достоверности в правильности выбранного количества эпох был построен график потерь – зависимость количества ложных срабатываний от числа эпох для обучения системы и в ходе проверки, представленный на рис. 7.

Стоит заметить, что «потери» в данном случае – среднее число ложных срабатываний, при которых система не смогла определить знак либо отнести его к правильному классу. Обращаясь к рис. 7, можно

заметить, что процент ложного срабатывания при конечной проверке ниже, чем при обучении, что говорит о верном построении нейронной сети, а также о надежности результатов распознавания.

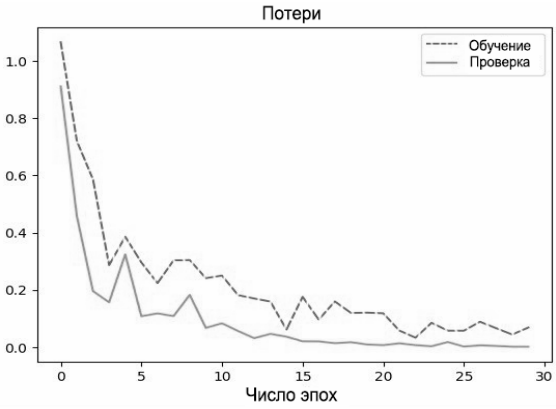


Рис. 7. Зависимость количества эпох от полученного значения потерь

Исходя из графиков, приведенных на рис. 6, 7, можно заметить, что для стабильной точности созданной нейронной сети необходимо не менее 18 эпох.

На рис. 8–10 приведены результаты распознавания. Заметно, что система правильно отнесла каждый из знаков к нужному классу вне зависимости от поворота, при этом вероятность достоверности для каждого из классов варьировалась от 85 до 99,9%.

Рис. 8. Результат распознавания для класса 0 – «Остановка запрещена»

Рис. 9. Результат распознавания для класса 1 – «Осторожно, дети»

Рис. 10. Результат распознавания для класса 2 – «Искусственная неровность»

Стоит отметить, что все входящие в систему данные, будь то видеопоток или изображения, проходили повторную предварительную обработку – эквализацию, масштабирование, т.е. входные данные также приводились в единый с обучающими данными вид.

Для наглядного представления полученных данных была построена таблица средних значений полученного процента вероятности для каждого из выделенных классов.

Среднее значение полученной вероятности для 30 эпох

Номер класса	Итоговый процент распознавания, %	
	Обучение	Проверка
0	75–99	95–99
1	85–99	95–99
2	85–99	95–99

Следует указать, что выбранного количества изображений (по 200 изображений для каждого класса) недостаточно для получения более стабильных и высоких результатов. Важно понимать, что в ходе отбора, на этапе предобработки данных, определенная часть изображений не обрабатывается и выбывает из процесса обучения. Повышение количества изображений в обучающей выборке приведет к получению более стабильных и верных результатов, однако увеличит вычислительную сложность системы и, как следствие, время на ее обучение.

Заключение

Таким образом, результатом обучения сверточной нейронной сети является обученная модель, способная распознать три дорожных знака – «Остановка запрещена», «Осторожно, дети», «Искусственная неровность». Полученные значения достоверности варьируются от 75 до 99% для обучения и от 95 до 99% для проверки, что показывает состоятельность полученной модели: она правильно распознает каждый из обученных классов, при этом значение полученных процентов достоверности в результате распознавания достаточно высоки. Не

исключались также и ложные срабатывания – ситуации, когда система неверно идентифицировала входящий в нее знак. Подобная неправильная работа модели объясняется несовершенством обучающей выборки. Увеличив количество изображений для каждого из классов, модель обучится более правильно, поскольку даже при «отбрасывании» неудачных фотографий системе будет достаточно остаточных образцов для обучения модели.

Исследование выполнено за счет гранта Российского научного фонда № 21-79-10200 в ТУСУРе.

Литература

1. Traffic sign recognition – How far are we from the solution? / M. Mathias, R. Timofte, R. Benenson, L. Van Gool // The 2013 International Joint Conference on Neural Networks. – 2013.

2. Towards fully autonomous driving: Systems and algorithms / J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J.Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, S. Thrun // Intelligent Vehicles Symposium (IV). IEEE. – 2011.

3. Multi-view traffic sign detection, recognition, and 3d localization / R. Timofte, K. Zimmermann, L. Van Gool // Machine Vision and Applications. – December 2011.

4. Chapter 3.5: Combining traffic sign detection with 3d tracking towards better driver assistance / R. Timofte, V.A. Prisacariu, L.J. Van Gool, I. Reid; C.H. Chen, Ed. // Emerging Topics in Computer Vision and its Applications, World Scientific Publishing. – September 2011.

5. Traffic sign recognition and analysis for intelligent vehicles / A. De la Escalera, J.M. Armingol, M. Mata // Image and Vision Computing. – 2003. – Vol. 21(3). – PP. 247–258.

6. Sermanet P., LeCun Y. Traffic sign recognition with multi-scale Convolutional Networks // The 2011 International Joint Conference on Neural Networks. – 2011.

7. Сукотнова М.Е., Эдель Г.Е Использование каскадов Хаара в задачах обнаружения дорожных знаков // Матер. докладов междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР–2021», Томск, 18–20 мая 2022 г. – Томск: В-Спектр, 2022.

8. Bishop C.M. Neural networks and their applications // Review of Scientific Instruments. – 1994. – Vol. 65(6). – PP. 1803–1832.

9. Сверточная нейронная сеть. – Ч. 1: Структура, топология, функция активации и обучающее множество [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/348000/, свободный (дата обращения: 02.09.2022).

10. Traffic Signs Recognition Using CNN & Keras In Python [Электронный ресурс]. – Режим доступа: https://sourcecodehero.com/traffic-signs-recognition-using-cnn-keras-in-python-with-source-code/, свободный (дата обращения: 05.09.2022).

Сукотнова Марина Евгеньевна
Магистрант каф. ТУ ТУСУРа
Эл. почта: msukotnova@mail.ru

Эдель Герман Евгеньевич
Магистрант каф. ТУ ТУСУРа
Эл. почта: german.edel99@gmail.com

Капустин Вячеслав Валериевич
Канд. техн. наук, доцент каф. ТУ ТУСУРа
Эл. почта: peregnun@mail.ru

УДК 004.932.4

А.В. Поднебесный, Н.С. Герасимова, А.С. Захлебин

Исследование влияния изменения размера многоэлементной апертуры пятикаскадного фильтра на его быстродействие

Произведена оценка быстродействия модифицированного пятикаскадного фильтра повышения чёткости изображения в случае изменения размера многоэлементной апертуры. Продемонстрированы результаты оценки быстродействия и графики.

Ключевые слова: цифровая обработка изображений, пятикаскадный фильтр повышения чёткости изображения, быстродействие фильтров.

В нынешних реалиях не всегда удаётся получить изображения отличного качества, с высокой чёткостью, изображения, которыми бы мы остались довольны. Связано это с тем, что на стадии формирования изображения могут воздействовать различные факторы, возникающие в момент получения, хранения или передачи изображения.

Чтобы увеличить четкость изображения, используют различные фильтры повышения четкости изображения. Они могут быть представлены в виде фильтра верхних частот [1].

Проблемы потери качества и чёткости изображения возможно решить с помощью использования адаптивных фильтров. Основной их особенностью является способность адаптироваться в ходе работы к изменениям характеристик входного сигнала, это является важным параметром во многих областях.

Основной целью данной работы является оценка быстродействия модифицированного пятикаскадного фильтра повышения чёткости (ПДС) изображения.

Экспериментальное исследование характеристик модифицированного фильтра при разных размерах апертуры

Тестирование проводилось на платформе со следующими характеристиками: процессор Intel(R) Core (TM) i7-10750H CPU 2,59 ГГц, оперативная память 16 ГБ, видеокарта GTX1650.

На данном этапе работы требуется экспериментально подтвердить не только быстродействие данного адаптивного фильтра, а также исследовать зависимость размера апертуры и количества вычислительных операций. Эксперименты проводились в программном обеспечение MATLAB, а в качестве фильтра для сравнения используется реализованный фильтр классической двумерной свертки (КДС) [2].

Для эксперимента используются изображения разной размерности. В их числе изображения размерности 640×480, 1920×1080, 5616×3744.

Исследование основывается на оценке быстродействия при свертке данных изображений КДС и ПДС. Для оценки зависимости размера апертуры и быстродействия каждое из изображений подлежит свертке различными размерами апертуры: 15×15, 25×25, 49×49, 99×99 [3].

Исследование процесса фильтрации изображения 640×480

Для изображения размером 640×480 проведём работу, подвергаем данное изображение свертке масками размерности 15×15, 25×25, 49×49, 99×99, результаты измерений для оценки быстродействия представлены в табл. 1.

Исследование процесса фильтрации изображения 1920×1080

Для изображения размером 1920×1080 проведём работу, подвергаем данное изображение свертке масками размерностей 15×15, 25×25, 49×49, 99×99, результаты измерений для оценки быстродействия представлены в табл. 2.

Таблица 1

Результаты для изображения 640×480

Результаты для апертуры 15×15											
Время обработки, с											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	0,0749	0,0835	0,0749	0,0635	0,0561	0,0714	0,0884	0,0646	0,0779	0,0629	0,0718
ПДС	0,0056	0,0072	0,0078	0,0081	0,0068	0,0097	0,0075	0,0075	0,0077	0,0088	0,0077
Результаты для апертуры 25×25											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	0,1731	0,1689	0,1638	0,1766	0,1744	0,1598	0,1663	0,1738	0,1756	0,1721	0,1704
ПДС	0,0135	0,0209	0,0196	0,0168	0,0180	0,0194	0,0191	0,0162	0,0208	0,0186	0,0183
Результаты для апертуры 49×49											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	2,6318	2,6238	2,6235	2,6684	2,7348	2,6677	2,6309	2,7804	2,6508	2,7403	2,6719
ПДС	0,0672	0,0589	0,0681	0,0683	0,0589	0,0681	0,0685	0,0714	0,0764	0,0647	0,0715
Результаты для апертуры 99×99											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	4,1119	4,6973	4,3387	4,7802	4,9301	4,0737	4,5474	4,6741	4,4225	4,2119	4,4787
ПДС	0,2993	0,2653	0,2533	0,2685	0,2484	0,251	0,2547	0,2436	0,2509	0,2695	0,2604

Таблица 2

Результаты для изображения 1920×1080

Время обработки, с											
Результаты для апертуры 15×15											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	0,3847	0,4436	0,3948	0,3636	0,3957	0,4369	0,2747	0,2357	0,3143	0,3847	0,3829
ПДС	0,0216	0,0189	0,0192	0,0220	0,0209	0,0276	0,0290	0,0194	0,0223	0,0219	0,0223
Результаты для апертуры 25×25											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	9,4863	9,4685	9,4796	9,2666	9,4735	9,2659	9,3643	9,2658	9,7484	9,2668	9,4086
ПДС	0,1163	0,1453	0,1326	0,1197	0,1134	0,1203	0,1084	0,1372	0,1268	0,1300	0,1250
Результаты для апертуры 49×49											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	19,458	19,573	19,390	19,737	19,316	18,208	18,684	18,437	19,528	18,725	19,106
ПДС	0,444	0,427	0,424	0,400	0,418	0,405	0,437	0,469	0,419	0,396	0,424
Результаты для апертуры 99×99											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	31,484	31,326	31,536	31,604	31,736	31,464	31,385	31,525	31,494	31,745	31,330
ПДС	1,648	1,595	1,679	1,625	1,631	1,577	1,607	1,883	1,624	1,587	1,645

Таблица 3

Результаты для изображения 5616×3744

Время обработки, с											
Результаты для апертуры 15×15											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	7,464	7,838	7,954	7,975	7,874	7,979	7,853	7,945	7,853	7,864	7,460
ПДС	0,772	0,635	0,590	0,596	0,531	0,796	0,692	0,556	0,517	0,637	0,692
Результаты для апертуры 25×25											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	34,464	34,838	34,954	34,975	34,874	34,979	34,853	34,945	34,853	34,864	34,460
ПДС	1,772	1,635	1,590	1,596	1,531	1,796	1,692	1,556	1,517	1,637	1,692
Результаты для апертуры 49×49											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	54,463	54,643	54,296	54,945	54,236	54,838	54,366	54,105	54,847	54,904	54,564
ПДС	4,537	4,424	4,320	4,583	4,357	4,423	4,694	4,552	4,428	4,616	4,543
Результаты для апертуры 99×99											
№	1	2	3	4	5	6	7	8	9	10	СЗ
КДС	85,485	85,235	85,134	85,754	85,235	85,074	85,240	85,956	85,379	85,364	85,584
ПДС	16,615	18,204	19,568	17,593	19,168	19,444	19,838	19,355	19,469	18,302	18,755

Исследование процесса фильтрации изображения 5616×3744

Для изображения размером 5616×3744, хранящегося в формате, проведём работу, подвергаем данное изображение свертке масками размерностей 15×15, 25×25, 49×49, 99×99, результаты измерений для оценки быстродействия представлены в табл. 3.

Анализ полученных результатов исследования

Анализируя полученные результаты, можно сделать вывод, что при увеличении размера изображения и размера апертуры увеличивается время обработки изображения. При сравнении между КДС и ПДС заметно, что ПДС обрабатывает изображение быстрее, чем КДС.

Средние значения времени обработки для изображения размерности 620×480 представлены в табл. 4.

Из полученных данных, представленных в табл. 4, были построены графики зависимости быстродействия от размера апертуры при обработке изображения размером 640×480. Сводный график представлен на рис. 1.

Средние значения времени обработки для изображения размерности 1920×1080 представлены в табл. 5.

Таблица 4

Средние значения времени обработки изображения 640×480

Р-р	Время обработки, с	
	КДС	ПДС
7×7	0,0166	0,00296
15×15	0,071866413	0,0077251
25×25	0,1704	0,0183
49×49	2,6719	0,0715
99×99	4,4787	0,2604

Из полученных данных, представленных в табл. 5, были построены графики зависимости быстродействия от размера апертуры при обработке изображения размером 1920×1080. Сводный график представлен на рис. 2.

Средние значения времени обработки для изображения размерности 5616×3744 представлены в табл. 6.

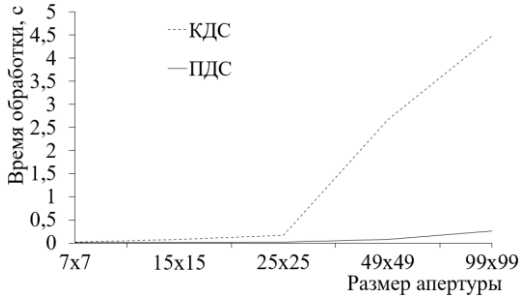


Рис. 1. График зависимости быстродействия от размера апертуры для изображения 640×480

Таблица 5

Средние значения времени обработки изображения 1920×1080

Р-р	Время обработки, с	
	КДС	ПДС
7×7	0,2576	0,0125
15×15	0,3829	0,0223
25×25	9,4086	0,1250
49×49	19,1062	0,4246
99×99	31,3306	1,6459

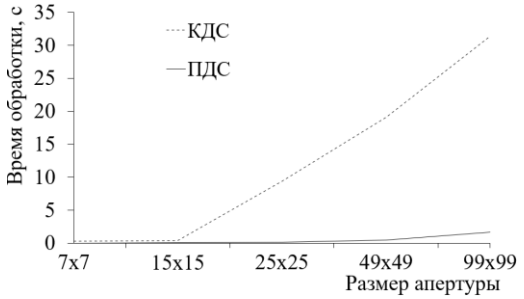


Рис. 2. График зависимости быстродействия от размера апертуры для изображения 1920×1080

Таблица 6

Средние значения времени обработки изображения 5616×3744

Р-р	Время обработки, с	
	КДС	ПДС
7×7	3,1048	0,27617
15×15	7,4603	0,6927
25×25	34,4603	1,6927
49×49	54,5648	4,5438
99×99	85,5849	18,7559

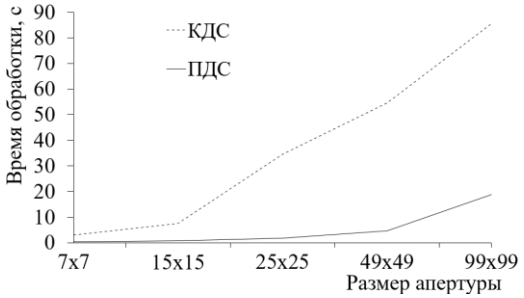


Рис. 3. График зависимости быстродействия от размера апертуры для изображения 5616×3744

Из полученных данных, представленных в табл. 6, были построены графики зависимости быстродействия от размера апертуры при обработке изображения размером 5616×3744. Сводный график представлен на рис. 3.

Анализируя полученные графики, видно, что КДС затрачивает больше времени, чем ПДС.

Заключение

В процессе выполнения поставленной задачи выяснилось, что чем больше размер изображения, тем больше необходимо времени на выполнения обработки, а также при увеличении значения апертуры растёт время обработки изображений.

Из этого следует вывод, что для обработки изображений удобным и эффективным в вопросе быстродействия является использование данного адаптивного пятикаскадного фильтра, это же касается областей, где одним из важных аспектов является уровень быстродействия вычислительных операций.

К недостаткам классической двумерной свертки относится ручное введение маски определенного размера апертуры, что в данном случае дополнительно требовало лишнего времени. Классическая свертка использует уже готовую маску и тем не менее тратит больше времени, нежели адаптивные фильтры, которые строят эту же маску в процессе обработки изображения.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования РФ (проект № FEWM-2020-0039 от 01.03.2020 г.).

Литература

1. Курячий М.И. Цифровая обработка сигналов: учеб. пособие. – Томск, 2002. – 100 с.

2. Комарницкий М.А. Оценка быстродействия двумерного рекурсивно-сепарабельного пятикаскадного фильтра повышения четкости изображений // Матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР–2022». – Томск: В-Спектр, 2022. – Ч. 1. – С. 99–102.

3. Kamenskiy AV. High-speed recursive-separable image processing filters // Computer Optics. – 2022. – Vol. 46(4). – PP. 659–665. DOI: 10.18287/2412-6179-CO-1063.

Поднебесный Андрей Владимирович
Магистрант каф. ТУ ТУСУРа
Эл. почта: andrey_podnebesniy36071@mail.ru

Герасимова Наталья Сергеевна
Магистрант каф. ТУ ТУСУРа
Эл. почта: natashageraismova@mail.ru

Захлебин Александр Сергеевич
М.н.с. каф. ТУ ТУСУРа
Эл. почта: aerosnimok@gmail.com

УДК 004.932

А.А. Киселева

Исследование методов построения карт глубин по стереоизображениям

Представлены измерения дальности по экспериментальным изображениям с использованием стереокамеры. Исследуются результаты двух методов построения карты глубины с помощью классов функций «StereoBM» и «StereoSGBM».

Ключевые слова: карта глубины, изображение, стереопара, класс функций, расстояние, камера, ректификация.

Цифровые камеры обеспечивают быстрое и простое получение изображения. Однако большинство популярных цифровых камер способны снимать только двумерный (2D) вид сцены, тогда как компоненты, отвечающие за глубину, не записываются. Для обычного пользователя 3D-изображения могут быть созданы путем извлечения информации о глубине из двумерного изображения с использованием различных стереокамер. Этот метод наиболее популярен для генерации карты глубины из стереопары изображений и находит множество применений в различных областях.

Карта глубины – это 2D-изображение с кодировкой черно-белового изображения, которое дает представление о глубине по интенсивности цвета. Темные участки на карте глубины говорят о том, что объект расположен далеко от камеры, а этот темный цвет будет плавно уменьшаться до более яркого с уменьшением глубины и, наконец, станет белым для более близких объектов [1].

Необходимым для создания карты глубины по стереопаре является решение трех задач.

1. Захват изображений.

Предлагаемый алгоритм использует два изображения одной и той же сцены. Левое и правое изображения могут быть сделаны с помощью двух цифровых камер. Они захватывают изображения одной и той же сцены в одно и то же время. Эти камеры смещены на некоторое расстояние по горизонтали. Для каждого набора изображений положение камеры должно тщательно контролироваться относительно высоты и вектора обзора камеры.

2. Предварительная обработка изображений.

Алгоритм находит совпадающие пиксели, сравнивая компоненты RGB-пикселей на обоих изображениях.

3. Оценка глубины.

Оценка глубины – это вычисление глубины различных объектов в сцене из стереопары.

Необходимо определить идентичные пиксели на изображении, называемые точками соответствия, опознающие одинаковые 3D-точки (рис. 1).

По одному пикселю левого изображения происходит поиск этого же пикселя на правом изображении. Однако выбранный пиксель на правом изображении должен иметь такие координаты, в которых по оси абсцисс вычтена величина, называемая смещением (рис. 2, 3) [2].

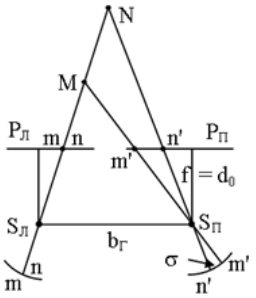


Рис. 1. Стереомодель

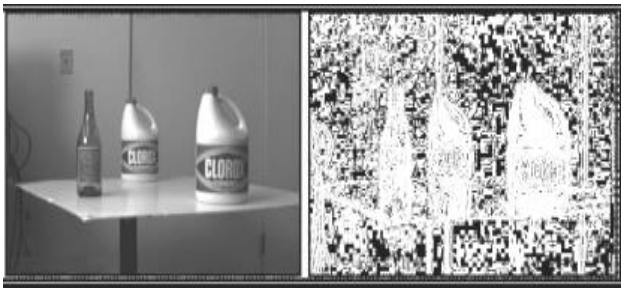


Рис. 2. Карта глубины

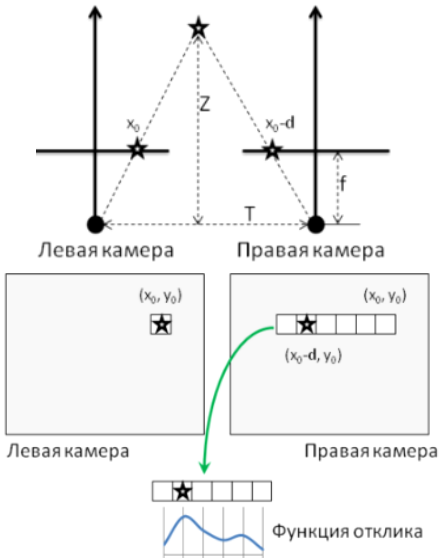


Рис. 3. Вычисление карты глубины

Когда в каждом пикселе посчитано смещение, то найти глубину можно из следующего выражения:

$$D = base \times (focus / distance), \quad (1)$$

где *focus* – фокусное расстояние камеры, *base* – база стереосистемы в метрах, *distance* – дальность до объектов интереса.

Так как расстояние от центра линзы до основных фокусных точек и расстояние между камерами неизменны, то смещение воспринимается как условная глубина точек. Следовательно, карта глубины может быть построена по известной глубине точек.

По стереопаре изображений измерить расстояние до объекта лучше до ближних, чем до дальних. Если погрешность измерений стабильна, то верность нахождения глубины будет расти при увеличении значения базы. Когда объект расположен далеко от снимаемой камеры, изображения мало схожи [3]. Если обнаруживается, что разница между сравниваемыми пикселями меньше заранее заданного допуска, то эти пиксели рассматриваются алгоритмом как «совпадающая пара» пикселей. Затем для совпадающих пикселей вычисляется бинокулярное несоответствие, которое в дальнейшем используется для оценки информации о глубине.

Ректификация стереопары

После получения виртуального перспективного изображения стереоректификация может быть достигнута путем отображения изображений в общую плоскость и выравнивания столбцов с использованием матриц поворота и перемещения. Эти матрицы могут быть получены путем разложения фундаментальной матрицы. Два двумерных проективных преобразования применяются к стереопаре для схождения с эпиполярными линиями. Они могут быть взяты так, чтобы схожие точки также имели примерно одинаковую координату *x*. Следовательно, наложенные двумерные изображения будут совпадать, а расхождения будут параллельны оси *x* (рис. 4).

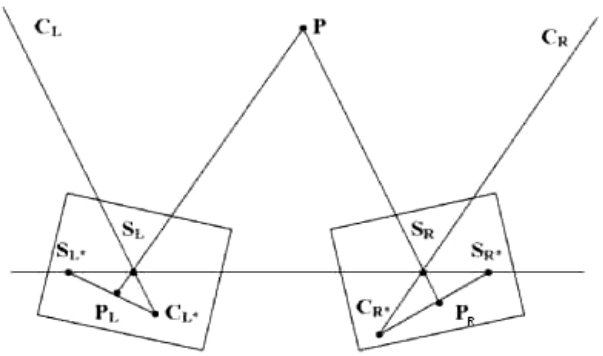


Рис. 4. Преобразование к эпиполярной стереопаре

Когда камеры созависимы друг от друга, их оптические оси параллельны и расположены друг от друга на некотором расстоянии, то воспроизведение сцены будет простым. Однако невозможно выставить стереокамеры идеально. Для этого нужно ставить камеры так, чтобы все исследуемые объекты находились в их поле зрения и оптические оси совпадали друг относительно друга в области расположения объектов сцены.

Однако при разделении двух камер, расположенных близко с их параллельными главными осями, могут возникать некоторые несоответствия на изображениях. И несоответствия становятся причинами, которые ухудшают производительность карты предполагаемых несоответствий. После этого исправления поиск совпадающих точек значительно упрощается благодаря простой эпиполярной структуре и близкому соответствию двух изображений (рис. 5) [4].

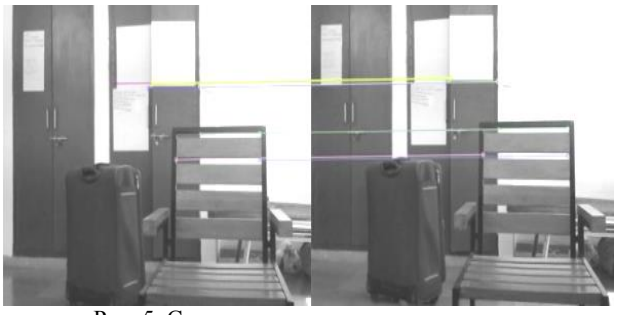


Рис. 5. Соответствие точек на стереопаре

Построение карты глубины

Для начала необходимо получить стереоизображения для заданных значений стереобазы. Используемая база равна 50 мм. Предметами, для которых будет реализована карта глубины, являются книга и огнетушитель. Устанавливаем на видеокамеру VAC-136-USB объектив с нужным фокусным расстоянием, в нашем случае это объектив RF 4 мм, и проводим настройку камеры в программе EvsCap3 (рис. 6) [5].

Рис. 6. Видеокамера VAC-136-USB, установленная на подвижной оси

Измеренное расстояние лазерной рулеткой от объектива камеры до исследуемых объектов представлено в табл. 1.

Таблица 1

Результаты измерения дальности объектов, выставленные на реальной сцене

Объект	Положение объекта, измеренное лазерным дальномером, см
Огнетушитель	206
Книга	322

Сохраняем первый кадр на начальном положении камеры, это будет нашим правым изображением. Далее в терминале задаем положение оси x в миллиметрах согласно длине базы ×50, что сдвинет камеру на 50 мм. Сохраняем второй кадр, это будет левым изображением (рис. 7).

а *б*

Рис. 7. Снятое изображение правого (*а*) и левого положения (*б*)

Далее будем рассматривать два метода построения карты глубины с помощью классов функций «StereoBM» (рис. 8) и «StereoSGBM» (рис. 9) с использованием модуля «OpenCV» [6].

Как видно из рис. 8, карта глубины при использовании класса функций «StereoBM» получилась с дефектами, т.е. неточной и с большим количеством шумов, рассмотреть каждый объект трудно [7].

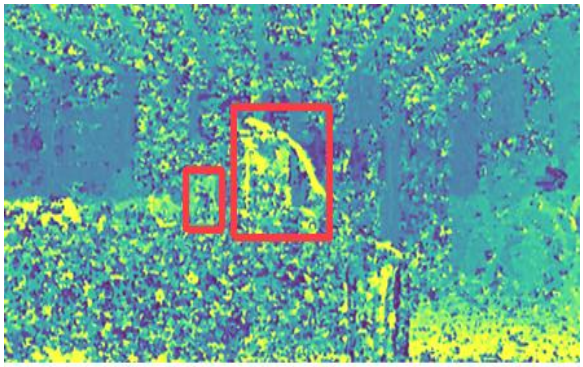


Рис. 8. Карта глубины при использовании «StereoBM»

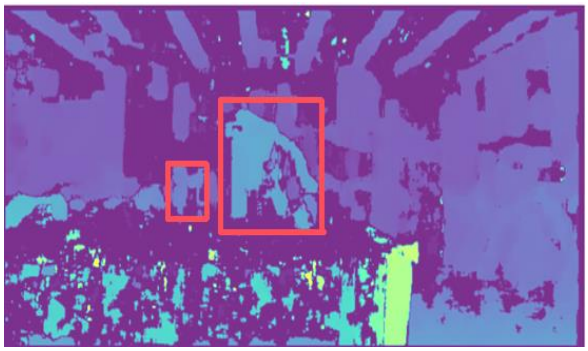


Рис. 9. Карта глубины при использовании StereoSGBM

Получение карты глубины, применяя класс функций «StereoSGBM», является более точным, границы объектов выделяются четче, а разница между оттенками цвета более явная [8].

Для повышения качества карты глубины следует выполнить ректификацию некалиброванной камеры, чтобы все эпиполярные линии были параллельны оси абсцисс и на обоих изображениях имели одинаковые ординаты. Для этого необходимо:

– найти не менее пяти хорошо подобранных точек на обоих изображениях, которые будем использовать для вычисления фундаментальной матрицы;

– вычислить фундаментальную матрицу **F** при помощи findFundamentalMat;

– исправить искажения изображения с помощью stereoRectifyUncalibrated и warpPerspective [9];

– вычислить диспаритет (карта глубины) с StereoSGBM (рис. 10, 11).

Рис. 10. Ректификация изображений

Можно заметить, что исследуемые объекты стали более отчетливыми. Добавим шкалу дальности для определения расстояния от камеры до объектов и снимем расстояние до объектов (рис. 12).

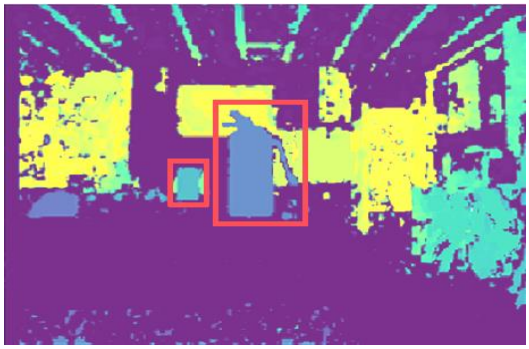


Рис. 11. Карта глубины с ректификацией

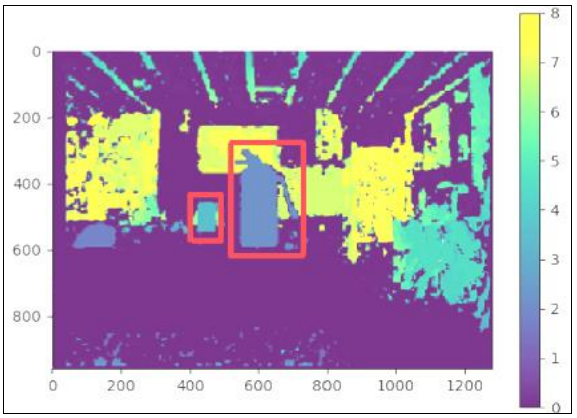


Рис. 12. Карта глубины с выделенными объектами, до которых измеряется расстояние

Проанализировав полученные результаты, представленные в табл. 2, было выяснено, что погрешность измерения небольшая.

Таблица 2

Сравнение полученных результатов измерения дальности объектов на реальной сцене и экспериментальных изображениях

Объект	Положение объекта, измеренное лазерным дальномером, см	Положение объекта, измеренное с помощью стереокамер, см
Огнетушитель	206	204
Книга	322	326

Литература

1. Стереопара, стереоскопическая модель [Электронный ресурс]. – Режим доступа: https://studopedia.ru/10_250162_ stereopara-i-stereoskopicheskaya-model.html, свободный (дата обращения: 09.09.2022).

2. Карта глубины [Электронный ресурс]. – Режим доступа свободный https://neerc.ifmo.ru/wiki/index.php?title=Карта_глубины (дата обращения: 09.09.2022).

3. Хорн Б.К.П. Зрение роботов. – Электронная книга, 1989. – 296 с.

4. Козулин Ю.А., Белкин Е.С. Анализ качества стереореконструкции модельной облачной сцены // Электронный журнал «Исследовано в России». – 2002. – 1556 с. (дата обращения: 09.09.2022).

5. Телевизионная камера VAC-136-USB [Электронный ресурс]. – Режим доступа свободный http://skud-russia.ru/shop/korpusnye_kamery/televizionnaya_kamera_vac_136_usb/ (дата обращения: 10.09.2022).

6. Документация по OpenCV [Электронный ресурс]. – Режим доступа: https://docs.opencv.org/master/dd/d53/tutorial_py_depthmap.html, свободный (дата обращения: 11.09.2022).

7. Документация по OpenCV [Электронный ресурс]. – Режим доступа: https://docs.opencv.org/master/d9/dba/classcv_1_1StereoBM.html, свободный (дата обращения: 11.09.2022).

8. Документация по OpenCV [Электронный ресурс]. – Режим доступа: https://docs.opencv.org/4.3.0/d2/d85/classcv_1_1StereoSGBM.html, свободный (дата обращения: 11.09.2022).

9. OpenCV – Depth map from Uncalibrated Stereo System [Электронный ресурс]. – Режим доступа: https://waksoft.susu.ru/2021/01/26/sozdanie-nedorogoj-stereokamery-s-ispolzovaniem-opencv/, свободный (дата обращения: 11.09.2022).

Киселева Анастасия Алексеевна
Студентка каф. ТУ УСУРа
Эл. почта: anastasiyak2301@mail.ru

УДК 004.932.72'1

И.Д. Мусихин, В.В. Капустин

Исследование методов поиска особых точек для автоматического слежения за объектами интереса

Проводится сравнительный анализ методов поиска особых точек SIFT, AKAZE, BRISK и ORB. Также производится оценка методов трекинга CSRT, KCF и GOTURN. Критериями оценки служат быстродействие, эффективность и количество найденных особых точек с использованием исследуемых методов.

Ключевые слова: особые точки, дескрипторы, сопоставление особых точек, SIFT, AKAZE, BRISK, ORB, CSRT, KCF, GOTURN.

Сравнение изображений – это первостепенная задача компьютерного зрения. С помощью сопоставления изображений можно получить различные характеристики объектов и использовать полученные данные для автоматического слежения за объектами интереса.

Целью работы является исследование методов поиска особых точек и методов трекинга для автоматического отслеживания положения объекта в видеопотоке. Работа является актуальной, потому что автоматическое слежение за положением объекта интереса может быть использовано в различных сферах. Полученные результаты исследований можно использовать в видеонаблюдении, в телестудиях или в активно-импульсных телевизионных измерительных системах, которые применяются для видения в сложных погодных условиях [1].

Методы поиска особых точек

В работе выполнено исследование следующих методов поиска особых точек: SIFT, AKAZE, BRISK, ORB. Рассматриваемые алгоритмы используются для сравнения изображений, а также могут быть применены для трекинга объектов в видеопотоке.

Алгоритм SIFT [2] использует функцию ядра Гаусса для фильтрации при построении масштабного пространства, которое представлено пирамидой Гаусса. Чтобы найти особые точки, строится пирамида разностей Гаусса, после чего каждый пиксель сравнивается со всеми его соседями.

AKAZE [3] основан на применении алгоритма Fast Explicit Diffusion [4] на пирамидальной схеме. Для каждой октавы в пирамиде вычисляется определитель Гессиана, производные второго порядка вычисляются с помощью фильтра Шарра [5]. Особые точки ищутся в октаве, значение фильтра которой выше заданного порога, после чего сравниваются относительно результатов в соседних октавах.

Идея алгоритма BRISK [6] основана на выборе наилучшей особой точки с максимальным значением интенсивности в пирамиде, которая состоит из 4 внешних октав и 4 внутренних. Поиск особых точек в октавах осуществляется детектором FAST.

Алгоритм ORB [7] основан на вычислении ориентации особой точки при помощи быстрого древовидного FAST на исходном и на нескольких уменьшенных изображений. Для обнаруженных точек вычисляются мера Харриса и угол ориентации.

Методы трекинга

KCF расшифровывается как Kernelized Correlation Filters. Этот трекер использует тот факт, что множественные положительные образцы, используемые в трекере MIL, имеют большие перекрывающиеся области. Эти перекрывающиеся данные приводят к некоторым математическим свойствам, которые используются этим трекером, чтобы сделать отслеживание более быстрым и точным одновременно. Основной идеей отслеживания корреляционного фильтра является оценка оптимального фильтра изображения таким образом, чтобы фильтрация с входным изображением давала желаемый отклик. Желаемый отклик обычно имеет форму Гаусса, центрированную в целевом месте, поэтому оценка уменьшается с расстоянием [8].

Фильтр обучается из переведенных (сдвинутых) экземпляров целевого исправления. При тестировании оценивается отклик фильтра и его максимум выдает новое положение цели. Фильтр обучается в режиме online и обновляется последовательно с каждым кадром, чтобы трекер адаптировался к умеренным изменениям цели.

В методе CSRT для отслеживания используется карта пространственной надежности для настройки фильтра на часть выбранной области из кадра [9]. Это обеспечивает укрупнение, локализацию выбранного региона и улучшенное отслеживание непрямоугольных областей или объектов. Трекер также работает при сравнительно более низком fps (25 кадров в секунду), но дает более высокую точность для отслеживания объектов.

Показатели надежности отражают качество исследуемых фильтров по каналам и используются в качестве весов для локализации. Таким образом, используя HoGs и Colornames в качестве наборов функций, алгоритм работает относительно хорошо.

Трекер GOTURN, или Generic Object Tracking Using Regression Networks, представляет собой алгоритм отслеживания на основе глубокого обучения. Модель GOTURN обучается на тысячах видеопоследовательностей и не требует выполнения какого-либо обучения во время выполнения [10].

В первом кадре известно местоположение объекта, кадр вокруг объекта обрезается в два раза больше размера ограничительной рамки. Объект в первой обрезанной рамке всегда центрирован.

Необходимо предсказать местоположение объекта во втором кадре. Ограничительная рамка, используемая для обрезания первого кадра, используется и для обрезания второго кадра. Поскольку объект мог переместиться, он не центрируется во втором кадре. Сверточная нейронная сеть (CNN) обучена предсказывать расположение ограничительной рамки во втором кадре.

Экспериментальные исследования

Для исследования методов поиска особых точек и методов трекинга были проведены тесты при различных внешних условиях. Все эксперименты проводились на одном устройстве – ноутбуке Acer Aspire V3 771g 53234g50makk.

Основные параметры:

– процессор Intel Core i5-3230M;
– тактовая частота (MHz): 2600;
– 2 ядра, 4 логических процессора;
– кэш 2-го уровня (Kb): 512;
– кэш 3-го уровня (Kb): 3072;
– SDD-диск с интерфейсом SATA;
– видеокарта NVIDIA GeForce 710M 2ГБ;
– экран 17.3"/1600x900 пикс.;
– оперативная память: 8 Гб;
– частота ОЗУ: 1333 МГц.

Для экспериментов использовалась камера EVS VAC-136-USB. Черно-белая мегапиксельная телевизионная камера в корпусе внутреннего исполнения, модель VAC-136-USB, выполнена на многослойных печатных платах [11].

Реализация методов поиска особых точек и методов трекинга выполнялась с использованием ПО PyCharm, языка программирования Python, библиотек OpenCV и time. В библиотеке OpenCV содержатся необходимые функции для обработки изображений [12], а библиотека time используется для оценки времени работы алгоритмов.

Как уже было отмечено в статье [13], для реализации алгоритма автоматического трекинга объектов из рассмотренных методов поиска особых точек на изображениях по таким параметрам, как быстродействие и число сопоставлений особых точек, ORB является наиболее подходящим.

Для начала было проведено исследование на неподвижном объекте с множеством особых точек в виде таблички с текстом и рисунком. Используя метод ORB, было найдено 1 140 особых точек при разрешении съемки 640×480 элементов, что соответствует 0,3 Мп.

Было измерено расстояние, на котором объект интереса не будет найден алгоритмом. На камере установлен объектив computar с фокусным расстоянием 8 мм. Объект изначально был захвачен на расстоянии 85 см. При таком объективе объект перестает распознаваться на расстоянии 3 м.

С увеличением расстояния количество пикселей, приходящихся на объект, уменьшается. Из-за этого уменьшается разрешение объекта и уменьшается количество обнаруживаемых особых точек. Так, на расстоянии 85 см объект имеет размер 268×476 пикселей, а на расстоянии 3 м – 78×138 пикселей, что эквивалентно уменьшению размера в почти в 3,5 раза или уменьшению занимаемой площади кадра почти в 12 раз.

Разрешение съемки было повышено до 1280×720 элементов, объектив был заменен на computar с фокусным расстоянием 4 мм. Минимальное расстояние для захвата и обнаружения объекта равно 30 см с учетом того, что объект занимает почти весь кадр. При таком объективе объект, в роли которого выступает лицо, захваченный на расстоянии 40 см, перестает распознаваться с помощью методов поиска особых точек на расстоянии 3,5 м. Кадр с обнаружением с помощью метода ORB и трекера CSRT приведен ниже на рис. 1.

Рис. 1. Максимальное расстояние обнаружения

Было проведено исследование зависимости частоты кадров от используемого метода обнаружения и отслеживания. Видеокамера изначально передает кадры с частотой 30 кадров/с, программа может считывать и выводить кадры с частотой 26 кадров/с. Потери составляют 4 кадра, или, по-другому, программа отстает на 5 мс. В табл. 1 приведена зависимость частоты кадров от используемого метода обнаружения и отслеживания объекта.

Таблица 1

Зависимость частоты кадров в Гц от используемого метода обнаружения и отслеживания объекта

Метод	Трекер		
	CSRT	KCF	GOTURN
SIFT	17	24	15
AKAZE	18	25	17
BRISK	19	25	16
ORB	22	25	19

Для более наглядного представления по данным табл. 1 был построен график, представленный на рис. 2.

Как видно из табл. 1 и рис. 2, метод ORB показывает большую частоту обработки кадров по отношению к другим методам независимо от выбранного метода трекинга.

Среди представленных методов отслеживания, по скорости работы наилучший метод KCF, однако, KCF менее надежен, чем CSRT. GOTURN хуже всех методов трекинга по производительности, но лучший по точности. Для этого трекера ключевую роль

в точности и производительности играет набор предобучения, так что потенциально в отслеживании конкретных объектов GOTURN может достигать высокой точности при условии грамотного предобучения.

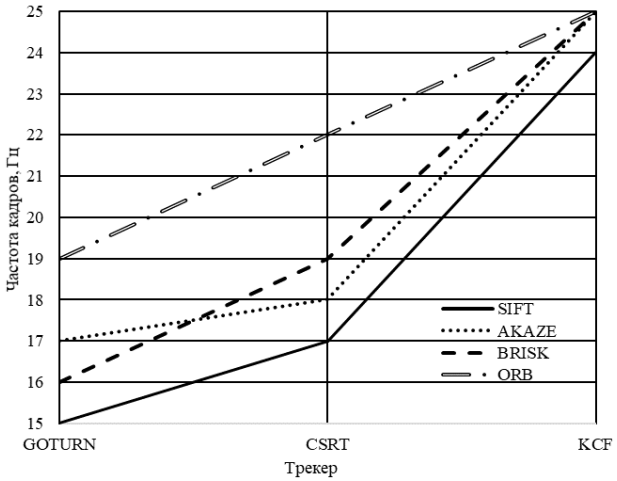


Рис. 2. Зависимость частоты кадров от используемого метода обнаружения и отслеживания объекта

Было проведено исследование зависимости найденных особых точек от величины задаваемой погрешности. В данном случае погрешность означает максимальное значение отклонения найденной особой точки от эталонной. Максимальное значение погрешности для точного сопоставления особых точек равно 70%, все точки, погрешность которых выше этого значения, учитывать нельзя. За эталон взяты точки, найденные при выборе объекта в первом кадре. На рис. 3 приведен объект, для которого находились особые точки.

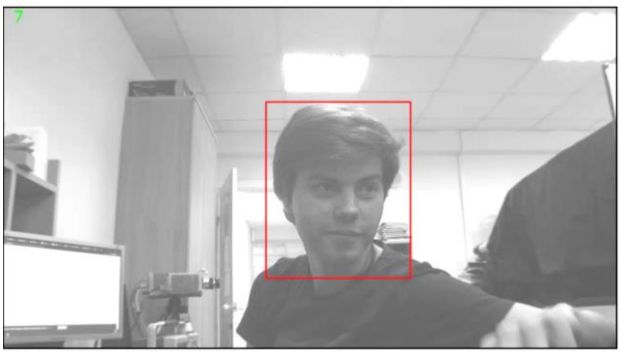


Рис. 3. Исследуемый объект

Полученные данные приведены в табл. 2, по этим данным построен график, приведенный на рис. 4.

Таблица 2

Зависимости найденных особых точек от величины задаваемой погрешности

Метод поиска	Точки объекта	Погрешность, %										
		0	10	20	30	40	50	60	70	80	90	100
SIFT	95	7	35	43	49	55	56	59	66	69	67	70
AKAZE	105	3	12	28	30	33	45	47	49	53	60	68
BRISK	97	7	40	58	69	75	80	88	90	96	97	97
ORB	287	9	67	120	165	196	221	237	253	265	268	275

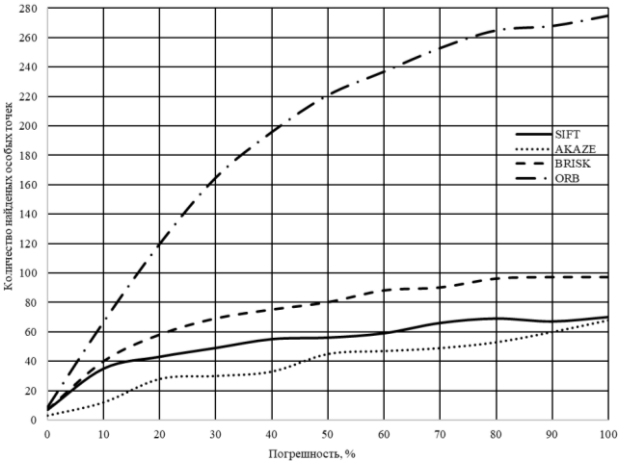


Рис. 4. Зависимости найденных особых точек от величины задаваемой погрешности

Как видно из табл. 2 и рис. 4, метод ORB показывает лучшие результаты. При использовании ORB в эталонном кадре найдено больше всего точек – 287, остальные методы находят примерно одинаковое количество особых точек. Также можно заметить, что при увеличении погрешности у метода ORB наибольшая величина нарастания найденных особых точек – примерно 70 точек/10%, у остальных методов – примерно 10–15 точек/10%. Метод AKAZE имеет худший результат по сравнению с другими методами.

Заключение

В результате проведения экспериментов при нормальном освещении (примерно 300 лк) было обнаружено, что минимальное расстояние обнаружения объекта равно 30 см, а максимальное расстояние – 3,5 м. Было обнаружено, что метод ORB показывает большую частоту обработки кадров по отношению к другим методам независимо от выбранного метода трекинга. Среди методов трекинга по скорости работы наилучший метод KCF, однако, он менее надежен, чем CSRT. GOTURN хуже всех методов трекинга по производительности, но лучший по точности за счет предобучения. Среди методов поиска особых точек лучший результат показывает метод ORB, который находит больше всего точек.

Исследование выполнено за счет гранта Российского научного фонда № 21-79-10200 в ТУСУРе.

Литература

1. Мовчан А.К. Методы и алгоритмы прецизионного измерения дальности активно-импульсными телевизионными измерительными системами / А.К. Мовчан и др. // Доклады ТУСУР. – 2020. – Т. 23, № 2. – С. 7–14.

2. Lowe D.G. Distinctive image features from scale-invariant keypoints. // International Journal of Computer Vision. – 2004. – Vol. 60, Iss. 2. – PP. 91–110.

3. Alcantarilla P. Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces. / P. Alcantarilla., J. Nuevo, A. Bartoli // Procedings of the British Machine Vision Conference. – 2013. – Vol. 13. – PP. 13.1–13.11.

4. Cyclic Schemes for PDE Based Image Analysis / S. Grewenig, J. Weickert, C. Schroers, A. Bruhn // International Journal of Computer Vision. – 2013.

5. Weickert J. A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance / J. Weickert, H. Scharr // Journal of Visual Communication and Image Representation. – 2002. – PP. 103–118.

6. Leutenegger S. BRISK: Binary Robust invariant scalable keypoints / S. Leutenegger, M. Chli, R.Y. Siegwart // Computer Vision (ICCV) IEEE International Conference. – 2011. – PP. 2548–2555.

7. ORB: an efficient alternative to SIFT or SURF / E. Rublee, V. Rabaud, K. Konolige, G. Bradski // IEEE International Conference on Computer Vision. – 2011. – Vol. 58(11). – PP. 2564–2571.

8. High-Speed Tracking with Kernelized Correlation Filters / J. Henriques, R. Caseiro, P. Martins, J. Batista // Transactions on Pattern Analysis and Machine Intelligence. – 2014. – Vol. 37(3). – PP. 583–596.

9. Desai L. Crypto-Stego-Real-Time (CSRT) System for Secure Reversible Data Hiding / L. Desai, S.N. Mali // VLSI Design. – 2018. – № 4804729. – PP. 1–8.

10. Held D. Learning to Track at 100 FPS with Deep Regression Networks / D. Held, S. Thrun, S. Savarese // Proceedings of European Conference on Computer Vision. – 2016. – PP. 749–765.

11. Модель VAC-136-USB [Электронный ресурс]. – URL: https://www.evs.ru/d_sheet/VAC-136-USB.pdf (дата обращения: 05.09.2022).

12. OpenCV Online Documentation, 2021 // OpenCV: OpenCV modules [Электронный ресурс]. – Режим доступа: https://docs.opencv.org/4.5.5/index.html, свободный (дата обращения: 31.08.2022).

13. Мусихин И.Д. Исследование методов поиска особых точек на изображениях для автоматического трекинга объектов / И.Д. Мусихин, В.В. Капустин // Видеоинформационные технологии: матер. избр. статей междунар. науч.-практ. конф. «Научная сессия ТУСУР–2022»: в 3 ч. – 2022. – Ч. 1. – С. 61–65.

Мусихин Илья Денисович
Магистрант каф. ТУ ТУСУРа
Эл. почта: musilya333@yandex.ru

Капустин Вячеслав Валериевич
Канд. техн. наук, доцент каф. ТУ ТУСУРа
Эл. почта: vk@tu.tusur.ru

УДК 004.855.5

С.А. Сальников

Обучение моделей нейронных сетей для автоматического обнаружения дорожных знаков

Обучена нейронная сеть для автоматического распознавания дорожных знаков. Проведены исследования зависимости вероятности распознавания дорожных знаков от числа эпох обучения нейронной сети.
Ключевые слова: нейронные сети, эпоха, свертка, библиотека, оптимизаторы, обучение, модель, выборка.

В основе архитектур видеонаблюдения лежит анализ, первым шагом которого является распознавание изображений. Классифицировать нейронные сети можно по задачам, с которыми они работают, так, можно выделить многослойные нейронные сети, которые обрабатывают лишь данные, связанные с числами, и сверточные нейронные сети, которые работают непосредственно с изображениями для определения того или иного объекта, и т.д. Также любая нейронная сеть подвержена переобучению, что пагубно сказывается на ее функциях. Говорят, о переобучении тогда, когда искусственная нейронная сеть (ИНС) по итогам обучения научилась хорошо распознавать примеры из обучающей выборки, но не обретает свойство резюмирования. Иными словами, она не справляется с задачей распознавания любых других примеров изображения, кроме тех, на которых она была обучена. Поэтому зачастую чтобы избежать переобучения, выборку делят на тестовую и обучающую [1].

Классифицировать такие методы обучения можно на две группы, обучение с учителем и обучение без учителя. Говоря о нейронных сетях, обученных с учителем, подразумевается то, что для любого входного вектора существует так называемый целевой вектор, иначе говоря, выход, в этимологии языка это называют обучающей парой. Векторы из обучающегося множества располагаются последовательно, в которых происходит вычисление ошибок, после которого вся полученная информация переносится в весах до тех пор, пока минимизация ошибки не достигнет некоторого удовлетворительного уровня. Обучение без учителя не предполагает сравнения с уже известными верными ответами, таким образом, нейронная сеть сама учится находить некие сходства и закономерности в ряде множества изображений, на основе которых и составляет некие цепочки, по которым и происходит распознавание объекта.

Нейронные сети можно разделить на полносвязные и многослойные. В полносвязных – выходной сигнал передается от каждого нейрона другому. В многослойных – группы нейронов образуют слои. В каждом слое есть набор нейронов, имеющих единые выходные сигналы. Количество нейронов в каждом слое может быть разным, независимо от того, сколько нейронов содержится в последующих слоях [2]. Перед тем как распознавать изображения, нейронную сеть нужно обучить. Для этого используются наборы данных, т.е. для того, чтобы нейронная сеть успешно выполняла свою задачу по распознаванию определенного объекта на изображении, для начала нужно показать ей другие примеры изображений с этим объектом. Все нейронные сети состоят из слоев, существует всего 3 типа основных слоев: входной, скрытый и выходной. Во входном слое располагаются нейроны, задача которых принять сигнал и направить его дальше через синапсы, здесь не происходит никакой обработки входной информации. Синапсы – это связь между двумя нейронами, в случае нейронных сетей синапсы имеют вес, благодаря которому входная информация претерпевает изменения, когда происходит передача от одного слоя к другому [3].

Как можно видеть на рис. 1, с выхода каждого нейрона через синапсы w_1, w_2, w_3 идет вес на вход другого нейрона, после прихода новой информации, содержащейся в весах, следующий слой нейронов, выбирает из всего перечня приходящей в него информации ту, которая имеет наибольшей вес, она будет являться более весомой. Скрытый слой как раз-таки включает в себя нейроны, которые обрабатывают данные, полученные с предыдущих слоев, а после обработки передают всю полученную информацию на выходной слой. Изображение на рис. 2 схематично показывает архитектуру нейронных сетей. Как можно видеть, скрытый слой может состоять из сколь угодно различных слоев, но вход и выход всегда состоят из одного слоя [4].

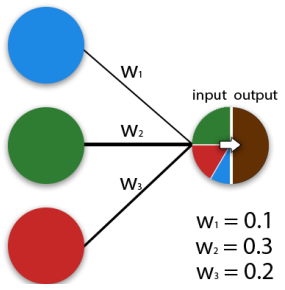


Рис. 1. Распределение весов на примере смешения цветов

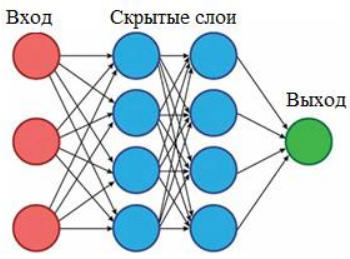


Рис. 2. Архитектура нейронной сети

Существует множество различных слоев, которые участвуют в процессе обучения, из основных можно выделить слои типа Danse, Dropout, Flatten, Conv2D, BatchNormalization. По большей части данные слои являются вспомогательными, а поэтому зачастую и используются при написании большинства моделей нейронной сети. Одной из немаловажных составляющих нейронных сетей является алгоритм обратного распространения ошибки. Обучение с его использованием подразумевает под собой два прохода информации по всем имеющимся слоям в прямом и обратном направлении. В результате этого формируется набор выходных сигналов, который является воздействием нейронной сети на выходные данные.

Наиболее распространенным методом распознавания объектов на изображении является метод распознавания на основе ключевых признаков, однако данный способ все же имеет некую погрешность. Поэтому для более эффективного распознавания также применяются операции детектирования и классификации. В задачи детектирования входит сбор характерных признаков, присущих объекту, к примеру, последовательность цветовых кодов. После детектирования, как правило, и применяют классификацию, а это уже использование самой нейронной сети и различных генетических алгоритмов, благодаря которым можно обозначить область искомого объекта на изображении.

Основной операцией в сверточных нейронных сетях является свертка, конечно же, можно обойтись и без нее, но в таком случае придется закреплять один нейрон ко всем пикселям на изображении, в таком случае количество весов, передаваемых другим слоям, будет колоссальным, это очень высокоёмкая вычислительная операция, а также занимающая продолжительное количество времени.

На рис. 3 представлена операция свертки. Осуществляется она за счет того, что ядро свертки проходит по всему изображению, затрагивая каждый пиксель изображения. Такая операция позволит сократить количество весов, а значит, напрямую ускорит вычислительные процессы [5].

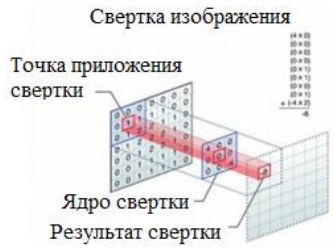


Рис. 3. Операция свертки изображения

Так как распознавание объектов на изображении строится на операции свертки, следует обратить внимание на два метода ее реализации, а именно padding (дополнение отступа) и strides (выбор шага). Опять же выбор метода зависит напрямую от того, что именно реализует пользователь в результате работы нейронной сети. При скольжении ядра свертки исходный массив обрезается по краям, например преобразуя матрицу 7×7 в 5×5, и из-за этого пиксели, которые находятся по краям, будут утеряны, так как само ядро свертки не может выйти за пределы края. Поэтому если задачей является получение выходного массива, равного входному, то операция дополнения пикселями в данном случае подойдет как нельзя кстати. Если же перед пользователем стоит цель уменьшить размеры выходного сигнала, то в данном случае уже стоит использовать stride [6]. Благодаря данному методу можно уменьшить размеры входной сетки вдвое.

Функции активации и потери – еще один важный сегмент в обучении нейронной сети. Функция активации отвечает за то, какие нейроны будут активированы и какая информация будет передаваться в следующие слои. Без использования функции активации использование глубоких нейронных сетей становится нерациональным и менее эффективным, ведь без нее сети утрачивают большую часть способности к обучению. Функция потерь расположена в самом центре нейронной сети. Благодаря ей происходит вычисление ошибок между полученными сетью и реальными ответами, задача любой нейронной сети – минимизация расхождений в предсказании [7].

Также одну из немаловажных ступеней в реализации нейронных сетей занимают оптимизаторы. Они играют важную роль в процессе обучения, помогая делать сети наиболее точные прогнозы в распознавании изображений. Оптимизаторы определяют самые рациональные параметры модели, такие как веса, смещение. Рассмотрим два наиболее известных оптимизатора: SGD (стохастический градиентный спуск) и Adam [8].

Оптимизатор Adam является одним их самых эффективных алгоритмов оптимизации. Данный оптимизатор прост в реализации, имеет высокую вычислительную эффективность и подходит для большинства задач, реализуемых в нейронных сетях. Рассмотрим данные оптимизаторы на примере обучения нейронной сети по распознаванию дорожных знаков на изображении. Для сравнения работы этих двух оптимизаторов была обучена нейронная сеть на 20 эпохах. Результаты обучения представлены в табл. 1.

Таблица 1

Результаты обучения нейронной сети

Сравниваемый параметр	Adam	SGD
Время на 1 эпоху, с	180	165
Общее время обучения, с	3600	3300
Точность предсказания, %	98	94

Исходя из результатов, представленных в табл. 1, видно, что SGD показал наиболее быстрое время прокрутки каждой эпохи и как следствие общее время обучения по сравнению с оптимизатором Adam. Но, в свою очередь, Adam показал более точную вероятность предсказания, а это все-таки наиболее важный критерий в обучении нейронной сети, поэтому для решения задач, требующих наивысшей

точности, стоит отдать предпочтение оптимизатору Adam. Также огромную роль в машинном обучении играет количество эпох. В обучении нет какого-либо оптимального количества эпох, ведь их количество напрямую связано с многообразием в данных. Так, датасет может состоять из изображений черных машин, либо же это куда более обширный набор данных, содержащий в себе различные классы, изображения, где уже малым количеством эпох большой точности не добиться [9]. Для того чтобы оценить влияние количества эпох на точность предсказания, была обучена одна и та же модель на разном количестве эпох. На рис. 4 представлена работа обученной нейронной сети на 20 эпохах.

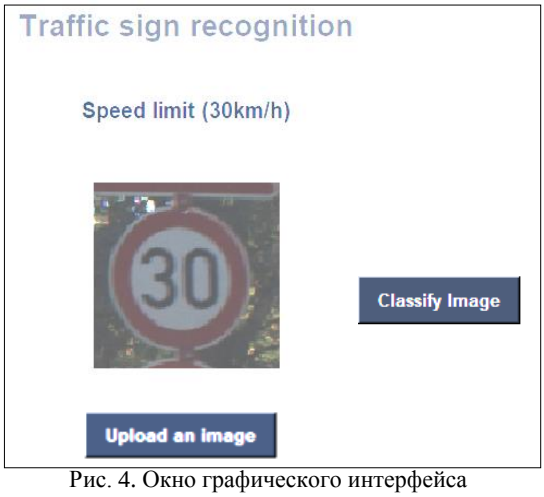


Рис. 4. Окно графического интерфейса
для определения класса выводимого изображения

Данной моделью было распознано порядка 25 изображений, каждое из которых было безошибочно определено нейронной сетью. Впоследствии эта же модель была обучена на 1, 5, 10 эпохах с использованием одного и того же датасета.

Рис. 5. Окно графического интерфейса
на примере знака «стоп»

Модель, обученная на 10 эпохах, также показала хороший результат в определении класса по изображению. На рис. 5 представлен один из примеров определения класса модели с 10 эпохами. Было рассмотрено также 25 различных изображений, из которых нейронная сеть не допустила ни одной ошибки. Однако модель, обученная при одной и пяти эпохах, все-таки допускала ошибки в определении класса знака на изображении.

Рис. 6. Ошибочное определение класса объекта
на изображении

В целом в силу того, что все изображения в датасете были уже обработаны и обрезаны для более легкой и точной работы нейронной сети, т.е. на изображении не присутствовали никакие лишние объекты и помехи, нейронной сети было нетрудно определить знак на загружаемом изображении. В табл. 2 приведена зависимость количества распознанных изображений от количества эпох.

Таблица 2

Зависимость количества распознанных изображений от количества эпох

Кол-во эпох	Кол-во изображ.	Верно распознанных изображ.	Неверно распознанных изображ.
20	25	25	0
10	25	25	0
5	25	24	1
1	25	20	5

Как можно видеть из результатов, приведенных в табл. 2, для данного датасета нет необходимости в большом количестве эпох. Тем не менее даже на такой идеальной для модели тестовой выборке можно наглядно убедиться во влиянии количества эпох на результат распознавания.

Литература

1. Нейронные сети как инструмент интеллектуального моделирования [Электронный ресурс]. – Режим доступа: https://studbooks.net/2252431/informatika/pereobuchenie_neyronnyh_setey, свободный (дата обращения: 09.09.2022).

2. Искусственные нейронные сети и приложения: учеб. пособие / Ф.М. Гафаров, А.Ф. Галимянов. – Казань: Изд-во Казан. ун-та, 2018. – 121 с.

3. Нейронные сети для начинающих [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/312450/, свободный (дата обращения: 09.09.2022).

4. Глубокое обучение и искусственные нейронные сети [Электронный ресурс]. – Режим доступа: https://tirendazakademi.medium.com/derin-öğrenme-ve-yapay-

sĭnĭr-ağlari a80b9bc716b3, свободный (дата обращения: 09.09.2022).

5. Самое главное о нейронных сетях [Электронный ресурс]. – Режим доступа: https://www.pvsm.ru/programmirovanie/169045, свободный (дата обращения: 09.09.2022).

6. Как работает сверточная нейронная сеть: архитектура, примеры, особенности [Электронный ресурс]. – Режим доступа: https://neurohive.io/ru/osnovy-data-science/glubokaya-svertochnaja-nejronnaja-set/, свободный (дата обращения: 10.09.2022).

7. Как работает нейронная сеть: алгоритмы, обучение, функции активации и потери [Электронный ресурс]. – Режим доступа: https://neurohive.io/ru/osnovy-data-science/osnovy-nejronnyh-setej-algoritmy-obuchenie-funkcii-aktivacii-i-poteri/ свободный (дата обращения: 10.09.2022).

8. Реализуем и сравниваем оптимизаторы моделей в глубоком обучении [Электронный ресурс]. – Режим доступа: https://habr.com/ru/company/skillfactory/blog/525214/ свободный (дата обращения: 08.09.2022).

9. Эпоха, батч, итерация – в чем различия? [Электронный ресурс]. – Режим доступа: https://neurohive.io/ru/osnovy-data-science/jepoha-razmer-batcha-iteracija/ свободный (дата обращения: 08.09.2022).

Сальников Сергей Алексеевич
Студент каф. ТУ ТУСУРа
Эл. почта: salnikovs802gmail.com

УДК 004.932

С.В. Сай, В.С. Никонов

Нейросетевой классификатор блоков изображения по признакам детальности

Представлены особенности разработки нейросетевого анализатора детальности изображения. Нейросетевой анализатор представляет собой бинарный классификатор, построенный на основе алгоритма поиска и идентификации мелких структур изображения с разным уровнем детальности (FDL). Разработанная нейросеть обладает небольшим количеством нейронов, требует малого количества данных и может быть применена во встраиваемых устройствах. Нейросетевой классификатор рассматривается в контексте применения нейросетей для оптимизации вычислений.

Ключевые слова: нейросеть, анализ изображения, машинное обучение, оптимизация.

Классификация изображений по их признакам используется в различных задачах поиска, распознавания, идентификации и сегментации. К информативным признакам изображения относятся его яркость, цветность, контраст, детальность, контурные и структурные признаки. Информативная часть изображения состоит из деталей и структур, изменения контраста которых различаются глазом. Остальная часть изображения считается фоновой.

Одним из информационных признаков изображения является его детальность, которую можно оценить отношением площади информативной части к площади всего изображения. Детальность изображения можно оценивать разными математическими способами [1]. Наиболее популярным считается метод вычисления количества мелких структур (блоков) изображения, в которых среднеквадратичное отклонение по сигналу яркости превышает пороговое значение.

В работе [2] предложено развитие этого метода на основе вычислений цветового контраста K между соседними пикселями блока с размерами 3×3 элемента в нормированном цветовом пространстве N-CIELAB. Критерий идентификации блока следующий: 1) если количество цветовых переходов с контрастом $K>1$ превышает единицу, то такой блок содержит мелкие структуры, которые различаются глазом; 2) если первое условие не выполняется, то такой блок идентифицируется как фоновый блок. Отношение количества идентифицированных блоков к общему числу блоков будет определять коэффициент детальности (FDL) изображения. К особенностям алгоритма относится его вычислительная сложность за счет математических преобразований сигналов из RGB-пространства в систему N-CIELAB.

В данной работе представлен нейросетевой классификатор блоков изображения на основе алгоритма вычисления FDL. Разработанная нейросеть предназначена для оптимизации объемных вычислений. Основные достоинства нейросети – малый размер данных, необходимых для обучения, небольшое количество нейронов в скрытом слое и векторизация вычислений.

Материалы и методы

Рассмотрим более подробно вычисление контраста в системе N-CIELAB. Преобразование из цветового пространства *RGB* в CIELAB выполняется через пространство *XYZ*. Формулы этих преобразований [3] (для стандартного осветителя D65) представлены ниже:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{pmatrix} 0,4124 & 0,3576 & 0,1805 \\ 0,2126 & 0,7152 & 0,0722 \\ 0,0193 & 0,1192 & 0,9505 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (1)$$

$$\begin{aligned} L^* &= 116 f(Y/Y_n) - 16, \\ a^* &= 500[f(X/X_n) - f(Y/Y_n), \\ b^* &= 200[f(Y/Y_n) - f(Y/Y_n)], \end{aligned} \quad (2)$$

$$f(t) = \begin{cases} t^{1/3}, & \text{если } t > 0,008856, \\ 7,787t + 16/116, & \text{если } t \le 0,00856. \end{cases} \quad (3)$$

Значения X_n, Y_n и Z_n являются координатами эталонного белого цвета. Из приведенных уравнений следует, что преобразование RGB в $L^*a^*b^*$ не является линейным, что, в свою очередь, приводит к повышению вычислительной сложности преобразования.

Контраст между соседними (j и i) пикселями блока вычисляется по следующему выражению:

$$K_{j,i} = \sqrt{\left(\frac{L_j^* - L_i^*}{\delta_L}\right)^2 + \left(\frac{a_j^* - a_i^*}{\delta_a}\right)^2 + \left(\frac{b_j^* - b_i^*}{\delta_b}\right)^2}, \quad (4)$$

где $\delta_L = 6$, $\delta_a = 40$ и $\delta_b = 55$ – значения весовых коэффициентов для мелких деталей с размером в один пиксель по светлоте и по цветности.

Для идентификации блока с размером 3×3 пикселя потребуется вычислить 12 значений контраста (4) и на каждом шаге проверять выполнение условия

$$K_{j,i} > 1. \quad (5)$$

После идентификации всех блоков строится битовая карта изображения, по которой можно определить пространственное положение и количество детальных (N_d) и фоновых (N_f) блоков и вычислить коэффициент детальности:

$$\text{FDL} = N_d / (N_d + N_f). \quad (6)$$

Анализ выражений (1)–(5) показывает, что для идентификации одного блока потребуется не менее

550 вычислительных операций, которые будут выполняться последовательно в процессорном конвейере. Для повышения производительности вычислений было принято решение реализовать алгоритм идентификации блоков с помощью нейронной сети.

Для разработки нейросети использовался язык программирования Python совместно с библиотеками для машинного обучения и обработки данных: Tensorflow, NumPy, Pandas, Keras. Для визуализации результатов использовалась библиотека Matplotlib.

В качестве архитектуры был выбран вариант полносвязной нейросети из 27 входных нейронов, одним скрытым слоем (256 нейронов) и одним выходным нейроном. Данный выбор был обусловлен малым количеством параметров для обучения и вычисления значений нейросети. Для устранения проблемы «мертвых нейронов» [4] был использован метод Leaky ReLU [5] со значением утечки 0,5 в качестве функции активации между нейронами.

В качестве набора данных, на котором проводилось обучение нейросети, был выбран архив тестовых изображений [6] университета Южной Калифорнии (раздел «Miscellaneous») из 45 изображений.

Результаты исследований

Нейросеть была обучена на 45 образцах тестовых изображениях с различной детальностью. Для подготовки входного набора данных каждое изображение разбивалось на блоки с размерами 3×3 пикселя, т.е. на 27 значений RGB-координат. В процессе обучения для каждого блока по формулам (1)–(5) вычислялся бинарный параметр идентификации, который определял целевое значение выхода сети.

Итоговая точность нейросети после обучения на протяжении 50 эпох составила 93,35%. Показатель Area Under Curve (AUC), полученный при проведении ROC-анализа, равен 0,98. История обучения и кривая ROC-анализа представлены на рис. 1 и 2 соответственно.

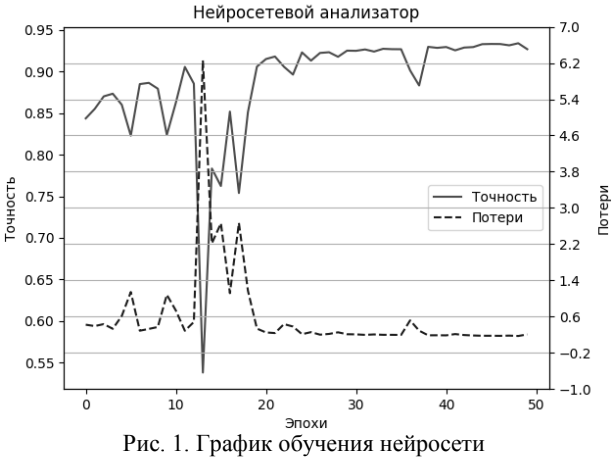


Рис. 1. График обучения нейросети

Полученные результаты свидетельствуют об успешном обучении бинарного классификатора.

На рис. 3 показаны сегментированные участки с мелкими деталями тестового изображения «Лена» после выполнения алгоритма оценки FDL по формулам (1)–(5). На рис. 4 представлено аналогичное изображение, на котором блоки с мелкими деталями идентифицированы с помощью разработанной нейросети.

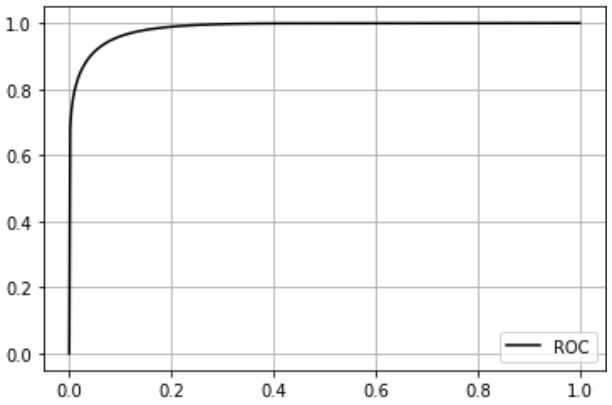


Рис. 2. Кривая ROC нейросети

Рис. 3. Результат алгоритма вычисления FDL

Рис. 4. Результат работы нейросетевого классификатора

Можно отметить, что визуальные отличия работы алгоритма FDL и нейросетевого классификатора малозаметны. При этом оценка коэффициента де-

тальности (6) по алгоритму равна FDL=0,197 и по результатам нейросети FDL=0,185.

В результате анализа более 50 тестовых изображений с различной детальностью получено, что средний разброс оценок коэффициента детальности по алгоритму FDL и с помощью нейроклассификатора не превышает 5%.

Заключение

Представленный способ реализации нейросети может стать эффективным средством вычислений для больших наборов данных, так как содержит встроенные средства векторизации и может легко быть перенесен для параллельного исполнения в распределенных вычислительных средах.

Также представленная нейросеть может использоваться во встраиваемых системах на кристалле благодаря небольшому количеству нейронов в скрытом слое и такими инструментами, как Tensorflow Lite [7] и keras2c [8]. В случае необходимости можно легко обучить новые варианты нейросетей с меньшим и большим количеством нейронов. Для более быстрого, но менее точного вычисления или более точного и медленного – в зависимости от цели.

В рамках дальнейших исследований предполагается внедрение нейроклассификатора в видеосистему на кристалле на базе FPGA [9] и технологий CUDA [10] с целью реализации эффективных алгоритмов сжатия и передачи изображений с высоким качеством.

Исследование выполнено при финансовой поддержке Российского научного фонда в рамках научного проекта № 22-21-00394 «Развитие нейросетевых методов повышения качества передачи цифровых изображений в интеллектуальных видеосистемах».

Литература

1. Старовойтов В.В. Цифровые изображения: от получения до обработки / В.В. Старовойтов, Ю.И. Голуб. – Минск: ОИПИ НАН Беларуси, 2014. – 202 с.

2. Сай С.В. Метрика искажений мелких структур компрессированных изображений // Компьютерная оптика. – 2018. – № 5 (42). – С. 829–837.

3. Fairchild M.D. Color Appearance Models. – Wiley, 2005. – 416 p.

4. Dying ReLU and Initialization: Theory and Numerical Examples / L. Lu, Y. Shin, Y. Su, G.E. Karniadakis // Communications in Computational Physics. – 2020. – № 5 (28). – PP. 1671–1706.

5. Maas A.L. Rectifier nonlinearities improve neural network acoustic models / A.L Maas, A.Y. Hannun, A.Y. Ng. – Workshop on Deep Learning for Audio, Speech and Language Processing. – ICML, 2013.

6. SIPI Image Database [Электронный ресурс]. – Режим доступа: https://sipi.usc.edu/database/, свободный (дата обращения: 13.09.2022).

7. Развертывание моделей машинного обучения на мобильных и периферийных устройствах TensorFlow Lite [Электронный ресурс]. – Режим доступа: https://www.tensorflow.org/lite?hl=ru, свободный (дата обращения: 13.09.2022).

8. Conlin R. keras2c [Электронный ресурс]. – Режим доступа: https://github.com/f0uriest/keras2c, свободный (дата обращения: 13.09.2022).

9. Шипицин С.П. Развитие аппаратно-ориентированных нейронных сетей на FPGA и ASIC / С.П. Шипицин, М.И. Ямаев // Вестник Перм. нац. исслед. политехн. ун-та. – 2019. – № 3. – С. 177–192.

10. Парубец В.В. Применение технологии CUDA для ускорения вычислений в нейронных сетях / В.В. Парубец, О.Г. Берестнева, Д.В. Девятых // Изв. Том. политехн. ун-та. – 2012. – Т. 320, № 5 (320). – С. 121–125.

Сай Сергей Владимирович
Д-р техн. наук, профессор, зав. каф. вычислительной техники (ВТ) Тихоокеанского государственного университета (ТОГУ)
Эл. почта: sai1111@rambler.ru

Никонов Василий Сергеевич
Аспирант каф. ВТ ТОГУ
Эл. почта: 2021104211@pnu.edu.ru

УДК 004.932.2

А.Н. Шлыкович, С.А. Забуга

Сравнение классических методов интерполяции изображения с моделью искусственной нейронной сети SRGAN

Проводится сравнение классических методов интерполяции изображений с искусственной нейронной сетью, которая используется для повышения пространственного разрешения изображений. Критерием оценки является эффективность работы методов и алгоритма.

Ключевые слова: интерполяция, увеличение изображений, нейронная сеть, пространственное разрешение изображений, SRGAN.

Интерполяция – под этим определением понимается то, что происходит нахождение промежуточных средних значений по уже известным значениям.

Другими словами, в нашем случае – для увеличения разрешения изображений и нахождения новых пикселей за счет уже существующих пикселей.

Целью данной работы является сравнение различных методов и алгоритмов увеличения изображения с использованием искусственной нейронной сети.

Актуальность данного вопроса заключается в том, что потребность в использовании компьютерного зрения увеличивается, и для улучшения работы алгоритмов компьютерного зрения нужны изображения лучшего качества. Также и в медицине есть потребность в увеличении изображений, так как получаемое изображение может иметь меньший размер, чем разрешение монитора, на котором работает специалист в данной сфере.

Методы интерполяции изображения

В данной работе исследованы следующие методы интерполяции: бикубическая интерполяция и метод Ланцоша.

Бикубическая интерполяция использует массив из 16 соседних точек, которые расположены в вершинах плоскости.

По сравнению с рассмотренными выше методами интерполяции данный метод показывает лучшее увеличение контрастных изображений. Изображение более гладкое по сравнению с методом ближайшего соседа или билинейной интерполяции. Также алгоритм является оптимальным по соотношению затраченного времени на обработку изображения и полученного результата качества изображения. Поэтому бикубическую интерполяцию используют в обработке большинство графических редакторов, таких как Adobe Photoshop, Corel Photopaint [1].

Метод Ланцоша используется в таких графических редакторах, как GIMP, Krita, для интерполяции изображений [2]. В основе метода Ланцоша лежит функция sinc (кардинальный синус).

Этот метод позволяет получить высокую четкость изображения, а также сохранить резкие линии. Однако в результате его работы зачастую на изображении могут появиться такие артефакты, как эффект Гиббса.

Искусственная нейронная сеть SRGAN (Super-Resolution Generative Adversarial Network)

Работа данной модели производится по следующему алгоритму:

1. Необходимо подготовить изображения – эталоны.

2. Генерируется шум, благодаря которому генератор получает данные.

3. После этого происходит формирование набора данных для последующего обучения дискриминатора. Данные делятся на два типа (эталонные и полученные генератором), каждому из них присваивается метка для эталонных – это 1, а для изображений, полученных генератором, – 0.

4. После обучения дискриминатора следует обучение генератора, при этом обучение дискриминатора приостанавливается. На вход приходит шум, а в конце ожидается получение отметок 1. Другими словами, распознавание дискриминатором данных как реальных. При этом, когда происходит обучение, не используются реальные изображения, а только отметки дискриминатора [3].

Методы сравнения изображения

Для сравнения рассматриваемых методов были использованы объективные метрики оценки качества изображений.

Нахождение метрик качества MSE, PSNR, SSIM, ТВЛ производилось по следующим формулам:

$$\mathrm{PSNR}(x,y) = 10\lg\left(L^2/\mathrm{MSE}(x,y)\right), \qquad (1)$$

где MSE – среднеквадратичное отклонение.

$$\mathrm{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left|I(i,j)-K(i,j)\right|^2, \qquad (2)$$

где m, n – размер изображений в пикселях, I, K – матрицы яркостей исходного и искаженного изображений соответственно.

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \qquad (3)$$

где y, x – исходное и обработанное изображение; μ_y – среднее y; μ_x – среднее x; $\sigma^2{}_y$ – дисперсия y; $\sigma^2{}_x$ – дисперсия s; σ_{yx} – ковариация y и x; c_1 – переменная, равная 6,5025; c_2 – переменная, равная 58,5225 [4].

Нахождение контрастно-частотной характеристики (КЧХ) разрешающей способности изображения производится по формуле (4):

$$\text{ТВЛ} = \left(\frac{\text{СРР}}{0,5}\right) \cdot \text{Предельное разрешение в ТВЛ}. \quad (4)$$

Предельное разрешение в ТВЛ – это разрешение, при котором каждый пиксель приходится на одну линию.

Экспериментальные исследования

Программная реализация выполнялась на языке программирования Python версии 3.10 с использование следующих библиотек: OpenCV, NumPy, PyTorch. Среда разработки для написания кода была выбрана PyCharm, так как является удобной и бесплатной.

OpenCV – данная библиотека в свободном доступе и код открыт для всех желающих, содержит различные наборы типов данных, функций и классы, используемые для обработки изображений с помощью алгоритмов компьютерного зрения [5].

NumPy – данная библиотека добавляет поддержку массивов и матриц, а также математические функции для операций с этими массивами [6].

PyTorch – библиотека машинного обучения для языка Python для желающих с открытым исходным кодом, созданная на базе Torch [7].

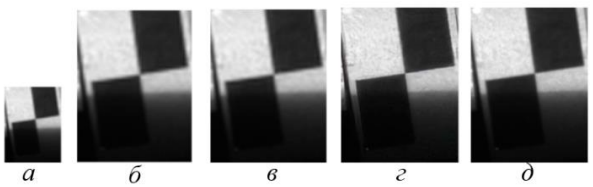
Выбраны классические методы, такие как бикубическая интерполяция и метод Ланцоша, для сравнения с моделью искусственной нейронной сети SRGAN.

В качестве эталонного было взято изображение разрешением 1280×720 пикселей в формате .jpg. Для сравнения изображений метриками качества нужно подготовить изображение, уменьшив его разрешение в четыре раза до размеров 640×360 пикселей. После данной операции можно будет сравнить эффективность работы классических методов интерполяции с моделью искусственной нейронной сети SRGAN.

Для обучение нейронной сети было взято 5000 изображений размером 96×96 писклей, из них на тестирование было взято 1 000, остальные пошли на обучение, время обучения составило 13 ч.

Обучение нейронных сетей проводилось на компьютере со следующей конфигурацией: процессор AMD Ryzen 7 3700×3.6 ГГц, видеокарта NVIDIA GeForce RTX 2060 6GB, оперативная память 16 Гб.

На рис. 1 приведены фрагменты изображения полученных изображений с помощью бикубической интерполяции, метода Ланцоша и модели SRGAN.



а *б* *в* *г* *д*

Рис. 1. Полученные результаты (слева направо: *а* – уменьшенное изображение; *б* – бикубическая интерполяция; *в* – метод Ланцоша; *г* – SRGAN, *д* – эталонное изображение)

Из табл. 1 видно, что по значению MSE и PSNR преобладает метод Ланцоша, а по значению SSIM бикубическая интерполяция.

Данные метрики не подходят, так как модель искусственного интеллекта изменяет изображения, вносит свои изменения от исходного изображения, а любое отличие приводит к повышению MSE и снижению PSNR, SSIM.

Таблица 1

Полученные результаты MSE, PSNR, SSIM

Алгоритм	MSE	PSNR	SSIM
Бикубическая интерполяция	15,9710	36,0974	0,9401
Метод Ланцоша	14,9112	36,3956	0,9413
SRGAN	22,5931	34,5910	0,9278

Поэтому было принято решение получить значения ССР и ТВЛ в программном обеспечении Imatest.

В данное приложение загружается чёрно-белое изображение, выбирается область, где четко происходит переход от белого к черному. На рис. 2–5 и в табл. 2 приведены полученные результаты ССР и ТВЛ для двух уровней контраста 0,5 и 0,3.

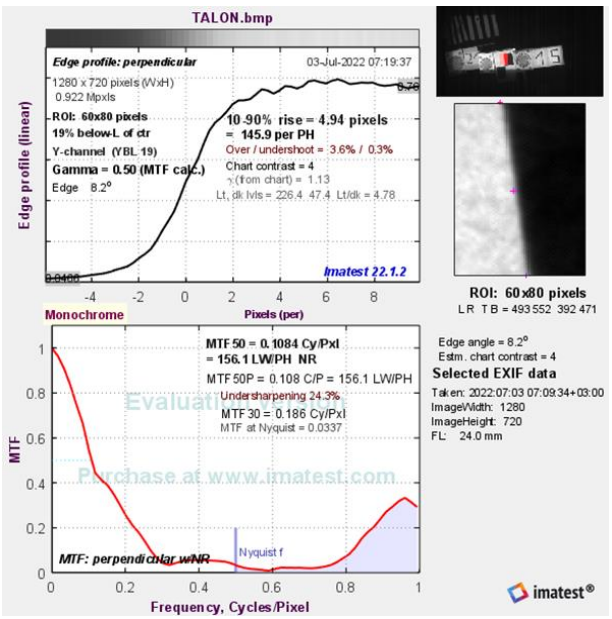


Рис. 2. Эталонное изображение

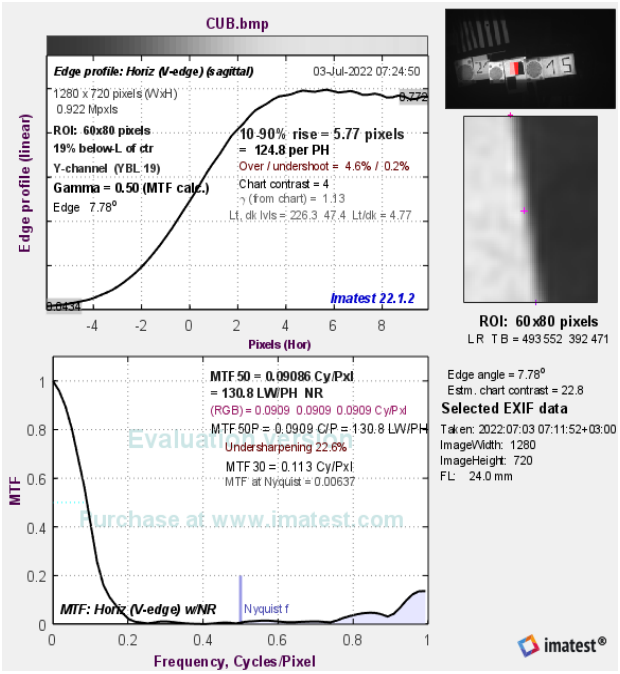


Рис. 3. Бикубическая интерполяция

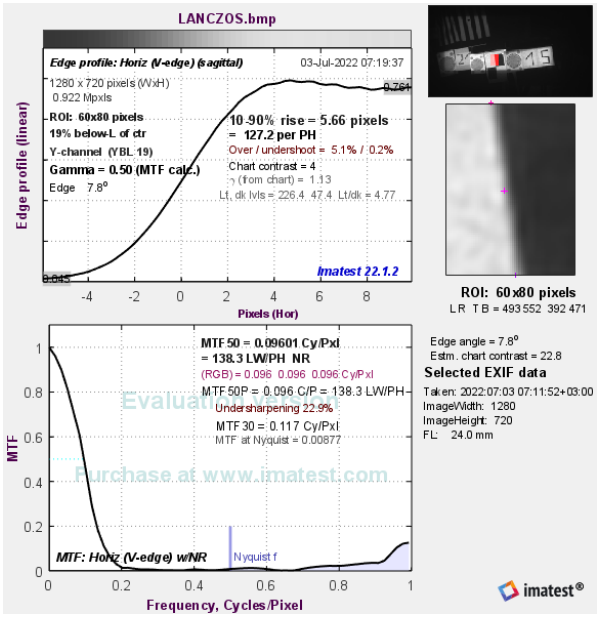


Рис. 4. Метод Ланцоша

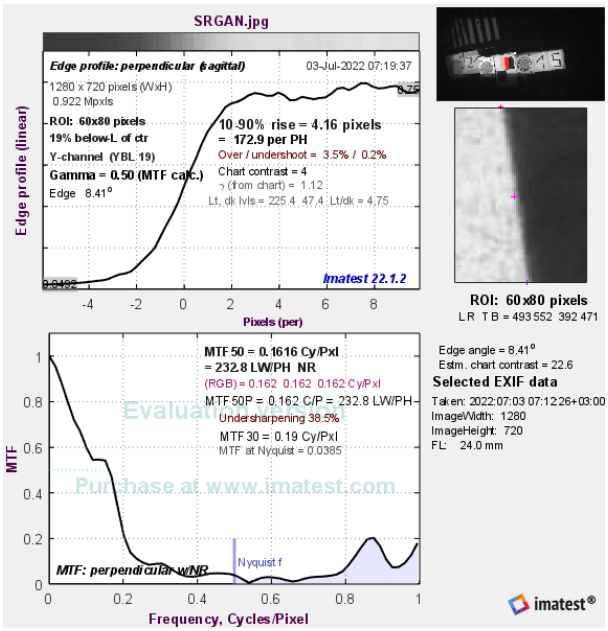


Рис. 5. Модель SRGAN

Таблица 2

Полученные результаты ТВЛ

Метод	CPP (MTF50)	ТВЛ	CPP (MTF30)	ТВЛ
Эталонное изображение	0,1084	156,1	0,186	267,84
Бикубическая интерполяция	0,09086	130,8	0,113	162,72
Метод Ланцоша	0,09601	138,3	0,117	168,48
SRGAN	0,1616	232,8	0,19	273,6

Из полученных результатов видно, что модель SRGAN показывает себя хорошо как визуально, так и по значениям CPP и ТВЛ.

Из рисунков с методами интерполяции видно, при использовании каждого метода присутствуют свои артефакты, такие как алиасинг, размытие, эффект Гиббса. При использовании методов Ланцоша и бикубической интерполяции проявляется эффект Гиббса, ореолы в местах яркостных переходов.

Заключение

При визуальной оценке качества полученных изображений искусственные нейронные сети делают изображение более резким и детальным по сравнению с другими методами.

При сравнении результатов с помощью объективных метрик результат искусственной нейронной сети хуже, связано это с тем, что нейронная сеть дорисовывает детали, которые отличаются от тех, что присутствуют в эталонном изображении, а любые отличия приводят к повышению MSE и снижению PSNR и SSIM.

По результатам КЧХ лучше всего себя показала модель SRGAN, среди классических методов хорошо себя показали бикубическая интерполяция и метод Ланцоша.

Литература

1. Метод бикубической интерполяции. – Robocraft, 2021 [Электронный ресурс]. – Режим доступа: https://robocraft.ru/computervision/3956 (дата обращения: 09.03.2022).

2. Метод интерполяции Лонцоша. – Gamedev, 2021 [Электронный ресурс]. – Режим доступа: http://www.gamedev.ru/code/forum/?id=202955, (дата обращения: 08.03.2022).

3. Rueckert D., Wang Z. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network // IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – 2016. – No. 3 (58). – PP. 448–456.

4. Классификация изображений с помощью SSIM. – Machinelearningmastery, 2021 [Электронный ресурс]. – Режим доступа: https://www.machinelearningmastery.ru/image-classification-using-ssim-34e549ec6e12/, (дата обращения: 08.03.2022).

5. Суков А.В. Компьютерное зрение // Библиотека OpenCV и ее применение в прикладных задачах робототехники. – Донецк: Донецкий нац. техн. ун-т, 2016 (дата обращения: 05.03.2022).

6. Хабр, 2021 // Библиотека NumPy Python [Электронный ресурс]. – Режим доступа: https://habr.com/ru/post/352678/, свободный (дата обращения: 06.03.2022).

7. PyTorch [Электронный ресурс]. – URL: https://habrahabr.ru/company/ods/blog/324898/ (дата обращения: 22.06.2022).

Шлыкович Андрей Николаевич
Магистрант каф. ТУ ТУСУРа
Эл. почта: iksis23@bk.ru

Забуга Сергей Александрович
Магистрант каф. ТУ ТУСУРа
Эл. почта: sergeizabuga@gmail.com

УДК 004.855.5

Г.Е. Эдель, М.Е. Сукотнова, В.В. Капустин

Трансферное обучение нейронной сети для повышения точности распознавания детей с использованием NVIDIA Jetson Nano

Проанализировано трансферное обучение модели для автоматического обнаружения детей. Главной целью работы являлось повышение точности обнаружения детей по сравнению со взятой за основу моделью MobileNet V1, с которой в дальнейшем и будет производиться сравнительный анализ. Для трансферного обучения был использован собственный набор данных. Само обучение было произведено в бесплатном облачном сервисе Google Colab. Обе модели были реализованы и протестированы на одноплатном микрокомпьютере NVIDIA Jetson Nano.

Ключевые слова: обнаружение детей, NVIDIA Jetson Nano, MobileNet V1, распознавание пешеходов, система помощи водителю, беспилотные автомобили.

Проблема автоматического обнаружения людей не нова и уже давно является краеугольным камнем в области компьютерного зрения. Людям свойственно большое разнообразие образов, например: пол, одежда, физическая форма, возраст, раса. Все это и многое другое сильно усложняет задачу обнаружения человека. Хоть в последние годы и удалось достигнуть огромного прогресса в области компьютерного зрения, системы обнаружения людей попрежнему остаются далеко не идеальными и нуждаются в доработке. Поэтому ежегодно интерес к данной области лишь возрастает, как и бюджеты, выделяемые на разработку и исследования. Это не удивительно, ведь создание систем, способных автоматически распознавать людей в реальном времени, позволяет решать множество задач как в беспилотном автотранспорте, так и в интеллектуальном видеонаблюдении.

Современные технологии уже дают возможность обрабатывать видеопоток и выполнять распознавание объектов в реальном времени. Модель MobileNet V1 является отличным вариантом для использования на одноплатном микрокомпьютере NVIDIA Jetson Nano, при ее запуске она способна работать в реальном времени, выдавая порядка 27–28 кадров в секунду (FPS) [1–3]. Однако точность распознавания порой оставляет желать лучшего. Особые проблемы возникают при обнаружении детей, именно их зачастую не получается распознать. Так, MobileNet V1 в некоторых тестах принимала ребенка за пожарный гидрант, как показано на рис. 1.

Рис. 1. Ложное обнаружение пожарного гидранта вместо ребенка

И трудности при обнаружении детей – это не проблема какой-то отдельно взятой модели. Об этом свидетельствуют периодически всплывающие новости о проваленных тестах и сбитых манекенах детей беспилотными автомобилями. Так уж получается, что именно самые незащищённые участники дорожного движения, а именно дети, являются наиболее уязвимыми и имеют наибольший риск быть нераспознанными. Каждое дорожно-транспортное происшествие, особенно с трагическими последствиями, является личной трагедией, а также создает большую проблему для общества в целом. Поэтому так важно уже сейчас привлекать все большее внимание и ресурсы для повышения безопасности на дорогах. В особенности это касается безопасности детей, чье попадание в дорожно-транспортное происшествие с моральной точки зрения неприемлемо [2–4].

С помощью собранного набора данных и трансферного обучения была предпринята попытка повышения эффективности автоматического распознавания детей с использованием NVIDIA Jetson Nano.

Набор разработчика NVIDIA Jetson Nano

В работе использовалась версия одноплатного микрокомпьютера NVIDIA Jetson Nano B01 (рис. 2).

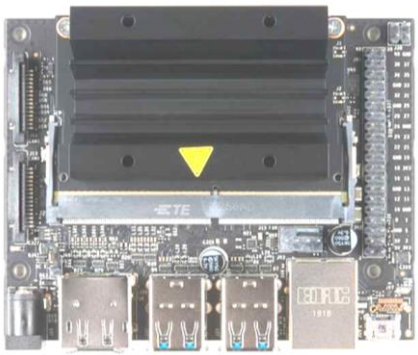


Рис. 2. Одноплатный микрокомпьютер NVIDIA Jetson Nano

Микрокомпьютер способен запускать несколько нейронных сетей параллельно, а также развертывать приложения компьютерного зрения и глубокого обучения, такие как классификация изображений, обнаружение объектов и сегментация. Jetson Nano совме-

стим с самыми популярными фреймворками искусственного интеллекта: TensorFlow, PyTorch, Caffe, Keras, MXNet и др. Благодаря своей невысокой стоимости и тому, что Jetson Nano дает широкие возможности разработки, развертывания и исследования компьютерного зрения с искусственным интеллектом [4-6].

Реализация трансферного обучения модели MobileNet V1

Как уже было сказано, точность обнаружения детей с помощью модели MobileNet V1 оставляет желать лучшего. Основными причинами низкого процента точности или отсутствия обнаружения можно назвать малое количество, а возможно, полное отсутствие изображений детей в наборе данных и меньшие линейные размеры детей, чем у взрослых.

В открытом доступе нет как моделей, обученных для обнаружения детей, так и готовых наборов данных с изображениями детей для обучения нейронных сетей. Поэтому был собран собственный набор данных из 2 323 изображений. На каждом изображении присутствовали дети. Набор данных формировался из как можно более разнообразных изображений. На фото дети были расположены как в профиль, так и фронтально, под различным углом и на изменяющейся дистанции [5–7]. Далее на каждом изображении с помощью графического инструмента аннотации изображений LabelImg были промаркированы все люди и названы классом «Person», как показано на рис. 3.

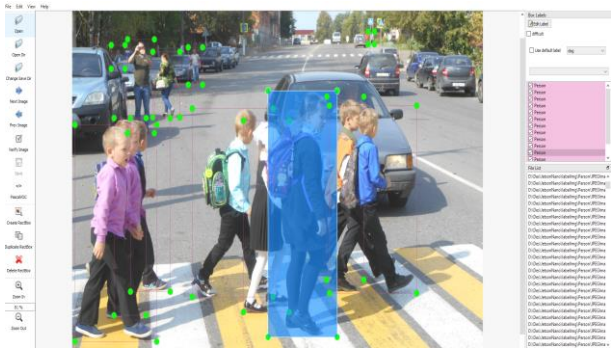


Рис. 3. Процесс маркировки изображений в LabelImg

После успешного создания набора данных было запущено трансферное обучение новой модели в облачном сервисе Google Colab с целью уменьшения временных затрат на обучение модели. Как показывает опыт, обучение модели на самом микрокомпьютере NVIDIA Jetson Nano весьма проблематично. При трансферном обучении разработчиками рекомендовано производить не менее 100 эпох обучения для достижения приемлемого результата точности распознавания. Именно такое количество эпох было запущено при трансферном обучении собственной модели. Это занимает очень много времени, а порой компьютер оказывается перегружен, из-за этого процесс переобучения может быть прерван. В результате трансферного обучения была получена переобученная на собственном наборе данных модель MobileNet TL [6–8].

Результаты тестирования моделей и их сравнительный анализ

Для исследования эффективности были отсняты видеоролики с находящимися в кадре детьми, воссоздающие различные ситуации на дороге. Съемка происходила на камеру мобильного телефона Xiaomi Redmi 8Pro при качестве 720p и 30FPS. Далее с помощью оригинальной MobileNet V1 и моделью, полученной в результате трансферного обучения MobileNet TL, был запущен процесс распознавания. При обнаружении человека он выделяется рамкой, а над ним появляется процент точности распознавания. После запуска модели MobileNet TL она продемонстрировала отличное быстродействие, выдавая около 37 FPS. Такой показатель быстродействия выше чем у оригинальной модели MobileNet V1, которая обычно выдает около 27 FPS. Первое, что было протестировано, как модели распознают детей на расстоянии 5 м при фронтальной и профильной съемке (рис. 4).

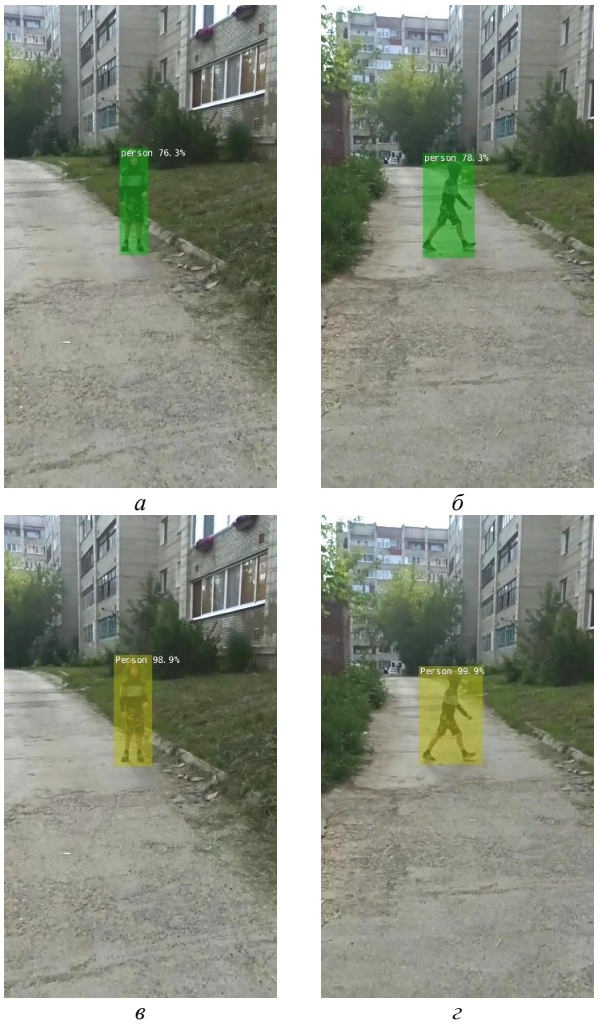
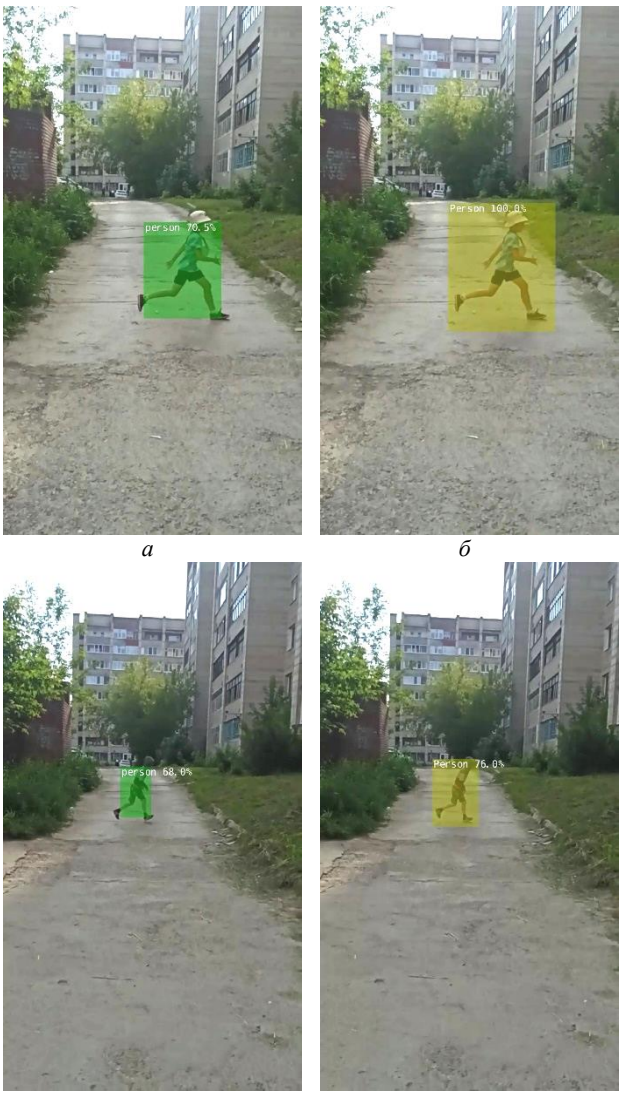


Рис. 4. Обнаружение ребенка при фронтальной и профильной съемке моделью MobileNet V1(*а*, *б*) и MobileNet TL (*в*, *г*)

Как видно из рис. 4, обе модели справились с обнаружением ребенка, однако процент точности распознавания у модели, полученной в результате

трансферного обучения, выше и составляет 98,9–99,9%, это более чем на 20% выше, чем у MobileNet V1 с ее 76,3–78,3%. Как известно, об этом свидетельствует и статистика, огромное количество дорожно-транспортных происшествий с наездом на пешеходов случаются при перебегании дороги в неположенном месте. Дети очень часто не выполняют правила дорожного движения из-за их незнания или пренебрежения. Благодаря этому существует даже знак «Осторожно, дети!». Взяв это во внимание следующий тест проводился на видеороликах с перебегающим дорогу ребенком на расстоянии 5 и 10 м (рис. 5).



а *б*

в *г*

Рис. 5. Обнаружение ребенка, перебегающего дорогу на дистанции 5 и 10 м моделью MobileNet V1 (*а*, *б*) и MobileNet TL (*в*, *г*)

По результатам тестов модель MobileNet TL вновь показала лучшие результаты и даже распознала ребенка с вероятностью 100% на расстоянии 5 м в отличие от MobileNet V1, которая выдала на том же кадре 70,5%. Как видно из рисунка, с увеличением дистанции точность распознавания обеих моделей снижается, однако модель, полученная в результате трансферного обучения, по-прежнему остается более точной и выдает 76 против 68%. Ниже приведен покадровый график точности распознавания перебегающего ребенка на дистанции, постоянно увеличивающейся от 5 до 10 м (рис. 6).

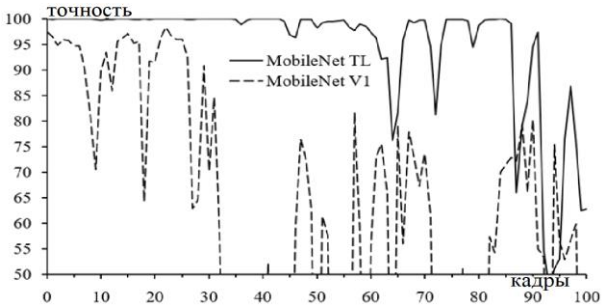


Рис. 6. Покадровый график точности обнаружения перебегающего ребенка на дистанции от 5 до 10 м

Точность распознавания ниже 50% было решено отбросить, так как такой процент можно считать случайным. По графику хорошо заметно, что MobileNet TL намного лучше распознает ребенка в приведенных 100 кадрах. График модели, полученный в результате трансферного обучения, имеет более стабильную характеристику и находится заметно выше. Так, средняя точность обнаружения для MobileNet TL равняется 95%, это на 41,6% больше, чем у MobileNet V1 с ее средней точностью 53,4%.

Также MobileNet TL была проверена на видеоролике, кадр из которого был приведен на рис. 1, где ребенок был обнаружен, но классифицирован как пожарный гидрант. Результат обнаружения переобученной моделью приведен на рис. 7.

Рис. 7. Обнаружение ребенка моделью MobileNet TL

Как видно из рис. 7, модель обнаружила ребенка, а вероятность точности распознавания равна 80,7%. Также был распознан и взрослый человек с более высокой точностью 99,8%.

Заключение

В результате выполнения данной работы было реализовано трансферное обучение нейронной сети с целью повышения точности распознавания детей. Был собран и промаркирован собственный набор данных, на основе которого и было произведено обучение. Воспользовавшись облачным сервисом

Google Colab, было сэкономлено время на обучение модели. Готовая модель получила название MobileNet TL и была протестирована совместно с MobileNet V1 на различных видеороликах с детьми. Созданная модель показала высокое быстродействие.

Результаты тестов показали, что поставленная задача повышения эффективности автоматического распознавания детей была достигнута. MobileNet TL показала более высокую точность обнаружения во всех тестах. В дальнейшем эти наработки могут быть использованы при создании системы помощи водителю с использованием NVIDIA Jetson.

Исследование выполнено за счет гранта Российского научного фонда № 21-79-10200 в ТУСУРе.

Литература

1. Боресков А.В. Параллельные вычисления на GPU. Архитектура и программная модель CUDA: учеб. пособие / А.В. Боресков, А.А. Харламов и др.; предисл. В.А. Садовничий. – М.: МГУ, 2012. – 336 с.

2. Мобильные сети: эффективные сверточные нейронные сети для приложений мобильного зрения [Электронный ресурс]. – Режим доступа: https://arxiv.org/abs/1704.04861 (дата обращения: 21.08.2021).

3. Rastegari M. Imagenet classification using binary convolutional neural networks / M. Rastegari, V. Ordonez, J. Redmon, A. Farhadi // European Conference on Computer Vision. – 2016. – PP. 525–542.

4. Комплект разработчика NVIDIA Jetson Nano [Электронный ресурс]. – Режим доступа: https://developer.nvidia.com/embedded/jetson-nano-developer-kit. (дата обращения: 21.08.2021).

5. MobileNetV2: Inverted Residuals and Linear Bottlenecks / M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen // IEEE Conference on Computer Vision and Pattern Recognition. – USA, Salt Lake City, 2018. – PP. 510–520.

6. Rethinking the Inception Architecture for Computer Vision / C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna // Computer Vision and Pattern Recognition. – USA, Las Vegas, 2016. – PP. 31–40.

7. Пастушков А.В. Методы и алгоритмы поиска объекта на видеопотоке / А.В. Пастушков, В.Т. Калайда // Сборник научных трудов SWorld. – 2013. – Т. 6, № 3. – С. 38–42.

8. Faster R-CNN: Towards real-time object detection with region proposal networks / S. Ren, K. He, R. Girshick, J. Sun // IEEE Transactions on Pattern Analysis and Machine Intelligence. – Canada, Toronto, 2015. – PP. 39–46.

Эдель Герман Евгеньевич
Студент каф. ТУ ТУСУРа
Эл. почта: german.edel99@gmail.com

Сукотнова Марина Евгеньевна
Студентка каф. ТУ ТУСУРа
Эл. почта: msukotnova@mail.ru

Капустин Вячеслав Валериевич
Канд. техн. наук, доцент каф. ТУ ТУСУРа
Эл. почта: peregnun@mail.ru

УДК 004.932.2

С.Е. Бояркин, А.В. Каменский

Адаптация метода Канни к работе с биомедицинскими изображениями

Представлено описание реализованного метода Канни для обработки биомедицинских изображений. Описан процесс его адаптации для работы с биомедицинскими изображениями путём подбора оптимальных коэффициентов их обработки, что требуется для улучшения качества изображений, обработанных методом Канни, и увеличения эффективности применения данного метода.
Ключевые слова: метод Канни, обработка изображений, коэффициент сглаживания.

Системы визуальной диагностики являются самыми востребованными в современной клинической практике, так как повышается общий уровень развития диагностических систем и всё больше ценится их малая инвазивность. Также имеют место быть и методы обработки видеоинформации посредством компьютерных технологий. Развитие методов компьютерной обработки графической информации, а именно медицинских изображений, положительно влияет на успешность проведения качественного анализа, постановки диагноза и дальнейшего лечения пациентов. Областью интереса данного исследования являются кости запястного сустава кисти руки.

Назначение и описание работы метода

На рис. 1 приведён алгоритм метода Канни для обработки биомедицинских изображений.

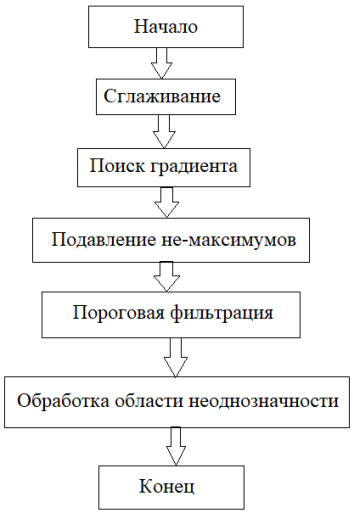


Рис. 1. Алгоритм метода Канни

Метод Канни предназначен для определения границ на изображении и состоит из следующих этапов [1].

1. Сглаживание исходного изображения.

На данном этапе происходит сглаживание изображения, что реализует размытие с целью уменьшения, подавления либо удаления нежелательных шумов.

2. Поиск градиентов.

Пиксели определяются как границы только там, где градиент изображения имеет максимальное значение.

В методе Канни могут использоваться различные операторы выделения границ, такие как Робертс, Собел, Превитт. В данной работе используется оператор Собеля, так как этот оператор не удаляет основные границы и передаёт большее количество деталей.

3. Подавление пикселей, не являющихся не-максимумами.

Только локальные максимумы отмечаются как границы.

На данном этапе определяются градиент и его направление. Далее те точки, в которых достигается локальный максимум градиента в направлении вектора этого градиента, определяются как пиксели границ. На следующем шаге все те точки, которые не определились как пиксели границ, подавляются путём проверки каждого пикселя на то, что принадлежит ли он локальному максимуму в своей окрестности в направлении градиента или нет.

4. Фильтрация путём установления порогов.

На данном этапе устанавливаются два порога: максимальный и минимальный. Если значения какого-либо пикселя превышает верхнюю границу, то этот пиксель приобретает максимальное значение и устанавливается как истинная граница. Если же значение пикселя меньше нижней границы, то данный пиксель подавляется.

5. Обработка области неоднозначности.

На данном этапе происходит выделение тех групп пикселей, которые попали в диапазон между порогов на предыдущем этапе. Эти группы пикселей определяются как границы либо подавляются в зависимости от того, соприкасаются ли данные пиксели с уже определёнными границами хотя бы по одному из восьми направлений. Конечные границы определяются путём подавления всех пикселей, которые не связаны с определенными границами [2, 3].

К преимуществам метода Канни относятся:

– менее подвержен шуму, чем классические операторы;

– использование двухпороговой фильтрации, чтобы исключить пропуск полезной части края или же истинной границы;

– с помощью коррекции некоторых параметров, например таких как, коэффициент и размер ядра сглаживания, данный метод можно адаптировать к любому конкретному изображению;

– хорошая локализация границ.

Все исследования проводились в ПО MATLAB с использованием Image Processing Toolbox [4, 5].

На рис. 2 и 3 представлены исходные изображения для обработки и результат работы метода Канни.

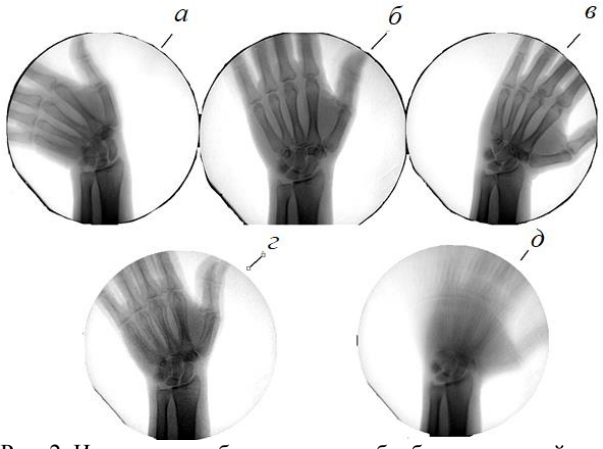


Рис. 2. Исходные изображения для обработки: *а* – крайнее левое положение; *б* – центральное положение; *в* – крайнее правое положение; *г* – промежуточное левое положение; *д* – промежуточное правое положение

Необходимость взятия изображений с различными положениями кисти обусловлена их смещением при движении кости сустава, что служит определённой сложностью для распознавания и детектирования границ.

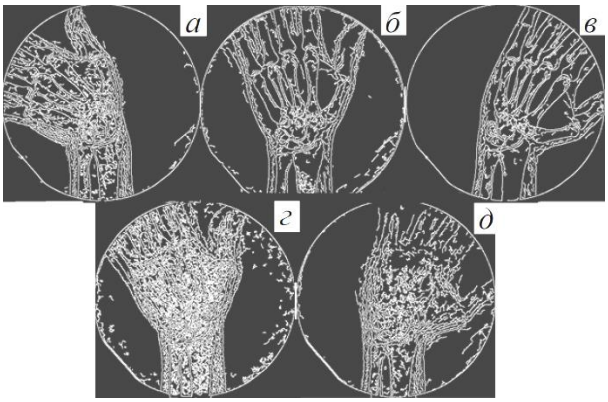


Рис. 3. Обработка исходных изображений методом Канни: *а* – крайнее левое положение; *б* – центральное положение; *в* – крайнее правое положение; *г* – промежуточное левое положение; *д* – промежуточное правое положение

Как видно, обработанные изображения сильно зашумлены, и провести анализ является затруднительной задачей. Поэтому было принято решение адаптировать данный метод обработки к приведённым изображениям.

Адаптация метода

Так как на получившихся изображениях много лишних элементов и запястные кости кисти прорисовываются нечётко, необходимо подобрать для каждого положения свои параметры, а именно коэффициенты сглаживания и размеры ядра фильтра, которые сведены в таблицу.

Подбор осуществлялся вручную для каждого положения и основывался на максимальной визуальной видимости области интереса.

Экспериментальным путём установлено, что приведённые в таблице параметры являются оптимальными для большинства изображений такого качества, независимо от того, что на них изображено.

Параметры сглаживания

Положение кисти	Коэффициент сглаживания σ	Размер ядра
Крайнее левое	0,08	2
Центральное	0,065	2
Крайнее правое	0,065	2
Промежуточное левое	0,065	3
Промежуточное правое	0,05	3

Далее реализована обработка исходных изображений методом Канни с подобранными параметрами сглаживания. Результаты обработки приведены на рис. 4.

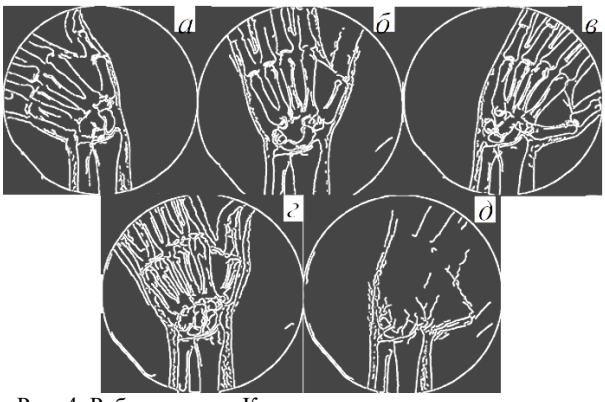


Рис. 4. Работа метода Канни с заданными параметрами: *а* – крайнее левое положение; *б* – центральное положение; *в* – крайнее правое положение; *г* – промежуточное левое положение; *д* – промежуточное правое положение

Влияние предварительной фильтрации

Исходные изображения были подвергнуты предварительной обработке фильтром, повышающим их резкость. Повышение резкости подразумевает под собой высокочастотную фильтрацию, т.е. увеличение уровней высоких частот в изображениях. Самым эффективным методом является обработка скользящим окном, имеющим небольшой размер. Из-за особенностей аппаратуры и погрешностей методов клинической диагностики на получаемом изображении присутствуют шумы, в основном аддитивного и импульсного характера [6, 7].

Выбор фильтра обусловлен необходимостью подчеркнуть яркостные переходы и выделить более мелкие детали изображения для более качественного выделения границ.

На рис. 5 представлены результаты выделения границ после применения фильтра, повышающего резкость изображения. На них чётче выделяются запястные суставы, и при более детальном рассмотрении можно визуально определить их полные границы.

Фильтр задан посредством функций библиотеки Image Processing Toolbox в ПО MATLAB и имеет следующие параметры: размерность 5×5, стандартное отклонение равно 0,5.

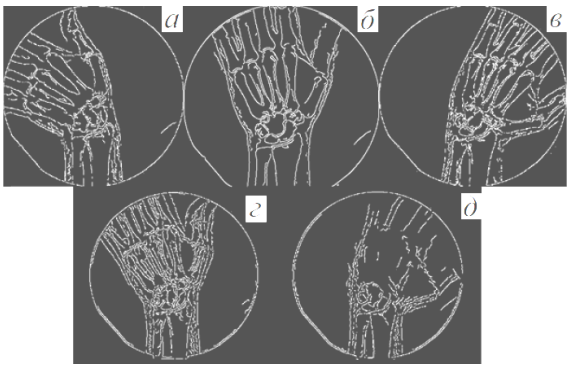


Рис. 5. Применение метода Канни к предварительно обработанным изображениям: *а* – крайнее левое положение; *б* – центральное положение; *в* – крайнее правое положение; *г* – промежуточное левое положение; *д* – промежуточное правое положение

На итоговых изображениях также видно, что появилось большее количество артефактов, но визуально границы запястных костей стали лучше определяться, также восстановлены «разъеденные» края костей сустава.

Заключение

Для исследования был реализован метод выделения границ Канни. Выполнен анализ полученных изображений и проведена модификация данного метода. В результате получены оптимальные коэффициенты для обработки изображений. Получены признаки, по которым можно определить размер ядра сглаживающего фильтра и степень сглаживания для метода Канни.

Таким образом, если начальное визуальное качество изображений достаточно хорошее и положение кисти отклонено от центрального, то коэффициент сглаживания будет равен примерно 0,065 при размере ядра, равном 2; если же положение центральное, то коэффициент сглаживания составит примерно 0,08. При этом начальное визуальное качество плохое, независимо от положения кисти на изображении, размер ядра фильтра будет равен 3 при коэффициенте сглаживания 0,065.

Проведено сравнение результатов обработки исходных и предварительно обработанных изображений. Как видно из полученных результатов, обработка дала более качественный визуальный результат. Также в изображениях устранился «разрыв» линий границ кости. Из этого следует, что обработка методом Канни сказывается лучше на изображениях, прошедших предварительную обработку двумерным фильтром повышения резкости изображения.

Из полученных результатов обработки изображений следует, что степень прорисовки границ зависит как от изначального качества изображений, так и от того, находится кисть в движении или нет. При этом запястные кости сустава кисти руки прорисовываются достаточно чётко для более качественного анализа. Приведённые результаты при дальнейшей работе позволят разработать универсальную методику фильтрации биомедицинских изображений, что, соответственно, улучшит их качество как для визуального восприятия врача-диагноста, так и для медицинской техники, оснащённой средствами компьютерного зрения.

Литература

1. Буй Т.Т.Ч., Спицын В.Г. Анализ методов выделения краев на цифровых изображениях // Доклады ТУСУР. – 2010. – № 2(22). – С. 221–223.

2. Курячий М.И. Цифровая обработка сигналов // Томск: ТУСУР, 2009. – 190 с.

3. Детектор границ Канни [Электронный ресурс]. – Режим доступа: https://habr.com (дата обращения: 17.06.2022).

4. Федотов А.А. Методы компьютерной обработки биомедицинских изображений в среде MATLAB / А.А Федотов, С.А. Акулов, А.С. Акулова. – Самара: Изд-во СГАУ, 2015. – 88 с.

5. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, В. Вудс. – М.: Техносфера, 2005. – 1072 с.

6. Беззубик В.В Метод повышения резкости цифровых изображений // Научно-технический вестник информационных технологий, механики и оптики. – 2014. – № 6. – С. 82–89.

7. Хомидов М.Э., Гоипов Э.А. Методы обработки биомедицинских сигналов и изображений // Universum: Технические науки. – 2020. – № 8. – С. 32–34.

Бояркин Степан Евгеньевич
Магистрант каф. ТУ ТУСУРа
Эл. почта: boyarkin-stepan@mail.ru

Каменский Андрей Викторович
Канд. техн. наук, доцент каф. ТУ ТУСУРа
Эл. почта: andru170@mail.ru

Секция 19

ЦИФРОВАЯ СВЯЗЬ И ВЕЙВЛЕТ-ФРАКТАЛЬНАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ

Председатель секции – ***Голиков Александр Михайлович****, доцент каф. РТС, к.т.н.*

УДК 621.376

А.С. Дернова

Высокоскоростная система сотовой связи на базе беспилотной аэроплатформы

Частные сети имеют преимущества перед традиционными решениями подвижной радиосвязи благодаря площади покрытия, глубине проникновения сигнала, низким задержкам в передаче сигнала, а также безопасности и простоте подключения оконечных устройств.

Ключевые слова: Simulink Matlab, QAM, SNR, BER, помехоустойчивость, шумы.

Корпоративная беспроводная сеть создается в рамках одной организации и изолирована от общедоступных сетей мобильной связи.

Данный стандарт обеспечивает скорость передачи данных выше 100 Мбит/с.

Изолированная сеть работает эффективнее, чем Wi-Fi, потому что она может удовлетворять потребности предприятий и бизнес-структур.

Одной из важных особенностей является способность обеспечивать дополнительную гибкость при использовании спектра.

Схема представляет собой 4 базовых и 5 опциональных компонентов. Структурная схема представлена на рис. 1.

Результаты работы модели для адаптивного случая модели Adaptive System with OSTBC представлены в табл. 1.

Таблица 1

Зависимость FER (частоты ошибок в кадре) от SNR для адаптивного случая

SNR, дБ	0	5	10	15	20	25	30
FER	1	0,711	0,199	0,189	0,065	0,022	0,008

Для этого в блоке Adaptive Control Panel переключатели переведены в соответствующее положение (рис. 2).

Результаты работы для модели MIMO 4×4 представлены в табл. 2.

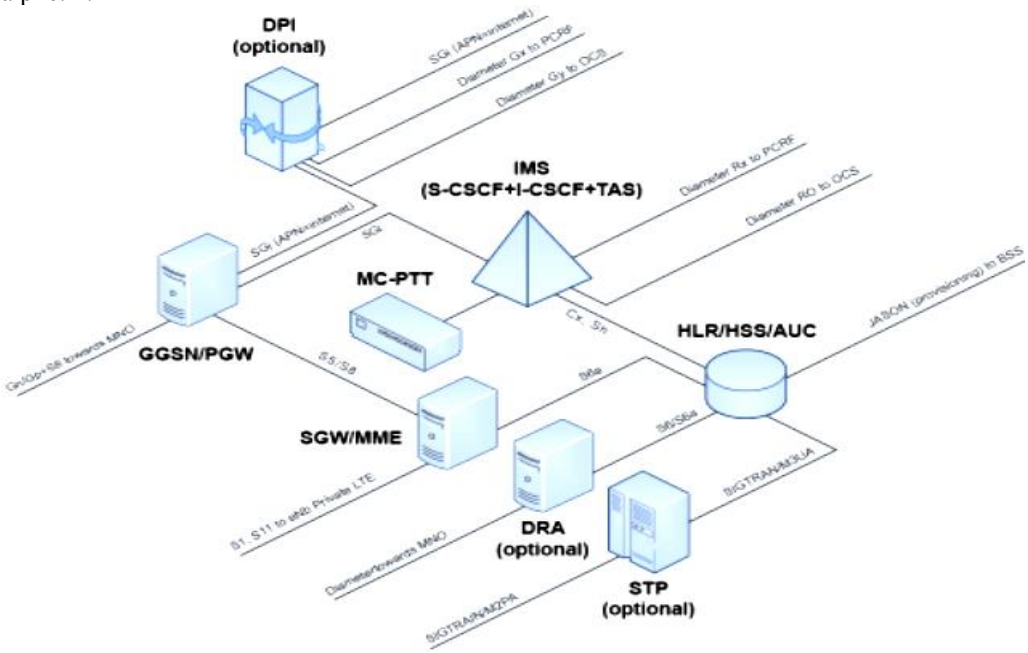


Рис. 1. Структурная схема сети

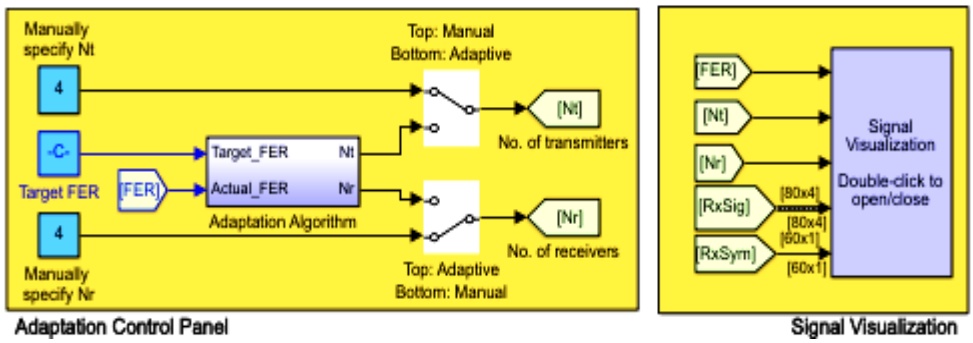


Рис. 2. Положение переключателей в блоке Adaptive Control Panel

Таблица 2

Зависимость FER (частоты ошибок в кадре) от SNR

SNR, дБ	0	5	10	15	20	25	30
FER	1	0,696	0,018	0	0	0	0

Построим зависимость вероятности частоты ошибок от отношения сигнал/шум (рис. 3).

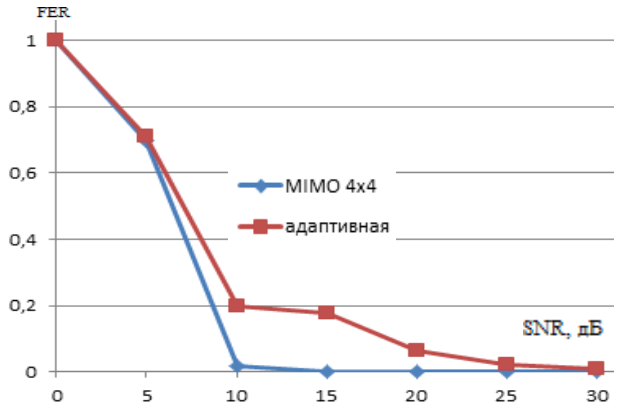


Рис. 3. Зависимость вероятности ошибок от отношения сигнал/шум

В работе было проведено исследование помехоустойчивости системы связи с использованием модели Simulink Matlab, а именно: зависимость битовой вероятности ошибки (BER – bit error rate) от отношения энергии бита к спектральной плотности мощности шума (E_b/N_0) для разных видов модуляций.

Схема, составленная в программном обеспечении Simulink, приведена на рис. 4.

Схема включает в себя: блок Random Integer, который является источником случайного сигнала; блок Rectangular M-QAM Modulator Baseband позволяет производить модуляцию M-QAM, и для демодуляции стоит блок Rectangular M-QAM Demodulator Baseband1; блок AWGN channel имитирует шум в канале с заданным SNR; в блоке Display в первом окошке наблюдаем BER, во втором – количество ошибочных бит, в третьем – переданных бит.

На рис 5–7 представлены модулированные созвездия при M-QAM без шумов и на различных значениях SNR.

На рис. 8 представлены графики зависимостей BER от SNR для различных уровней M-QAM.

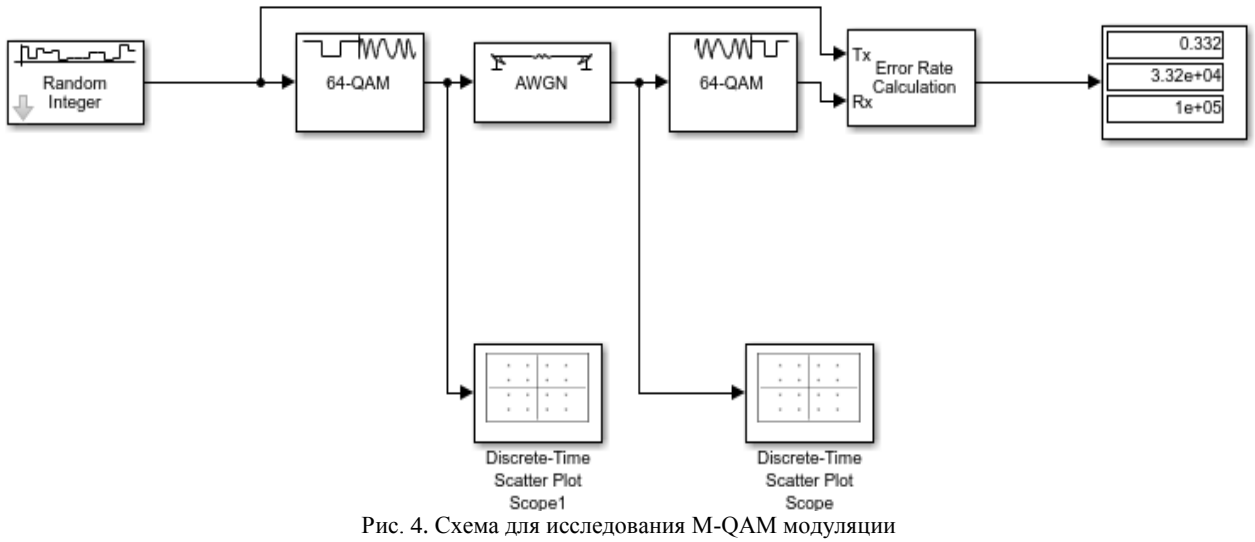


Рис. 4. Схема для исследования M-QAM модуляции

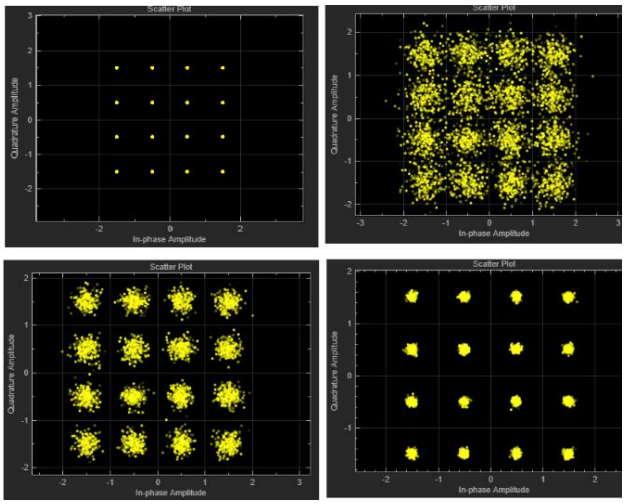


Рис. 5. Сигнальное созвездие для 16-QAM: без шумов; с шумом в 10 дБ; 15 дБ и 25 дБ

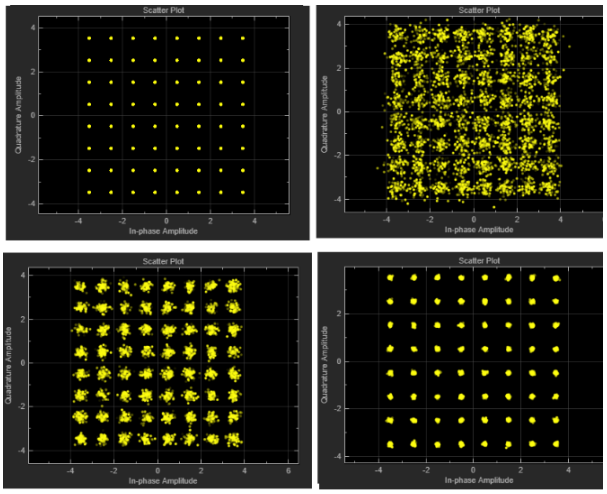


Рис. 6. Сигнальное созвездие для 64-QAM: без шумов; с шумом в 10 дБ; 15 дБ и 25 дБ

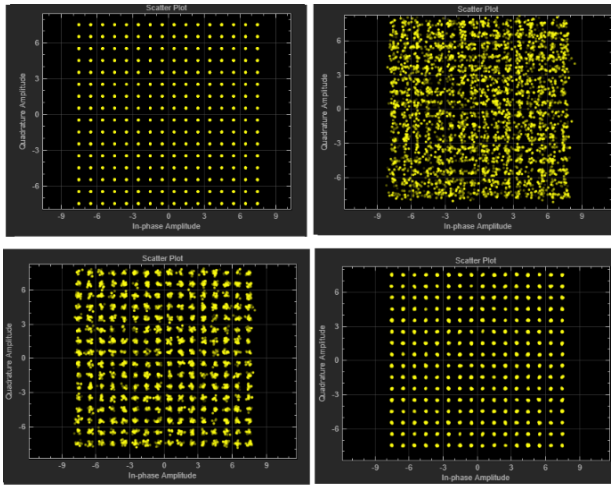


Рис. 7. Сигнальное созвездие для 256-QAM:
без шумов; с шумом в 10 дБ; 15 дБ и 25 дБ

Исходя из полученного графика, можем сделать вывод, что чем выше уровень манипуляции, тем хуже помехоустойчивость, так как расстояние между соседними точками становится меньше.

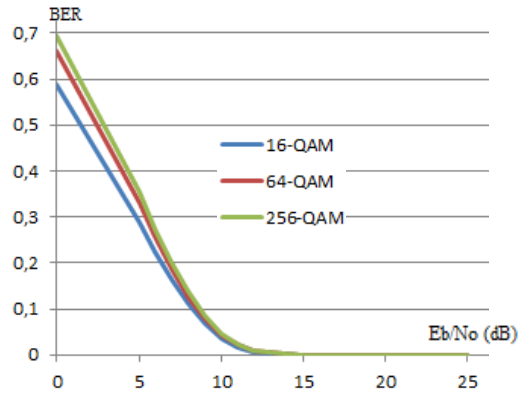


Рис. 8. График зависимости BER от SNR для M-QAM

Литература

1. Голиков А.М. Модуляция, кодирование и моделирование в телекоммуникационных системах. Теория и практика: учеб. пособие. – СПб.: Лань, 2018. – 452 с.

Дернова Алина Сергеевна
Студентка каф. радиотехнических систем (РТС) ТУСУРа
Эл. почта: Alinadernova611@gmail.com

УДК 621.376

М.М. Муруева

Исследование системы мобильной связи UMTS на базе беспилотной аэроплатформы

Проведено исследование помехоустойчивости, быстродействия и энергетики системы мобильной связи UMTS (CDMA). Локальная система мобильной связи UMTS предназначена для обеспечения мобильной связи в удалённых районах на базе беспилотной аэроплатформы ООО «Системы. Технология. Коммуникации».
Ключевые слава: CDMA, последовательность Уолша, канал прямого трафика, мобильный приемник, сотовая связь, помехоустойчивость.

Метод CDMA охватывает одновременно три характеристики: частоту, время и энергию. По стандарту работа основывается на шифровании сигнала в код, который во множественном числе отправляется одним каналом до получателя. И раз существует шифрование, то на приёмнике, чтобы получить необходимый сигнал, он расшифровывается тем же кодом, который использовался при отправке.

По системе CDMA существуют два вида передачи – прямое и обратное. Из двух видов образовалась структура по стандарту IS-95, представленная на рис. 1.

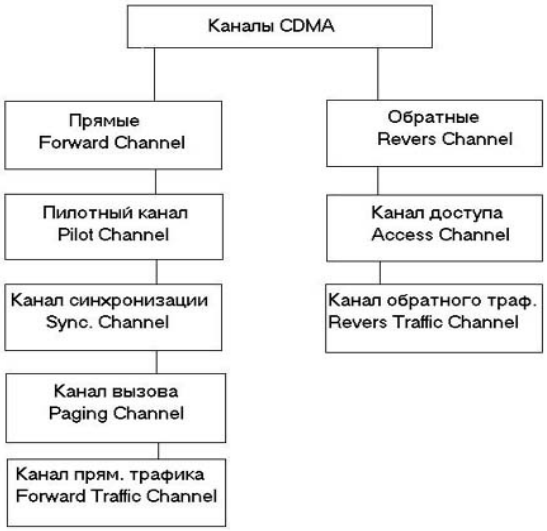


Рис. 1. Структура каналов в CDMA

В прямом канале расположены блоки пилотного канала, канала синхронизации, канала вызова и канала прямого трафика. В краткой форме сигнал подвергается отслеживанию, установке соединения, распоряжение полным доступом к соединению базовой станции и прямое соединение с базовой станцией.

В обратном канале расположены блоки канала доступа и канал обратного трафика. В краткой форме сигнал подвергается бесплатному соединению с базовой станцией для рассмотрения работоспособности и даёт доступ управления информацией с возможностью адресации сообщений.

При синхронном предоставлении услуг множествам клиентов CDMA-системы употребляют метод прямого расширения спектра, созданный по первоисточнику КПШП [2]. Схема изображает следующие детали, что отправленные символы модулируются тремя КПШП: длинной и короткой КПШП, включая КПШП Уолша (рис. 2).

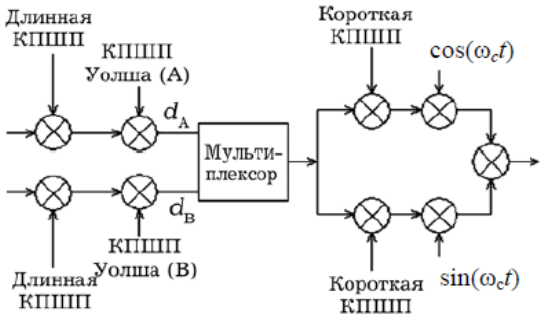


Рис. 2. Схема передачи CDMA-системы

Длинная КПШП идентична абсолютно каждой сети базовой станции, предназначенная оказать безопасность отправления данных, и для полной работы схемы был сформирован 42-разрядный регистр сдвига.

Короткая КПШП опознаёт базовую станцию в сети и генерируется благодаря 15-разрядным регистрам сдвига. Существует уникальная особенность в короткой КПШП для всех базовых станций, что применяются две ортогональные.

Число КПШП Уолша выявляет максимальное число параллельно оказываемых услуг клиентам лишь одной базовой станцией.

Цифровая радиосвязь способна предоставлять мультидоступ, предназначенный для синхронной передачи информации мгновенно всем пользователям, соединённым с общим каналом и в одном устройстве. Примечательно то, что мультидоступ способен поделить общий канал на частоту (FDMA), время (TDMA) и код (CDMA), это разделение можно увидеть на рис. 3.

В нашей модели CDMA в передатчике, представленном на рис. 4, создан заранее длинный программный код, из чего после образовывается сигнал в векторную форму вследствие установленного блока «To Column Vector». Образованный вектор отправляется по каналу и до самого приёмника.

Когда обнаруживается сигнал в начале модели приёмника, сигнал течёт в RC-фильтр, АЧХ данного

фильтра изображена на рис. 6. После того как сигнал профильтровался, необходимо усилить сигнал блоком Gain, благодаря которому сигнал не пропадёт по пути. Дальше необходимо поделить сигнал на квадратурные части, умножив мнимую часть на «–1», и после в обратном порядке собрать реальную и мнимую части. Восстанавливаем созвездие благодаря Rake-приёмнику и подаём сигнал окончательно на демодуляцию. В итоге символы модуляции переходят на декодер.

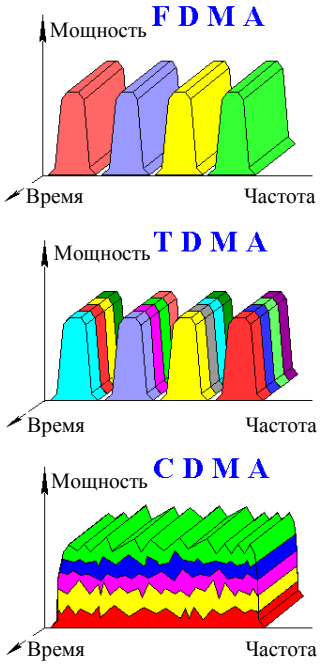


Рис. 3. Разделение общего канала по частоте, времени и коду

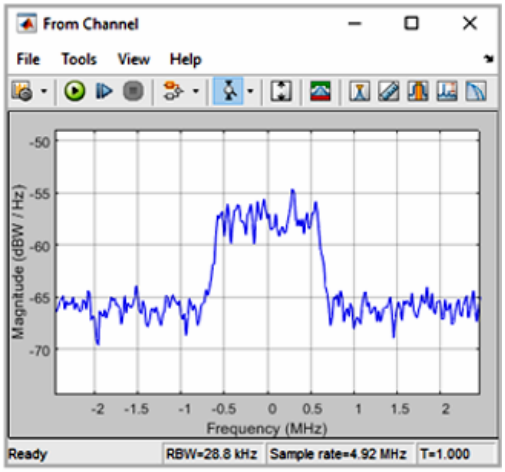


Рис. 4. Осциллограмма спектра сигнала с шумами

Приобретённые символы модуляции протекают по задержке на 756 тактов, чтобы потом добиться сбора до 756 символов. После заполнения до 756 символов даётся разрешение на протекание сигнала дальше по модели для обратного перемежения. Дальше на сигнал воздействует возобновление нулевых последовательностей, которые замещены на специальные символы ради последовательного воздействия сигнала при помощи обратного преобразования кодов с коэффициентом повторения 2. В итоге блоки «Viterbi Decoder» и «Encoder Tail Bits» заключительно подвергают изменения сигнала прибавлением нулей и ограничением количества бит до 80.

BER (Bit Error Rate) – частота возникновения ошибок. В системах цифровой передачи данных возможны возникновения ошибок из-за шума, помех, искажения или битовой рассинхронизации. Вероятность битовой ошибки – это ожидаемое значение коэффициента ошибок по битам.

Допустим следующее условие, что ошибок не возникает при нулевом значении BER, или, в ином смысле, ошибки успешно ликвидированы. Тогда при таких обстоятельствах необходимо уравновесить фреймы при помощи прохождения через задержки на сигнал базовой станции.

Радиопокрытие волнами в канале наземной подвижной связи говорит о том, что в области видимой территории будет существовать большое количество отражений, дифракций, поэтому данная технология подвергнется многолучевому распространению, которое ведёт к ослаблению энергии сигнала благодаря естественным препятствиям.

Функция «Open Scopes» в правом верхнем углу отображает все осциллограммы, которые имеют высокую значимость по процессу преобразования сигнала от начала и до конца. Результат моделирования проведён для режима AWGN Channel и представлен на рис. 5, 6.

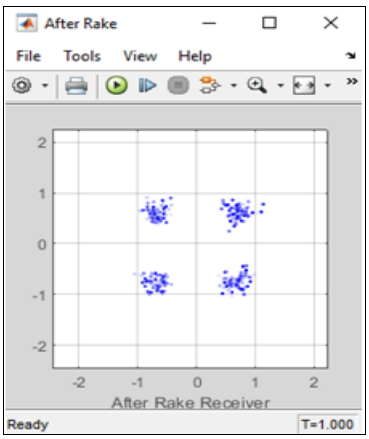


Рис. 5. Осциллограмма декодированного сигнала с шумами

График по результатам в таблице представлен на рис. 6.

Зависимость частоты возникновения ошибок от отношения сигнал/шум в канале с шумами

SNR	–30	–25	–20	–15	–10	–5
BER	0,478	0,4758	0,4471	0,000114	0	0

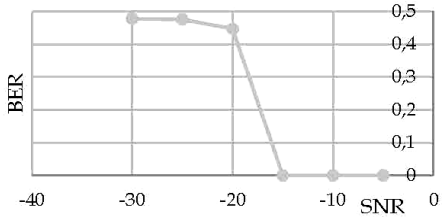


Рис. 6. Зависимость частоты возникновения ошибок от отношения сигнал/шум с шумами

Литература

1. Описание стандарта CDMAOne [Электронный ресурс]. – Режим доступа: http://kunegin.com/ref6/cdma/4.htm

2. Голиков А.М. Модуляция, кодирование и моделирование в телекоммуникационных системах. Теория и практика: учеб. пособие для вузов. – 2-е изд. – СПб.: Лань, 2021. – 452 с.

3. Голиков А.М. Исследование системы мобильной связи с кодовым разделением каналов CDMA: учеб.-метод. пособие по лаб. работе [Электронный ресурс]. – Томск: ТУСУР, 2019. – 22 с. – Режим доступа: https://edu.tusur.ru/publications/9178

Муруева Марина Матвеевна
Студентка каф. РТС ТУСУРа
Эл. почта: muruevamar@gmail.com

УДК 621.376

А.М. Голиков, Д.С. Кирпичников, А.Н. Жариков

Обнаружение и оценка координат малоразмерных наземных объектов MIMO РЛС с вейвлет-обработкой сигналов

Проведено исследование алгоритмов обнаружения малоразмерных наземных объектов (МНО) с использованием вейвлет-фильтрации Хаара и Добеши. Получены зависимости вероятности правильного обнаружения от отношения сигнал/шум в каналах MIMO 2×2 и MIMO 4×4 с использованием вейвлет-фильтров Хаара и Добеши при разных скоростях БПЛА. Проведено исследование алгоритмов оценок «ошибок» углового положения МНО с использованием вейвлет-фильтров Хаара и Добеши. Получены зависимости оценок «ошибок» углового положения МНО от отношения сигнал/шум при скоростях БПЛА – 20, 40 и 100 м/с для MIMO 2×2 и MIMO 4×4.
Ключевые слова: обнаружение, «ошибка» определения углового положения, вейвлет-фильтрация Хаара и Добеши, MIMO РЛС.

Вейвлет-преобразование похоже на оконное преобразование Фурье, но с иной оценочной функцией. Основное различие лежит в следующем: преобразование Фурье раскладывает сигнал на составляющие в виде синусов и косинусов, т.е. функций, локализованных в Фурье-пространстве; напротив, вейвлет-преобразование использует функции, локализованные как в реальном, так и в Фурье-пространстве [1, 2].

Вейвлет Хаара представлен на рис. 1.

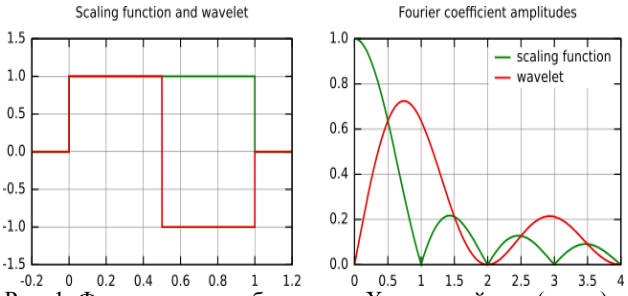


Рис. 1. Функция масштабирования Хаара и вейвлет (слева) и их частотные составляющие (справа)

В работе представлены Simulink MatLab-модели генератора радиолокационных (РЛ) сигналов ЛЧМ и ФКМ. Разработана Simulink MatLab модель радиолокационного канала с аддитивным шумом и доплеровским сдвигом (для разных скоростей носителя) для передачи/приема сигналов ЛЧМ и ФКМ, а также имитатором цели, позволяющим оценить дальность до цели в зависимости от ЭПР.

В результате моделирования получены сигналы на выходе РЛ-канала с шумом, отфильтрованный сигнал и аддитивный шум в РЛ-канале для вейвлет-фильтров Хаара, Добеши. Разработан программный комплекс, который включает в себя формирователь РЛ-сигналов, модель радиолокационного канала с аддитивным шумом и доплеровским каналом, вейвлет-фильтрами Хаара и Добеши и блоком имитации цели для и оценки эффективности обнаружения целей (рис. 2).

Зависимость вероятности правильного обнаружения (при вероятности ложной тревоги 0,0001 и ЭПР 1 м^2) от отношения сигнал/шум в канале MIMO 4×4 (слева) и MIMO 2×2 (справа) (зеленый – без фильтрации, синий – фильтр Хаара, красный – фильтр Добеши) показана на рис. 3.

Из графиков зависимости вероятности правильного обнаружения от отношения сигнал/шум (см. рис. 3) следует, что для радиолокационного канала MIMO 4×4 эффективность вейвлет-фильтрации увеличивается по сравнению с радиолокационным каналом MIMO 2×2. Превышение вероятности правильного обнаружения единицы объясняется «фильтрацией» при построении графиков в ПО Excel.

Для исследования влияния вейвлет-фильтрации на оценку параметров сигналов была разработана Simulink MatLab модель, в которую были добавлены каналы оценки «ошибки» углового положения цели [4, 5] (рис. 4).

Программный комплекс включает в себя формирователь РЛ сигналов, модель MIMO радиолокационного канала с аддитивным шумом, доплеровским каналом, вейвлет-фильтрами Хаара и Добеши, каналами оценки «ошибки» углового положения цели.

Зависимости оценок «ошибок» углового отклонения от отношения сигнал/шум в канале представлены на рис. 5.

В работе проведено исследование алгоритмов обработки сигналов при обнаружении с использованием вейвлет-фильтрации Хаара и Добеши. Показано, что для радиолокационного канала MIMO 4×4 эффективность обнаружения с использованием вейвлет-фильтрации выше, чем в радиолокационном канале MIMO 2×2. Показана зависимость эффективности обнаружения для разных скоростей носителя.

Исследование алгоритмов обработки сигналов при оценке их параметров с использованием вейвлет-фильтров Хаара и Добеши показало, что для MIMO 4×4 оценки «ошибки» углового положения уменьшаются на порядок и более по сравнению с MIMO 2×2. При увеличении скорости носителя оценки «ошибок» углового положения увеличиваются.

Показана возможность снижения отношения сигнал/шум при заданных вероятностях правильного обнаружения и ложной тревоги за счет использования вейвлет-преобразований, а также где вейвлет может достигать 8,9 дБ.

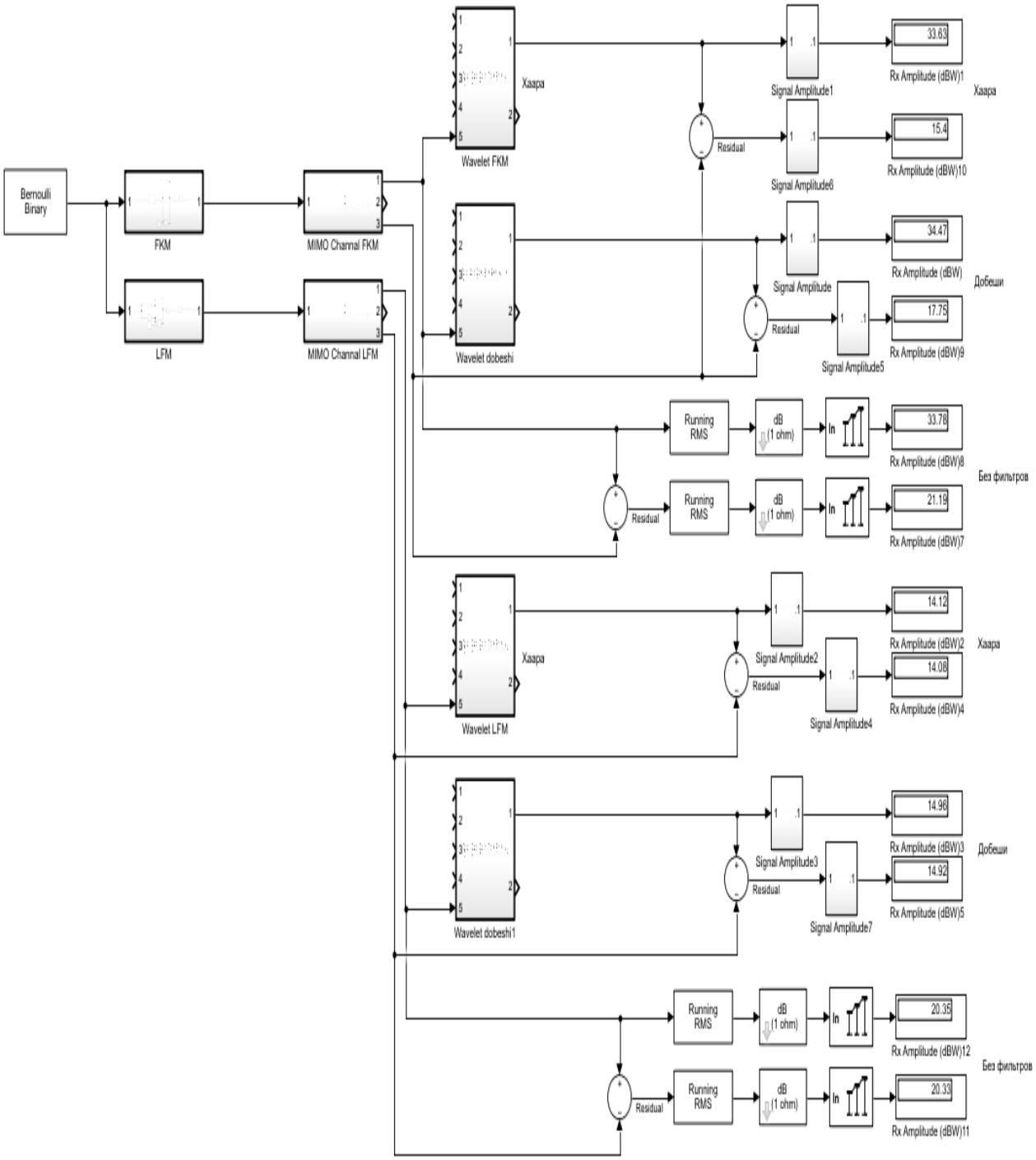


Рис. 2. Программный комплекс для и оценки эффективности обнаружения целей

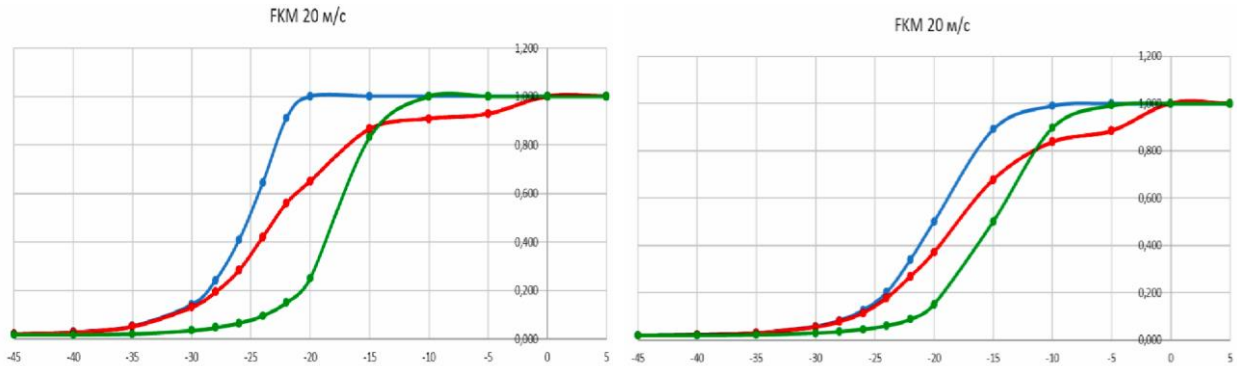


Рис. 3 (начало)

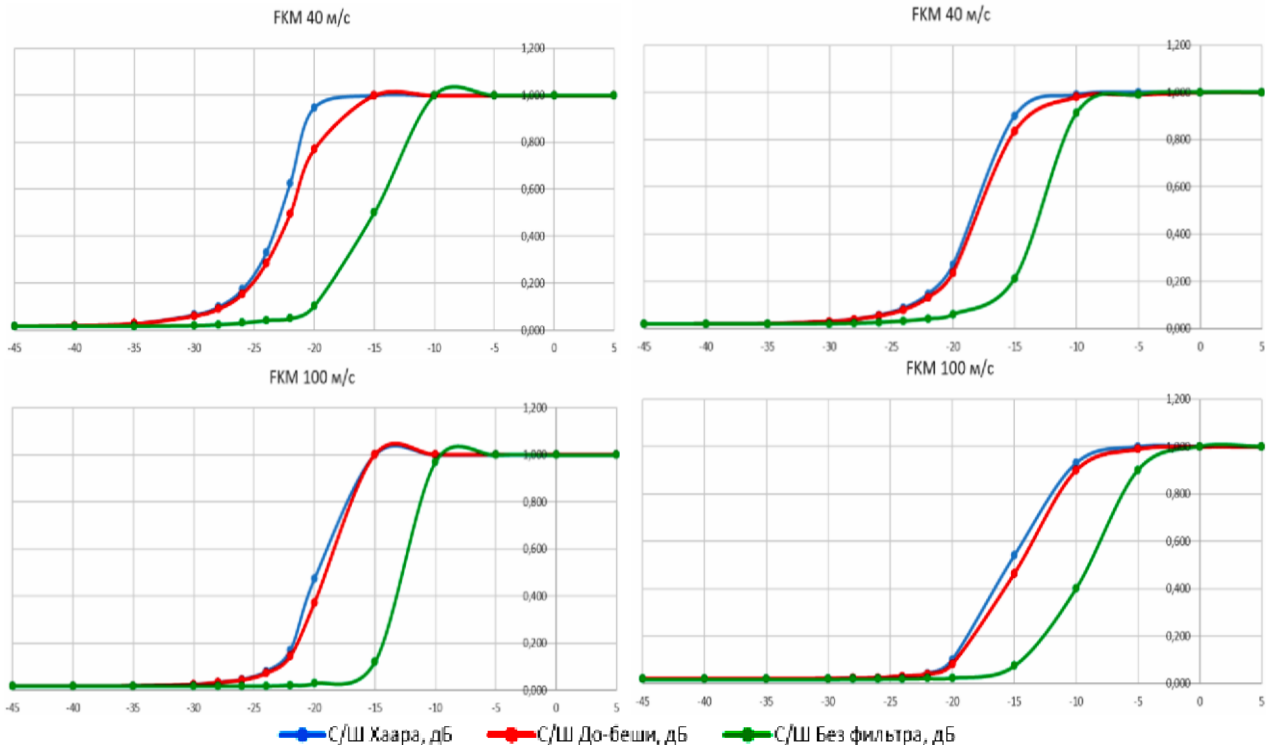


Рис. 3 (окончание). Зависимость вероятности правильного обнаружения от отношения сигнал/шум в канале – MIMO 4×4 (слева) и MIMO 2×2 (справа) (зеленый – без фильтрации, синий – фильтр Хаара, красный – фильтр Добеши) для скоростей БПЛА 20, 40 и 100 м/с (сверху вниз)

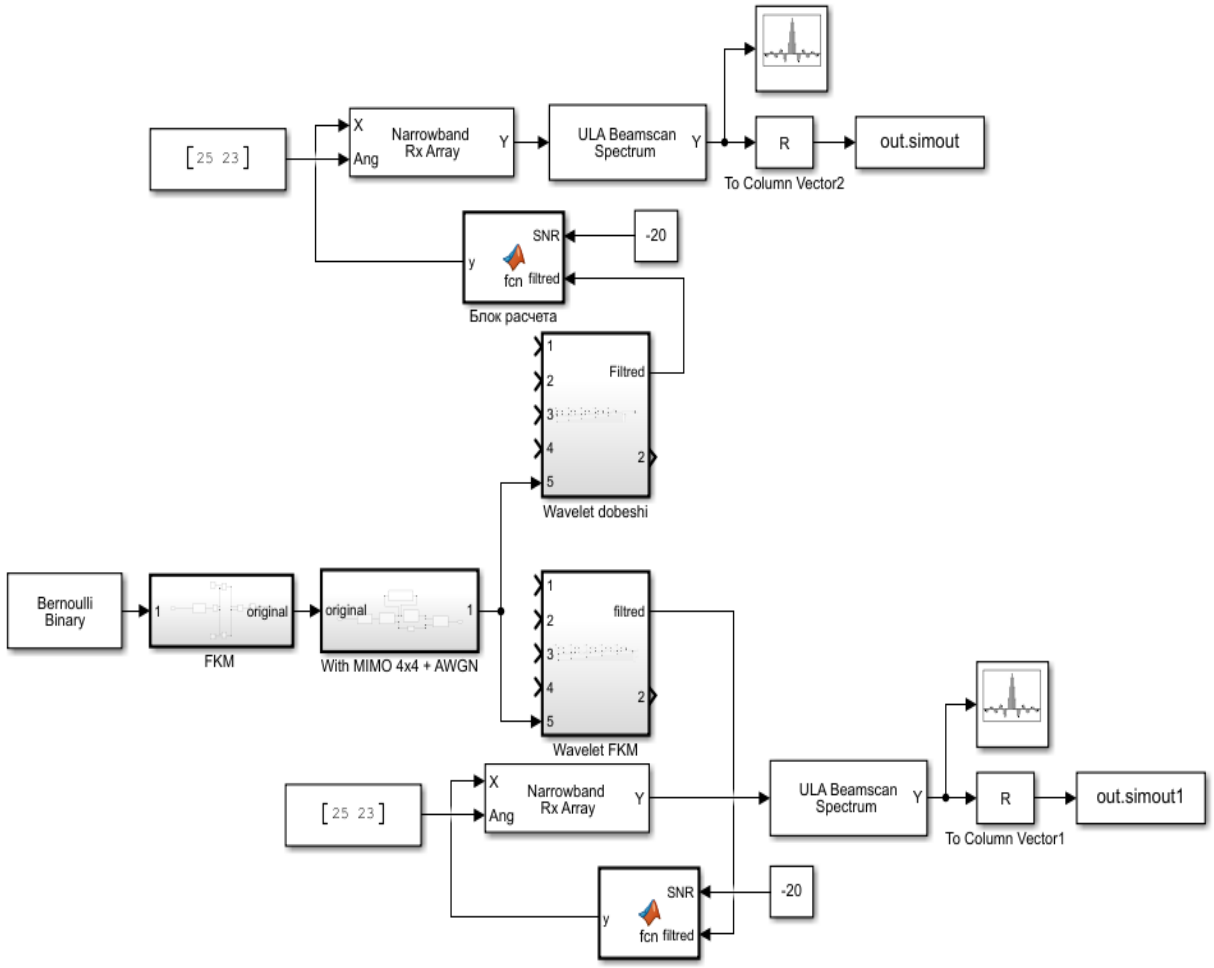


Рис. 4. Программный комплекс для исследования влияния вейвлет-фильтрации сигналов на ошибку определения углового положения цели

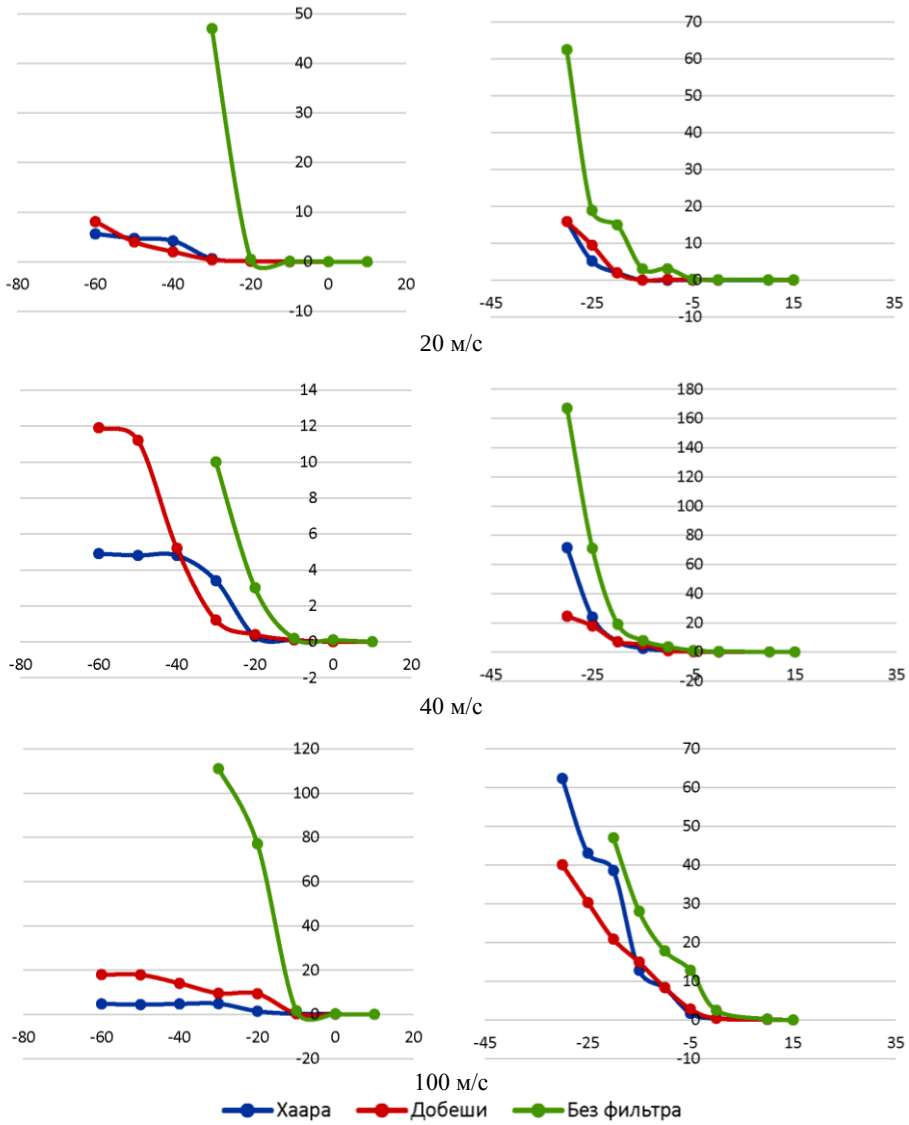


Рис. 5. Зависимости оценок «ошибок» углового отклонения от отношения сигнал/шум в канале MIMO 4×4 и MIMO 2×2 (справа) (зеленый – без фильтрации, синий – фильтр Хаара, красный – фильтр Добеши) для скоростей БПЛА 20, 40 и 100 м/с (сверху вниз)

Показана возможность повышения точности измерения параметров сигналов при использовании вейвлет-преобразований, а также при использовании вейвлетов Хаара и Добеши в два и более раз.

Литература

1. Голиков А.М. Системы цифровой радиосвязи: учебник. – М.: Ай Пи Ар Медиа, 2022. – 340 с.

2. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – 3-е изд., перераб. и доп.; пер. с англ. – М.: Техносфера, 2012. – 1104 с.

3. Голиков А.М., Рябинин Е.А., Кузьменко Ю.Д. Экспериментальные исследования и вейвлет-фильтрация сигналов морского радиолокатора на фоне взволнованной морской поверхности // Сб. докл. междунар. науч.-практ. конф. «Электронные средства и системы управления». – 2012. – С. 49–52.

4. Тейлор Дж. Введение в теорию ошибок / пер. с англ. – М.: Мир, 1985. – 272 с.

5. Шевченко М.Е. Алгоритмы современного обнаружения и пеленгования в широкой полосе обзора // Радиотехника. – 2012. – № 7. – С. 71–76.

Голиков Александр Михайлович
Канд. техн. наук, с.н.с., доцент каф. РТС ТУСУРа
Эл. почта: rts2_golikov@mail.ru

Кирпичников Дмитрий Сергеевич
Студент каф. РТС ТУСУРа
Эл. почта: kirpichnikov.d.127-@e.tusur.ru

Жариков Артем Николаевич
Студент каф. РТС ТУСУРа
Эл. почта: ar.zharikov2017@yandex.ru

УДК 621.376

Д.В. Скурихин, А.М. Голиков

Система передачи телеметрической информации LoRaWAN на базе беспилотной аэроплатформы

Объектом исследования является технология передачи данных для интернета вещей LoRaWAN, которая позволяет передавать данные с небольшой скоростью, но на очень большие расстояния. Датчики способны на протяжении 10 лет работать автономно без сети и проводов, а сигналы, поступающие от них, не глушатся стенами, крышами и даже погодными условиями.

Ключевые слова: LORAWAN, LORA, модуляция, FSK, MSK, GFSK, GMSK, LORA, помехоустойчивость.

Система LoRaWAN способна дистанционно отслеживать передачу информации с блока станции управления, затрубное и буферное давления, а также телеметрию забойного давления и температуры (рис. 1).

Модуляция FSK

FSK-вид манипуляции, с которой частота несущего сигнала изменяется скачкообразным образом в зависимости от значений символов информационной последовательности. Модель FSK представлена на рис. 2.

Данный вид манипуляции является самым помехоустойчивым. Результаты моделирования представлены на графиках (рис. 3–5).

На рис. 6 представлены графики зависимостей BER от SNR для различных уровней M-FSK.

Чем ниже значение M, тем лучше помехоустойчивость. Наиболее оптимальным выбором будет являться модуляция 2-FSK.

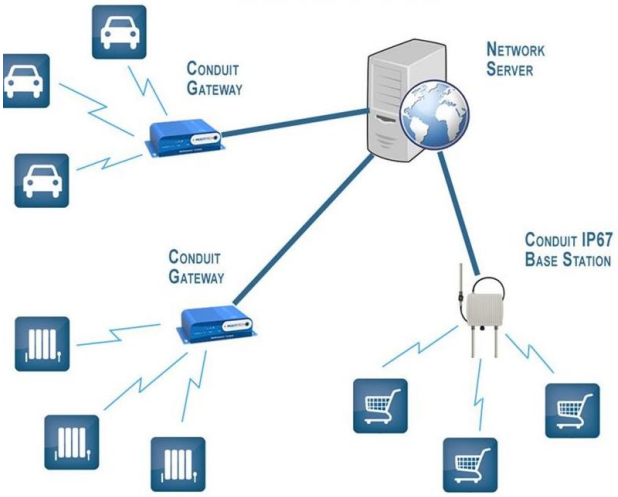


Рис. 1. Типовая беспроводная сеть LoRaWAN

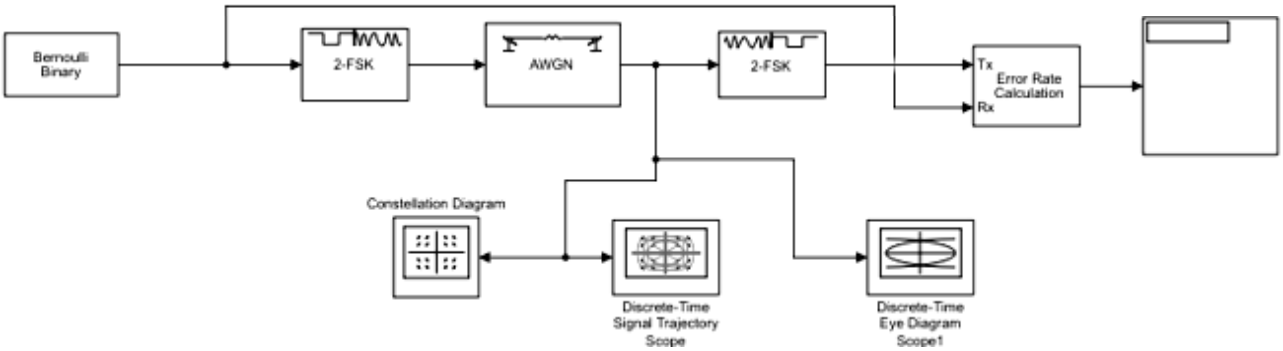


Рис. 2. Simulink-модель для исследования модуляции M-FSK

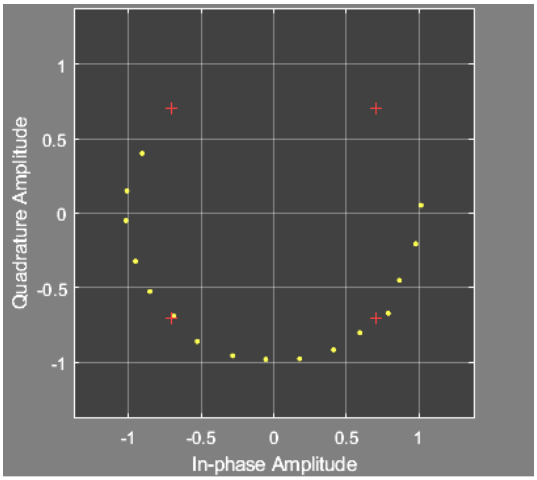


Рис. 3. Сигнальное созвездие полученного сигнала при $M = 2$

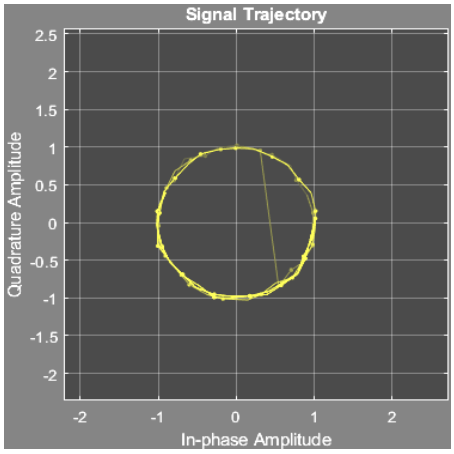


Рис. 4. Развёртка траектории сигнала с дискретизацией по времени при $M = 2$

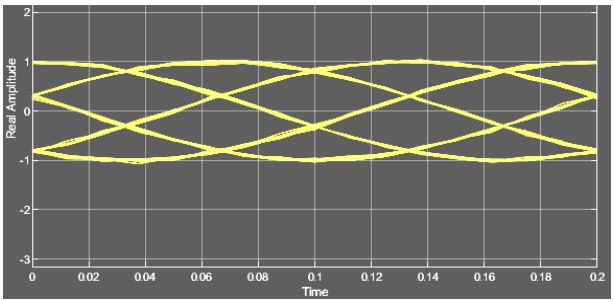


Рис. 5. Глазковая диаграмма сигнала при $M = 2$

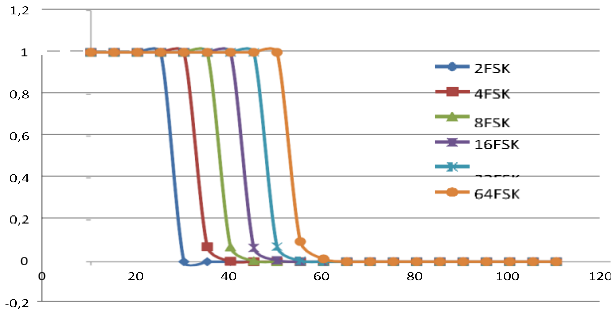


Рис. 6. График зависимости вероятности ошибки (BER) от отношения сигнал/шум (SNR) для M-FSK

Модуляция LORA

В отличие от других способов этот метод обеспечивает большую дальность зоны покрытия.

Данная технология позволяет осуществлять демодуляцию сигналов с уровнями на 19,5 дБ ниже уровня шумов.

Модель модуляции LoRa представлена в Mathlab Simulink на рис. 7.

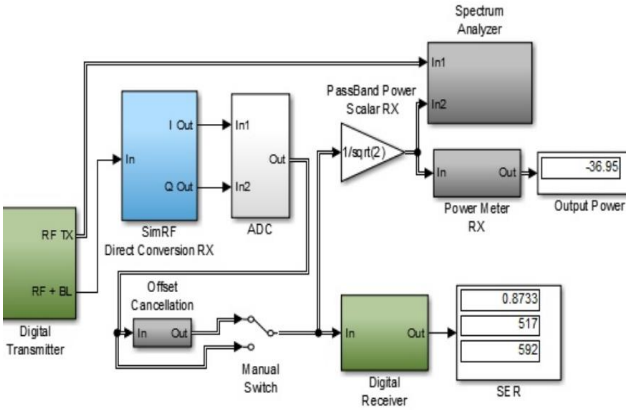


Рис. 7. Модель модуляции LoRa

Для того чтобы провести модуляцию, нужно задать частотное смещение, чтобы записывать битовую последовательность в сигнал.

Спектрограмму LoRa-модуляции можно увидеть на рис. 8.

Модуляция GFSK

Это вид частотной модуляции, при которой используется фильтр Гаусса.

Модель GFSK представлена на рис. 9.

Модуляция GMSK

Это гауссовская двухпозиционная частотная манипуляция с минимальным сдвигом.

Используется в системах радиосвязи, а также служит для обеспечения высокого качества передачи в узкой полосе, занимаемой сигналом (рис. 10).

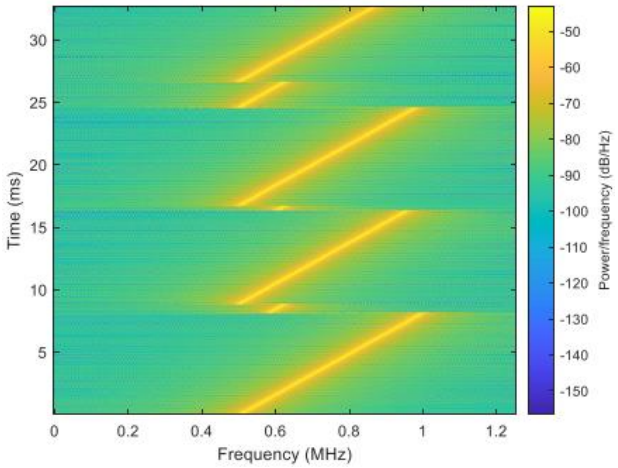


Рис. 8. Спектрограмма LoRa-модуляции

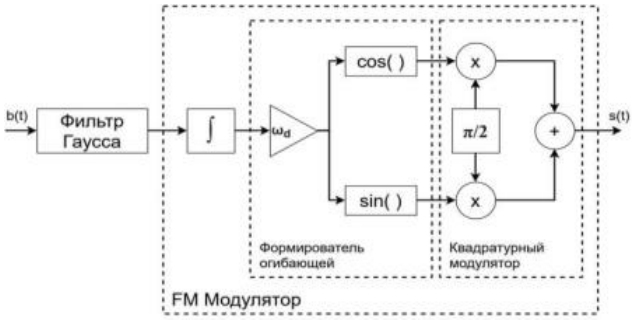


Рис. 9. Структурная схема модулятора GFSK

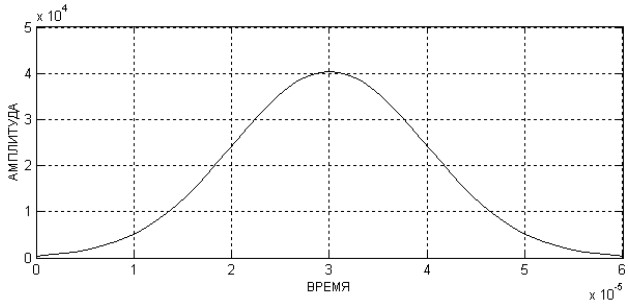


Рис. 10. График импульсной характеристики Гаусса

Литература

1. Голиков А.М. Системы цифровой радиосвязи: учеб. пособие. – М.: Ай Пи Ар Медиа, 2022. – 340 с.

Скурихин Даниил Вадимович
Студент каф. РТС ТУСУРа
Эл. почта: dencheezzz@gmail.com

Голиков Александр Михайлович
Канд. техн. наук, с.н.с., доцент каф. РТС ТУСУРа
Эл. почта: rts2_golikov@mail.ru

Секция 20

ВЫЧИСЛИТЕЛЬНЫЙ ИНТЕЛЛЕКТ И МАШИННОЕ ОБУЧЕНИЕ

Председатель секции – ***Ходашинский Илья Александрович****, профессор каф. КСУП, д.т.н.*

УДК 004.021

М.Б. Бардамова

Выбор меры оценки качества классификации для построения нечеткого классификатора на несбалансированных данных

Основным критерием эффективности классификатора считается способность выдавать правильный результат, т.е. его точность. Расчет точности проводится не только во время оценки готовой модели, но и в процессе её построения с применением инструментов машинного обучения. Выбор адекватной меры оценки качества классификации при дисбалансе данных имеет первостепенную важность. В данной работе исследуется влияние используемой меры оценки качества на результат построения нечеткого классификатора несбалансированных наборов данных с двумя классами. Были опробованы пять мер, для расчета которых достаточно знать только классы самих данных и классы, полученные классификатором: общая точность, F1-мера, коэффициент Жаккара, сбалансированная точность и средняя геометрическая точность. Предложены оригинальная гибридная метрика и способ расчета коэффициента для неё в зависимости от коэффициента дисбаланса данных.

Ключевые слова: машинное обучение, нечеткий классификатор, несбалансированные данные, меры качества классификации, метаэвристический алгоритм, алгоритм оптимизации, алгоритм прыгающих лягушек.

Разработчики моделей классификации на основе машинного обучения нередко сталкиваются с проблемой несбалансированного характера данных, когда количество экземпляров разных классов существенно отличается. При наличии дисбаланса важно выбрать в качестве целевой функции такую меру, которая позволяет адекватно оценивать качество классификации как во время обучения модели, так и после его завершения. Использование традиционной меры – общей точности – ведет к переобучению на классах с наибольшим представительством; при этом классы с наименьшим числом экземпляров могут быть абсолютно проигнорированы [1]. Таким образом, при получении высокой оценки эффективности модели без дополнительных оценок разработчик может и не иметь понятия, что классификатор работает неверно.

Актуальна эта проблема и для решающих алгоритмов, построенных на базе нечеткой логики. В данной работе исследуются альтернативные меры качества, позволяющие улучшить распознавание редких классов нечеткими классификаторами. Поскольку выходом классификатора, основанного на нечетких правилах, является степень принадлежности, а не вероятность, перечень возможных мер оценки качества, не нуждающихся в многоступенчатых расчетах, ограничен. Далее рассмотрены только такие оценки, которые рассчитываются на основе двух векторов – действительных меток классов из таблицы наблюдения и меток классов, определенных классификатором. Кроме того, в работе предлагается оригинальная гибридная функция оценки качества классификации несбалансированных данных и способ определения коэффициента приоритета в этой функции в зависимости от степени дисбаланса экземпляров разных классов.

Меры качества классификации

Введем следующие обозначения: n_k – число экземпляров с меткой k-го класса, $k \in [1, K]$, K – число классов в исследуемом наборе данных; n_k^* – количество экземпляров, верно отнесенных классификатором к k-му классу. Тогда точность k-го класса может быть определена как доля правильной классификации

$$c_k = n_k' / n_k \,. \qquad (1)$$

Точности отдельных классов могут быть использованы разработчиком классификатора как объективные показатели качества построенной модели, но для применения в роли фитнес-функции алгоритмов обучения в рамках однокритериальной оптимизации необходима единая комплексная мера оценки качества.

Общая точность (*acc*) – это традиционная мера эффективности классификатора, определяющаяся путем подсчета доли правильно определенных классификатором экземпляров среди общего их количества

$$acc = \sum_{k=1}^{K} n_k^* \Big/ \sum_{k=1}^{K} n_k \,. \qquad (2)$$

Так как приведенная формула не учитывает дисбаланс классов, она не может являться адекватной мерой оценки классификатора несбалансированных данных. Далее описаны альтернативные способы оценки; все указанные меры принадлежат промежутку от нуля, что соответствует худшему значению, до единицы – лучшего значения.

Часть мер качества, предназначенных для проведения оценки эффективности модели на несбалансированных данных, ограничена возможностью работы только на двух классах. Эти меры опираются в расчетах на такие показатели, как число истинно положительных экземпляров (*tp*), число истинно отрицательных экземпляров (*tn*), а также количество ложно положительных и ложно отрицательных образцов (*fp* и *fn*, соответственно). Если экземпляр принадлежал к наименьшему классу и был верно отнесен классификатором к нему же, экземпляр считается истинно положительным. Если же положительный экземпляр был ошибочно отнесен к противоположному классу, он считается ложно отрицательным. Аналогичным образов верно определен-

ный образец наибольшего класса считается истинно отрицательным, а ошибочно отнесенный к наименьшему классу – ложно положительным. Одной из таких оценок является $F1$-мера:

$$F1 = tp / (tp + 0,5(fp + fn)). \quad (3)$$

В описанной формуле напрямую не учитывается количество правильно определенных экземпляров наибольшего класса; она направлена на увеличение количества истинно положительных образцов при минимизации числа ложно положительных и ложно отрицательных образцов. Подобной мерой является и коэффициент Жаккара (KJ):

$$KJ = tp / (tp + fp + fn), \quad (4)$$

отличающийся от $F1$ только отсутствием коэффициента у суммы количества неверно классифицированных образцов.

Такие меры затруднительно использовать для данных, насчитывающих больше двух классов. Для этого набор данных должен быть разделен на множество пар вида «один класс против остальных», осуществлено построение классификатора на каждом множестве и полученные оценки усреднены. Кроме того, даже при наличии только двух классов в расчетах никак не учитывается ситуация возникновения неклассифицированных экземпляров данных.

Существуют меры качества, достоинством которых является отсутствие ограничений на число классов. Сбалансированная точность (BA) – это среднее арифметическое точностей всех классов:

$$BA = \sum_{k=1}^{K} c_k \Big/ K. \quad (5)$$

Средняя геометрическая точность (GM) для K классов вычисляется как корень K-й степени из произведения их точностей:

$$GM = \sqrt[K]{\prod_{k=1}^{K} c_k}. \quad (6)$$

У последней меры качества есть еще одно достоинство: в случае, когда один из классов будет полностью проигнорирован классификатором, т.е. если ни один ему принадлежащий экземпляр не будет правильно распознан, средняя геометрическая точность будет равняться нулю. Для разработчика возникновение такой ситуации послужит надежным индикатором, что на дисбаланс классов нужно обратить пристальное внимание.

Гибридная мера качества классификации

Вклад в среднюю геометрическую точность каждого правильно определенного экземпляра тем больше, чем меньше экземпляров этого класса в наборе. Это свойство позволяет сместить фокус обучения на наименьшие классы. Однако при высоком дисбалансе, когда на один экземпляр положительного класса приходятся десятки или сотни образцов отрицательного, значимость последних падает настолько, что появляется существенный риск потерь в точности наибольшего класса. Объединение в одну гибридную функцию двух противоположных мер – средней геометрической точности и общей точности – позволит гибко управлять приоритетом между классами

$$F_\gamma = \gamma \cdot GM + (1-\gamma) \cdot Acc, \quad (7)$$

где γ – коэффициент, принадлежащий промежутку $[0;1]$. Коэффициент отвечает за приоритет между двумя мерами и, как следствие, между положительными и отрицательными классами. Выбор коэффициента зависит от требований, предъявляемых к классификатору. Чем ближе коэффициент к единице, тем сильнее фокус обучения классификатора будет направлен на наименьшие классы. Нетривиальной задачей является подбор коэффициента при равенстве приоритета между классами. Закономерным решением является вычисление γ по количественной оценке степени дисбаланса данных:

$$\gamma = 1 - (1 / IR), \quad (8)$$

где IR – коэффициент дисбаланса, равный отношению числа экземпляров наибольшего класса к числу экземпляров наименьшего.

Эксперимент

Эксперимент направлен на исследование влияния мер качества на результат построения нечеткого классификатора несбалансированных данных. Для формирования структуры нечеткого классификатора был использован алгоритм экстремальных значений признаков классов, оптимизация параметров термов проводилась метаэвристическим алгоритмом прыгающих лягушек. Оба алгоритма отличаются высокой скоростью работы и в комплексе позволяют получить классификатор, сопоставимый по точности с аналогичными инструментами построения нечетких классификаторов, использующих базу правил большего объема [2]. Метаэвристика запускалась по десять раз на пятикратной кроссвалидации, далее значения точности были усреднены. Были установлены следующие параметры алгоритма прыгающих лягушек: 50 векторов в популяции, 5 мемплексов, 20 глобальных и 25 локальных итераций, коэффициент в операторе обновления вектора const = 1,2.

Для эксперимента использованы несбалансированные наборы данных из категории «высокий дисбаланс» репозитория KEEL [3]. Все наборы насчитывают только два класса. В табл. 1 приведен перечень данных, их коэффициент дисбаланса и полученный нечетким классификатором процент ошибки на наименьшем классе (количество неправильно классифицированных образцов по отношению к общему числу экземпляров). Полужирным начертанием выделены лучшие полученные значения ошибки.

Результаты показывают, что с распознаванием положительного класса лучше всего справились сбалансированная точность, средняя геометрическая точность и гибридная мера с коэффициентом приоритета, равным 0,75 или зависящим от коэффициента дисбаланса. В то же время применение общей точности, F1-меры и коэффициента Жаккара на некоторых наборах привели к получению ошибки свыше 90 %.

В табл. 2 содержатся полученные проценты ошибки классификации на наибольшем классе.

Таблица 1

Усредненные ошибки классификации наименьшего класса

Набор данных		*Acc*	*F*1	*KJ*	*BA*	*GM*	F_γ			
Название	*IR*						γ = 0,75	γ = 0,5	γ = 0,25	γ = 1–1/*IR*
yeast2vs4	9,1	36,6	35,2	37,8	22,8	21,9	24,6	28,5	30,6	**19,3**
yeast05679vs4	9,4	66,8	68,9	69,1	32,2	**31,6**	34,6	39,7	44,8	33,2
vowel0	10,0	36,3	35,7	36,0	6,7	**5,6**	8,2	13,4	26,2	9,2
glass2	11,6	98,2	96,5	99,5	**35,2**	37,0	41,5	51,8	73,2	45,7
ecoli4	15,8	21,0	23,0	26,5	**14,5**	16,0	16,5	17,5	21,5	17,5
glass4	15,5	62,3	54,3	60,3	27,7	29,0	**26,0**	28,7	29,3	33,0
page-blocks13vs2	15,9	11,8	17,1	16,8	10,3	11,1	**8,0**	12,4	9,4	8,6
abalone9-18	16,4	72,6	74,2	76,9	34,9	**32,9**	40,0	44,5	53,9	**32,9**
yeast1458vs7	22,1	99,0	99,3	97,7	**49,3**	50,3	61,7	71,0	85,0	55,7
yeast2vs8	23,1	49,5	51,0	48,5	45,0	**42,0**	45,5	46,0	46,5	44,0
yeast4	28,1	94,1	91,1	91,8	**26,9**	28,4	29,4	33,4	43,0	29,2
yeast1289vs7	30,6	98,7	96,3	97,0	**42,3**	44,3	51,0	70,3	83,7	47,0
yeast5	32,7	42,5	45,5	46,2	**10,2**	11,5	11,3	12,9	17,5	11,8
ecoli0137vs26	39,1	41,0	39,0	61,0	38,0	31,0	33,0	37,0	45,0	**30,0**
yeast6	41,4	63,4	67,1	70,3	21,4	**19,7**	20,3	24,9	35,7	21,4
Среднее	21,4	59,6	59,6	62,4	27,8	**27,5**	30,1	35,5	43,0	29,2

Таблица 2

Усредненные ошибки классификации наибольшего класса

Набор данных		*Acc*	*F*1	*KJ*	*BA*	*GM*	F_γ			
Название	*IR*						γ = 0,75	γ = 0,5	γ = 0,25	γ = 1–1/*IR*
yeast2vs4	9,1	1,9	1,9	**1,7**	8,3	7,1	5,6	4,1	2,8	6,6
yeast05679vs4	9,4	**2,0**	**2,0**	2,1	13,1	15,3	12,0	8,9	6,1	12,5
vowel0	10,0	1,4	**1,2**	**1,2**	8,9	8,5	7,9	5,8	2,9	6,1
glass2	11,6	2,0	1,6	**0,9**	26,8	26,1	20,4	15,9	8,0	21,3
ecoli4	15,8	**0,7**	0,8	1,0	2,4	2,4	2,1	2,2	2,0	2,2
glass4	15,5	2,2	**1,7**	2,1	4,7	5,1	3,8	3,0	3,6	4,8
page-blocks13vs2	15,9	**1,0**	1,3	1,1	2,3	2,2	1,7	1,9	1,4	1,7
abalone9-18	16,4	0,5	0,5	**0,3**	13,4	14,2	11,4	5,8	3,2	11,9
yeast1458vs7	22,1	0,4	0,3	**0,2**	28,3	27,1	18,5	11,7	4,1	23,3
yeast2vs8	23,1	**0,3**	0,4	**0,3**	6,9	8,5	2,8	1,2	0,7	6,7
yeast4	28,1	0,5	**0,4**	**0,4**	12,7	12,7	10,5	8,7	5,4	11,1
yeast1289vs7	30,6	**0,4**	**0,4**	**0,4**	26,0	23,4	18,6	9,0	3,1	19,9
yeast5	32,7	1,1	**1,0**	1,1	3,3	3,2	3,2	3,0	2,6	3,1
ecoli0137vs26	39,1	0,9	0,9	**0,8**	17,2	9,6	10,3	6,7	2,2	6,8
yeast6	41,4	0,6	**0,4**	0,5	6,7	7,0	6,0	4,5	2,7	6,1
Среднее	21,4	1,1	1,0	**0,9**	12,1	11,5	9,0	6,1	3,4	9,6

На отрицательном классе результаты показывают противоположную ситуацию: общая точность, F1-мера и коэффициент Жаккара позволили достичь лучшей точности. Однако если на наименьшем классе разница по среднему значению между лучшей (*GM*) и худшей (*Acc* и *F*1) мерой составила почти 32%, то на наибольшем классе разница между *KJ* и *BA* оказалась приблизительно равной 11%.

Таблица 3 представляет собой средние ранги, вычисленные критерием Фридмана по результатам классификатора. Нулевая гипотеза отклоняется в обоих случаях (асимптотическая значимость меньше 0,001), что говорит о наличии статистически значимых различий в результатах.

В табл. 3 также приведены расстояния до начала координат на фронте Парето. Сам фронт, построенный по средним рангам критерия Фридмана, продемонстрирован на рис. 1. Подсчет расстояния помогает установить, какая из мер качества способствовала получению нечеткого классификатора, достигнувшего наибольшей точности по обоим классам одновременно. Ближе всего к началу координат оказалась точка, соответствующая предложенной гибридной мере, сочетающей среднюю геометрическую и общую точность, с коэффициентом приоритета γ, равным 0,75. Близкие значения получили и другие точки, представляющие собой гибридную меру оценки.

Таблица 3

Средние ранги по критерию Фридмана

Мера качества	Ранг по меньшему классу	Ранг по большему классу	Расстояние до начала координат на фронте Парето
Acc	7,7	2,3	8,0
*F*1	7,7	2,0	7,9
KJ	8,4	1,7	8,6
BA	2,2	8,6	8,9
GM	2,1	8,2	8,5
γ = 0,75	3,1	6,2	6,9
γ = 0,5	4,9	5,2	7,1
γ = 0,25	5,9	4,1	7,2
γ = 1–1/*IR*	3,0	6,7	7,4

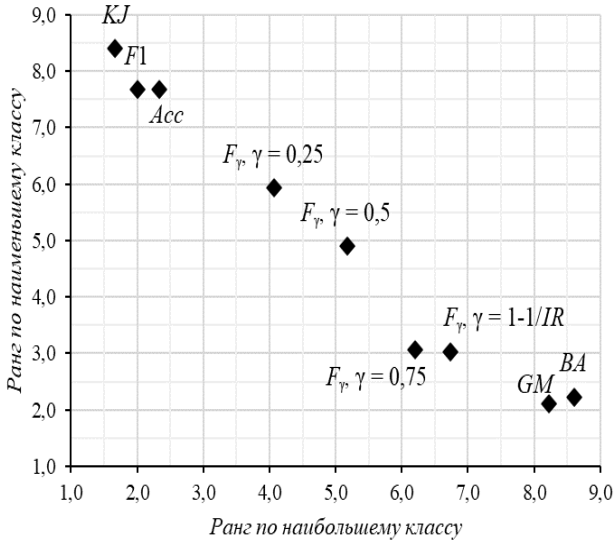


Рис. 1. Фронт Парето по средним рангам, полученным исследованными мерами оценки качества классификации

Заключение

Разработчик моделей классификации на несбалансированных данных должен внимательно относиться к выбору целевой функции для обучения. При построении нечеткого классификатора необходимо учитывать, что одни меры качества (общая точность, F1-мера, коэффициент Жаккара) прежде всего направлены на улучшение распознавания наибольшего класса, в то время как другие прежде всего сфокусированы на наименьшем (сбалансированная точность, средняя геометрическая точность). В случае наличия требования получить максимально высокую точность сразу по всем классам, а не только лишь наибольшему или наименьшему, полезно применение гибридной меры оценки качества, если нет возможности использовать многокритериальную оптимизацию.

Несмотря на то, что для преодоления дисбаланса часто применяются алгоритмы балансировки данных, на тестовых данных несбалансированный характер распределения классов сохраняется. Поэтому важен и выбор итоговых параметров, по которым будет оцениваться эффективность уже готовой модели. Подсчет точности каждого отдельного класса не является трудоемкой операцией, но дает разработчику важную информацию об эффективности классификатора.

Работа выполнена при поддержке Российского научного фонда (проект № 22-21-00021).

Литература

1. Gil M.A. Editorial of the special issue “Statistics with imperfect data”/ M.A. Gil, G. González-Rodríguez, R. Kruse // Information Sciences. – 2013. – Vol. 245. – PP. 1–3.

2. Hodashinsky I.A. Using shuffled frog-leaping algorithm for feature selection and fuzzy classifier design / I.A. Hodashinsky, M.B. Bardamova, V.S. Kovalev // Scientific and Technical Information Processing. – 2019. – Vol. 46. – PP. 381–387.

3. Imbalanced data sets for classification. Knowledge extraction based on evolutionary learning. Available at: https://sci2s.ugr.es/keel/imbalanced.php (Accessed: 07.09.2020).

Бардамова Марина Борисовна
Канд. техн. наук, ст. преп. каф. компьютерных систем в управлении и проектировании (КСУП) ТУСУРа
ORCID: 0000-0002-0567-590X
Эл. почта: 722bmb@gmail.com

УДК 004.021

А.В. Герасимов

Исследование эффективности метаэвристики Lion Algorithm при построении нечеткого классификатора несбалансированных данных

Исследуется эффективность использования метаэристического алгоритма льва (LA) для оптимизации термов в нечетком классификаторе. Алгоритм основан на социальном поведении львов и относится к генетическим алгоритмам. Классификация проводится на классических и несбалансированных наборах данных. Проверка эффективности проводится путем сравнения полученных результатов с аналогами.

Ключевые слова: классификация, нечеткие системы, несбалансированные данные, машинное обучение.

Задача классификации данных решается различными методами машинного обучения: нейронными сетями, деревьями решений, методом ближайших соседей и т.д.

Одним из вариантов решения задачи классификации является применение нечетких систем, основанных на правилах [3].

Нечеткий классификатор

Нечеткие системы опираются на базу правил вида «ЕСЛИ-ТО». В ЕСЛИ-части содержится перечень используемых в правиле термов (антецеденты), а в ТО-части находится метка класса (консеквент). Классификатор можно описать функцией (1):

$$c: \mathbf{X} \rightarrow \mathbf{Y}, \qquad (1)$$

где $\mathbf{X}$ – множество объектов, множество входных признаков; $\mathbf{Y}$ – множество меток классов;

Значение $Y_i = Y(\mathbf{a}, \boldsymbol{\theta}, \mathbf{S})$ является меткой для вектора признаков $\mathbf{a}$; $\boldsymbol{\theta}$ – вектор параметров классификатора, $\mathbf{S}$ – вектор участвующих в классификации признаков [3, 4, 8].

Процесс построения нечеткого классификатора можно разделить на два этапа:

1. Генерация структуры.

Совокупность базы нечетких правил и лингвистических термов составляет структуру нечеткого классификатора [3]. В данной работе будет использоваться алгоритм экстремальных значений признаков классов (АЭПК). Результат применения алгоритма – база правил, где каждый класс описывается одним правилом. Каждое правило содержит по одному терму для каждого признака [2].

Алгоритм экстремумов признаков классов (АЭПК) отличается простотой и высокой скоростью.

2. Оптимизация.

Оптимизация заключается в настройке параметров термов, отборе признаков, уточнении количества и видов правил, настройке весов.

Использование метаэвристики для нечеткого классификатора

Метаэвристики – не зависимые от предметной области стохастические алгоритмы, которые могут быть применены к широкому кругу задач оптимизации [1, kee].

Алгоритм льва (LA) будет принимать на вход готовую популяцию, в отличие от случайных величин, как на функциях. Популяция создается на этапе генерации структуры. Эту популяцию алгоритм оптимизирует, находит такие параметры, термы, которые увеличат его точность. Выходом алгоритма будет вектор параметров термов для нечеткого классификатора.

Алгоритм льва (LA)

Львы живут в социальной системе, называемой прайдом, что позволяет им укреплять свой вид в каждом поколении.

Доминирующий лев на территории (территориальный лев) контролирует территорию, сражаясь с другими нападающими животными, включая кочевых львов. Защита территории продолжается до достижения детенышами половой зрелости, вероятно, в течение 2–4 лет. В течение этих 2–4 лет кочевые львы пытаются вторгнуться в прайд. Между кочевыми львами и прайдом (за который отвечает территориальный лев) происходят частые бои на выживание для защиты территории. Если территориальный лев будет побежден, он будет убит или изгнан прайдом кочевого льва. Кочевой лев становится новым территориальным львом, принимая на себя власть над прайдом. Он убивает детенышей проигравшего льва, между львицей и новым территориальным львом начинается совокупление, в результате которого рождается потомство нового льва. Такое поведение продолжается до тех пор, пока детеныши на территории не станут взрослыми.

Для более приспособленной особи вероятность того, что в потомстве, полученном с ее участием, признаки, определяющие приспособленность, будут выражены еще сильнее [7, 9].

Как только детеныши становятся взрослыми, происходит захват территории. Как и при территориальной обороне, территориальный отстающий лев будет либо убит, либо изгнан из прайда полностью выросших детенышей. Новый, более сильный лев прайда изгоняет/убивает детенышей или слабых львов прайда и совокупляется с львицей прайда, чтобы родить своих собственных детенышей.

Базовая модель алгоритма льва была представлена как алгоритм поиска в 2012 г. [5]. Впоследствии она была реструктурирована с модернизацией в 2014 [6] и в 2018 гг. [7]. Он состоит из шести этапов обработки: 1) генерация прайда, 2) оценка фертиль-

ности, 3) спаривание, 4) территориальная защита, 5) захват территории, 6) остановка.

Генерация прайда является инициирующим процессом алгоритма льва.

Спаривание является наиболее ответственным процессом для получения новых львов (называемых потомством) от львов-родителей после оценки фертильности, которая является вторым процессом.

Территориальная защита и захват территории являются уникальными процессами по сравнению с другими алгоритмами оптимизации, поскольку они напрямую взяты из социального поведения льва. Эти два этапа играют главную роль в руководстве алгоритмом в определении оптимальных решений из огромного пространства поиска пространства.

Процесс завершения работы алгоритма льва зависит от проблемы и, следовательно, может быть основан либо на количестве итераций/генераций, либо на основе оптимальности полученных решений.

Псевдокод алгоритма:

Шаг 1: Инициализировать X^{male}, $X^{fe\ male}$.

Шаг 2: Вычислить $f(X^{male})$, $f(X^{fe\ male})$.

Шаг 3: Задать $N_g = 0$.

Шаг 4: Оценка фертильности; Если X^{male} или $X^{fe\ male}$ отстающие, заменить.

Шаг 5: Провести спаривание и получить потомство.

Шаг 6: Провести мутацию.

Шаг 7: Оценить потомство.

Шаг 8: Инициализировать A_{cub} как ноль.

Шаг 9: Защита территории: Провести защиту территории.

Инициализировать X^{nomad}.

Вычислить $f(X^{nomad})$.

Если $f(X^{nomad}) < f(X^{male})$, то обновить X^{male}, сбросить прошлое потомство, вернуться к шагу 4.

Шаг 10: Если $A_{cub} < A_{max} = 4$, вернуться к шагу 9.

Шаг 11: Захват территории: Если A^{m_cub} и A^{f_cub} оказались ближе к оптимальному решению, получить обновленные X^{male} и $X^{fe\ male}$.

Шаг 12: Увеличить N_g на единицу.

Шаг 13: Если не выполнен критерий останова, вернуться к шагу 4, иначе считаем вектор X^{male} оптимальным и завершаем процесс.

Эксперимент на классических наборах

Было проведено несколько экспериментов. Описание наборов, взятых из репозитория KEEL [10], приведено в табл. 1, где A – аббревиатура, *F* – количество признаков.

Таблица 1

Наборы данных

Название	A	*F*	Экземп.	Классы
banana	bnn	2	5300	2
haberman	hbm	3	306	2
titanic	tit	3	2201	2
iris	irs	4	150	3
balance	bln	4	625	3
bupa	bup	6	345	2
pima	pim	8	768	2
glass	gls	9	214	7
page-blocks	pbl	10	5472	5
magic	mag	10	19020	2
wine	win	13	178	3
cleveland	clv	13	297	5
heart	hrt	13	270	2
penbased	pbs	16	10992	10
segment	seg	19	2310	7
twonorm	twn	20	7400	2
satimage	sat	36	6435	7
coil2000	coil	85	9822	2

В табл. 2 представлены усредненные значения процента правильной классификации на выбранных наборах данных по 15 запускам, полученные с помощью нечетких классификаторов, структура которых сформирована алгоритмами LA, Gravitational Search (GS), Multi-objective fuzzy association rule-based classification algorithm (D-MOFARC), Fuzzy Association Rule-Based Classification Model for High-Dimensional Problems (FARC-HD) [2], где *R* – количество правил, *F* – количество признаков при использовании отбора, *L* – точность классификации (в процентах) на обучающих данных, *T* – точность (в процентах) на тестовых данных.

Таблица 2

Результаты тестирования нечетких классификаторов

Dataset	Тип функции принадлежности	GS				LA		D-MOFARC			FARC-HD		
		R	F	L	T	L	T	R	L	T	R	L	T
bnn	Треугольная	2	2	72,3	72,8	74,3	74,1	8,7	90,3	**89,0**	12,9	86,0	85,5
hbm	Треугольная	2	3	75,6	**74,4**	71,4	68,8	9,2	81,7	69,4	5,7	79,2	73,5
tit	Гауссова	2	3	77,8	78,6	75,3	73,9	10,4	78,9	78,7	4,1	79,1	**78,8**
irs	Треугольная	3	4	98,3	97,3	96,3	95,1	5,6	98,1	96,0	4,4	98,6	95,3
bln	Гауссова	3	4	83,7	81,8	79,7	78,0	20,1	89,4	85,6	18,8	92,2	**91,2**
bup	Треугольная	2	4	68,9	69	67,6	60,1	7,7	82,8	**70,1**	10,6	78,2	66,4
pim	Треугольная	2	2	76,9	74	76,5	74,7	10,4	82,3	75,5	20,2	82,3	**76,2**
gls	Гауссова	7	4	63,4	57,5	60,5	53,5	27,4	95,2	**70,6**	18,2	79,0	69,0
wis	Треугольная	2	4	96	96,3	91,9	95,0	9,0	98,6	**96,8**	13,6	98,3	96,2
pbl	Гауссова	5	2	90,8	90,8	87,5	87,6	21,5	97,8	**97,0**	18,4	95,5	95,0
mag	Гауссова	2	4	79,9	79,5	74,8	75,1	32,2	86,3	**85,4**	43,7	85,5	84,8
clv	Гауссова	5	2	63,4	**62,6**	57,8	53,6	45,5	90,9	52,8	42,1	82,2	58,3
hrt	Гауссова	2	6	86,5	84,1	83,1	81,1	18,7	94,4	**84,4**	27,8	93,1	83,7
pbs	Гауссова	10	8	55,1	55,0	62,4	62,2	119,2	97,4	**96,2**	152,7	97,0	96,0
seg	Треугольная	7	9	84,4	82,8	79,1	78,3	26,2	98,0	**96,6**	41,1	94,7	93,3
twn	Гауссова	2	14	94,4	94,4	97,0	**97,0**	10,2	94,6	93,1	60,4	96,5	95,1
sat	Гауссова	7	8	84,6	83,7	75,5	74,7	56,0	90,8	**87,5**	30,2	84,4	83,8
coil	Треугольная	2	1	92,2	92,1	94,0	**94,0**	89,0	94,0	94,0	2,6	94,0	**94,0**

Все выборки данных перед началом работы алгоритмов проходили через «минимаксную» нормализацию. В зависимости от обучающегося набора, для сравнения со статьей [2], использовались разные виды функции принадлежности. Для эксперимента использовалась десятикратная кроссвалидация.

На классических наборах данных метрикой качества классификации использована обобщенная точность accuracy, формула (2).

$$\mathrm{acc} = \sum_{j=1}^{m} \mathrm{inst}_j^* \Big/ \sum_{j=1}^{m} \mathrm{inst}_j \,, \qquad (2)$$

где inst_j^* – количество правильно определенных экземпляров j-го класса, inst_j – количество всех экземпляров j-го класса.

Сильное различие в количестве правил между GS и LA, с одной стороны, и D-MOFARC и FARC-HD – с другой, обусловлено тем, что в нечетких классификаторах с использованием метаэвристик GA и LA использован АЭПК, который позволяет получить базу правил минимального размера, где на каждый класс приходится строго одно правило.

Алгоритм LA показал наибольшую точность на двух наборах данных (coil2000 и twonorm). По критерию Фридмана были рассчитаны ранги – наилучшее значение у LA (табл. 3). При сравнении пары LA и GS алгоритм льва показал наибольшую точность в 6 из 23 наборов данных – это banana, phoneme, pima, penbased, twonorm, coil 2000. Отдельно для этих 6 наборов было также произведено сравнение с помощью критерия Уилкоксона.

Таблица 3

Критерий Фридмана, ранги, сравнение с аналогами

Алгоритм	Ранг	Алгоритм	Ранг
GS	2,28	D-MOFARC	3,28
LA	1,5	FARC-HD	2,94

Нулевая гипотеза гласит, что если между результатами нет статистических различий, уровень значимости равен 0,05. В эксперименте нулевая гипотеза была отклонена.

Эксперимент на несбалансированных наборах данных

Работа с несбалансированными данными приближена к работе с данными из реальной жизни, где имеются различия в количестве экземпляров классов. Во многих случаях именно определение редких классов является более приоритетной задачей.

Описание наборов данных из репозитария KEEL [12], выбранных для эксперимента, приведено в табл. 4, где *IR* – коэффициент несбалансированности. Выбирались те наборы данных, у которых отсутствуют пропуски и номинальные признаки. В наборах два класса (положительный и отрицательный).

Одним из способов борьбы с несбалансированными данными является правильный подбор метрики. Обобщенная точность accuracy не является подходящей метрикой для несбалансированных данных ввиду игнорирования экземпляров наименьшего класса. При использовании F1-меры считается true-positive, т.е. при этом игнорируется наибольший класс.

На несбалансированных наборах данных метрикой качества классификации было отдано предпочтение средней геометрической точности – geometric mean (GM) (3). Она рассчитываться на основе процента правильной классификации каждого класса.

$$GM = \left(\prod_{j=1}^{m} \frac{\mathrm{inst}_j^*}{\mathrm{inst}_j} \right)^{1/m} . \qquad (3)$$

Таблица 4

Наборы данных

Название	A	F	$\mathrm{inst}_{\mathrm{all}}$	inst^+	inst^-	*IR*
ecoli0v1	Ecl0/1	7	220	77	143	1,86
pima	Pm	8	768	268	500	1,87
yeast1	Yst1	8	1484	429	1055	2,46
ecoli1	Ecl1	7	336	77	259	3,36
ecoli2	Ecl2	7	336	52	284	5,46
yeast3	Yst3	8	1484	163	1321	8,1
ecoli3	Ecl3	7	336	35	301	8,6
page-blocks0	Pb0	10	5472	559	4913	8,79
page-blocks1-3vs4	Pb1-3/4	10	472	28	444	15,86
abalone9-18	Ab9/18	7/8	731	42	689	16,4
yeast1458vs7	Yst1458/7	8	693	30	663	22,1
yeast4	Yst4	8	1484	51	1433	28,1
yeast5	Yst5	8	1484	44	1440	32,73
ecoli0137vs26	Ecl0137/26	7	281	7	274	39,14
yeast6	Yst6	8	1484	35	1449	41,4

Для эксперимента использовалась пятикратная кроссвалидация. Перед сравнением с аналогами было решено сравнить результаты на разных функциях принадлежности и расчитать ранги по критерию Фридмана. Средние значения GM на тестовых выборках после применения оптимизации равны: 65,6 для гауссовой функции принадлежности (ранг 1,8), 77,7 для трапециевидной (ранг 2,4) и 74,7 для треугольной (ранг 1,8). Значения стремятся к максимуму, поэтому наибольший ранг является наилучшим. Количество запусков – 15. Проведено попарное сравнение полученных результатов до и после оптимизации с помощью критерия Уилкоксона. ССК (стандартизированная статистика критерия) показывает превосходство значений после оптимизации (табл. 5).

Таблица 5

Критерий Уилкоксона, проверка нулевой гипотезы

	p-value	ССК	Нулевая гипотеза
АЭПК – LA (гаусс.)	0,005	2,803	Отклоняется
АЭПК – LA (трап.)	0,009	2,599	Отклоняется
АЭПК – LA (треуг.)	0,005	2,803	Отклоняется

Следующим этапом было сравнение с аналогами. В табл. 6 представлены результаты и ранги по критерию Фридмана.

Наибольший ранг у алгоритма «прыгающие лягушки» (LF), затем «гравитационный поиск» (GS) и «алгоритм льва» (LA). Ранги рассчитаны с помощью критерия Фридмана.

Таблица 6

Результаты сравнения алгоритмов

Набор	LA	LF	GS
abalone-9_18	65,9	**83,0**	71,5
ecoli-0_vs_1	95,5	96,1	**96,2**
ecoli1	**89,46**	88,0	86,4
ecoli2	90,6	**91,0**	71,4
ecoli3	**87,1**	83,7	66,5
page-blocks0	83,3	81,2	**83,6**
page-blocks-1-3_vs_4	78,8	**95,4**	92,5
yeast-1-4-5-8_vs_7	**62,2**	60,9	52,8
yeast-2_vs_8	63,6	**69,5**	67,9
yeast1	67,73	**71,6**	69,4
yeast3	86,6	**91,1**	90,4
yeast4	75,65	**79,8**	78,9
yeast5	93,5	92,5	**94,3**
yeast6	83,3	**85,7**	84,8
pima	59,13	**72,1**	71,9
среднее	78,8	82,8	78,6
ранг	1,62	2,51	1,88

В трёх из пятнадцати наборах данных наилучший результат классификации показал «алгоритм льва» (LA). В восьми – LF, в 5 – GS. Средняя точность у алгоритма LA и GS различается на 0,2 в пользу алгоритма льва.

Заключение

Был сделан вывод о пользе применения метаэвристичесского алгоритма льва (LA), произведя сравнение с помощью критерия Уилкоксона тестовой точности до и после применения метаэвристики для оптимизации термов.

По результатам эксперимента на несбалансированных данных можно судить о том, что разработанный алгоритм настройки параметров термов может быть использован для построения нечетких классификаторов из различных предметных областей.

Литература

1. Вьюгин В.В. Математические основы машинного обучения и прогнозирования. – М.: МЦМНО, 2018. – 484 с.

2. Bardamova M. Fuzzy classifier with feature selection based on the gravitational search algorithm / M. Bardamova, A. Konev, I. Hodashinsky, A. Shelupanov // Symmetry. – 2018. – Vol. 10, Iss. 609. – P. 18.

3. Бардамова М.Б. Алгоритмы построения нечетких классификаторов несбалансированных данных на основе метаэвристик «гравитационный поиск» и «прыгающие лягушки». – Томск: ТУСУР, 2021. – 126 с.

4. Прикладные нечеткие системы / К. Асаи, Д. Ватада, С. Иваи и др.; пер. с яп. Ю.Н. Чернышова; под ред. Т. Тэрано, К. Асаи, М. Сугэно. – М.: Мир, 1993. – 368 с.

5. Rajakumar B.R. The lion's algorithm: a new nature inspired search algorithm // In: Second international conference on communication, computing and security. – 2012. – PP. 126–135.

6. Rajakumar B.R. Lion algorithm for standard and large scale bilinear system identification: a global optimization based on lion's social behavior // 2014 IEEE congress on evolutionary computation (CEC). – 2014. – PP. 2116–2123.

7. Rajakumar B.R. Optimization using lion algorithm: a biological inspiration from lion's social behavior // Evolutionary Intelligence. – 2018. – PP. 31–52.

8. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г.К. Вороновский, К.В. Махотило, С.Н. Петрашев, С.А. Сергеев. – Чарьков: ОСНОВА, 1997. – 107 с.

28. Бухенский К.В. О нечеткости правила Мамдани // Вестник РГРТУ. – 2021. – С. 85–97.

9. Sean L. Essentials of Metaheuristics. – Lulu, 2009. – 235 p.

10. Standard classification data sets. Knowledge extraction based on evolutionary learning [Электронный ресурс]. – URL: https://sci2s.ugr.es/keel/category.php?cat=clas (дата обращения: 11.03.2022).

11. Герасимов А.В. Метаэвристика lion algorithm для поиска минимума математических функций // Матер. междунар. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР–2022. – Томск: В-Спектр, 2022. – Ч. 3. – С. 76–79.

12. Imbalanced data sets for classification. Knowledge extraction based on evolutionary learning [Электронный ресурс]. – URL: https://sci2s.ugr.es/keel/imbalanced.php (дата обращения: 29.04.2022).

Герасимов Антон Васильевич
Студент каф. КИБЭВС ТУСУРа
Эл. почта: gerasimov.a.747@e.tusur.ru

УДК 004.934.2

П.Г. Букина, Д.В. Грунвальд, С.С. Харченко

Определение психоэмоционального состояния человека на основе анализа речи по частоте основного тона

Рассмотрены разработанные программы по составлению и анализу паттернов, а также выявлены некоторые признаки для различия эмоций.

Ключевые слова: паттерн, эмоция, частота основного тона.

Оценка психоэмоционального состояния человека по речи является актуальной задачей в таких областях, как криминалистика, медицина, системы контроля и управления доступом и т.д. [3]. Однако реализация данной задачи очень сложна. Основной проблемой является выделение существенных признаков, по которым можно судить об эмоциональном состоянии человека. Одним из вариантов решения данной проблемы является машинное обучение, где существует множество методов обработки аудиофайлов. Также при распознавании звуков используется долгая краткосрочная память, являющаяся видом рекурретных нейронных сетей. Она содержит модули, способные запоминать значения на разные промежутки времени [5].

Целью данной работы является реализация алгоритма формирования паттернов по частоте основного тона, а также выделение признаков, по которым можно определить эмоцию.

Алгоритм формирования паттернов

В рамках проекта была написана программа на языке программирования Python, перерабатывающая аудиозаписи в паттерны – наборы символов, где каждый символ имеет свой диапазон процентов «резкости» изменения частоты основного тона. Данная программа содержит три класса, которые применяются последовательно при составлении паттерна для конкретного аудиофайла по схеме

Аудиофайл → Массив частот (ЧОТ) →
→ Массив процентов → Паттерн.

Первый класс при помощи одного метода осуществляет настройку программы аналогично программе Praat, где указываются верхняя и нижняя границы частот в ГЦ, а также интервал разбиения файла в мс, и обработку поданного на вход звукового файла, конвертируя его в массив частот при заданных настройках и возвращая как результат работы метода.

Praat – это пакет программного обеспечения, написанный и поддерживаемый Полом Боерсмой и Девисом Веником из Амстердамского университета. Доступный бесплатно, с открытым исходным кодом, это один из лучших пакетов для анализа речи. Praat использует простой и надежный алгоритм для анализа речи, основанный на автокорреляции (метод запаздываний) [1].

Было проведено сравнение работы программы, содержащей данный класс, с программой Praat при одинаковых настройках, в результате чего были выявлены незначительные различия – около 0,06%. На рис. 1 представлен результат работы созданного класса, на рис. 2 – результат работы программы Praat.

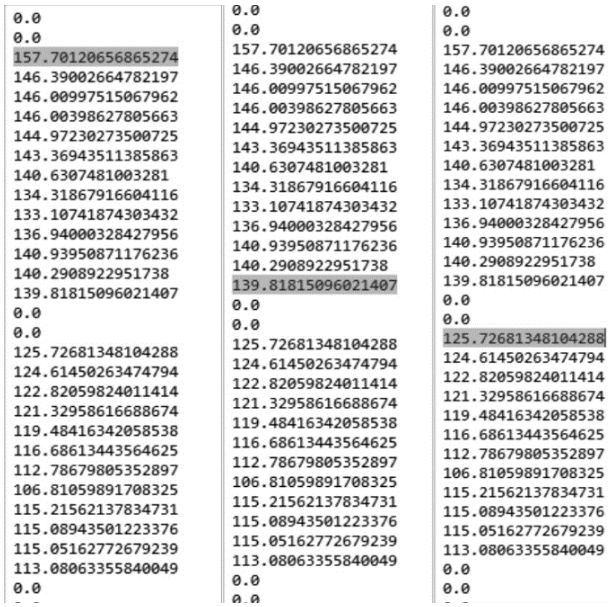


Рис. 1. Результат работы написанной программы

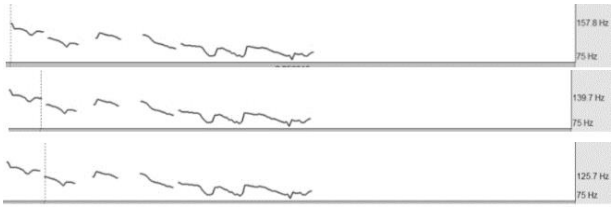


Рис. 2. Результат работы программы Praat

Второй класс отвечает за получение массива процентов из массива частот. Данный массив показывает процентное отношение максимального и минимального значений частот на отрезке времени, равном 10 мс. Расчет процентов нужен для дальнейшего определения границ интервалов для каждого символа. В классе был реализован метод, который производит расчет по формуле (1):

$$\Delta F = \frac{F_{\max} - F_{\min}}{F_{\min}} \times 100\%, \qquad (1)$$

где ΔF – абсолютное значение изменения частоты основного тона на выбранном участке времени в процентах; $F_{\max}$ – максимальное значение частоты основного тона за отрезок 10 мс; $F_{\min}$ – минимальное значение частоты основного тона за отрезок 10 мс.

Третий класс отвечает за формирование паттернов. В данном классе определяется массив кортежей следующего формата: [‘литерал’, минимальное значение процента «резкости» изменения ЧОТ, максимальное значение процента «резкости» изменения ЧОТ]. Данные кортежи впоследствии применяются для сопоставления конкретного элемента массива процентов и соответствующего литерала для отдельно взятых файлов. В результате чего все паттерны на выходе генерируются на основе одного общего множества литералов и соответствующих им интервалов процентов.

По написанной программе был обработан набор ravdess, содержащий аудиофайлы 24 актеров. В результате получился датасет, состоящий из папок для каждого актера, в которых хранятся полученные паттерны, использующие разное количество символов исходного алфавита. Исходный алфавит содержит 101 символ и состоит из букв латиницы и кириллицы. Минимальное количество используемых символов алфавита – 5, максимальное количество – 101.

Выделение признаков в паттернах

Следующим этапом было определение нужного количества символов в алфавите, при котором в паттернах можно было бы выделить отличительные признаки. При 5 символах определить характерные черты той или иной эмоции невозможно, как и при большом количестве. Было решено остановиться на 50 символах, т.к. при данном алфавите прослеживается зависимость наличия определенных символов в паттерне от частоты основного тона.

Далее была написана программа, выявляющая часто встречаемые символы в паттернах для каждой эмоции. Для выявления существенных признаков были взяты паттерны, построенные на основе алфавита, включающего 50 символов. Для наглядности в табл. 1 показаны интервалы для первых и последних 10 символов.

Таблица 1

Интервалы символов

Символ	Интервал	Символ	Интервал
a	0–0,3617	O	11,6081–12,4922
b	0,3617–0,5702	P	12,4922–13,5052
c	0,5702–0,7343	Q	13,5052–14,6945
d	0,7343–0,9073	R	14,6945–16,2668
e	0,9073–1,0735	S	16,2668–18,3125
f	1,0735–1,2262	T	18,3125–21,1401
g	1,2262–1,3863	U	21,1401–26,1199
h	1,3863–1,5439	V	26,1199–40,151
i	1,5439–1,7047	W	40,151–303,0192
j	1,7047–1,8572	X	303,0192–inf

В табл. 2 представлено число повторений каждого символа в паттернах для эмоций «спокойствие» и «гнев».

Самыми повторяющимися символами в паттернах для эмоции «спокойствие» являются «f», «b», «d», «a», «c». Самыми повторяющимися символами в паттернах для эмоции «гнев» являются «W», «n», «S», «Q», «R».

Таблица 2

Частота появлений символов в паттернах

Символ	Количество повторений	
	Спокойствие	Гнев
a	183	113
b	193	116
c	179	120
d	187	96
e	174	92
f	201	86
g	153	95
h	163	117
i	157	93
j	136	102
k	176	94
l	151	103
m	162	84
n	126	142
o	142	87
p	161	99
q	135	98
r	135	99
s	174	105
t	127	109
u	126	94
v	129	107
w	151	124
x	127	111
y	108	97
z	117	127
A	87	110
B	126	101
C	118	114
D	111	112
E	115	110
F	99	119
G	100	93
H	87	105
I	105	113
J	83	110
K	77	111
L	109	110
M	94	121
N	84	113
O	74	111
P	81	111
Q	64	130
R	63	129
S	64	133
T	67	123
U	28	128
V	30	116
W	22	148
X	1	5

Исходя из результатов, можно увидеть, что для эмоции «спокойствие» характерны символы, отражающие менее резкие скачки частоты основного тона, а для эмоции «гнев» – более резкие. Соответственно, эмоцию можно определить по тому, какие символы в паттерне встречаются чаще, а также какие комбинации символов в нем преобладают.

Заключение

Полученные паттерны имеют характерные особенности, которые в последующем можно использовать для обучения нейронной сети.

Литература

1. Praat: doing phonetics by computer [Электронный ресурс]. – Режим доступа: https://www.fon.hum.uva.nl/praat/, свободный (дата обращения: 15.09.2022).

2. Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound [Электронный ресурс]. – Режим доступа: https://www.fon.hum.uva.nl/paul/papers/Proceedings_1993.pdf, свободный (дата обращения: 15.09.2022).

3. Автоматическое определение эмоций по речи [Электронный ресурс]. – Режим доступа: https://www.iedtech.ru/files/journal/2012/3/determination-of-emotions.pdf, свободный (дата обращения: 15.09.2022).

4. Выбор решений при распознавании эмоций по речи [Электронный ресурс]. – Режим доступа: http://jmlda.org/papers/doc/2016/no4/Kalyan2016Emotions.pdf, свободный (дата обращения: 15.09.2022).

5. Распознавание речи при помощи нейронных сетей [Электронный ресурс]. – Режим доступа: http://meridian-journal.ru/site/article?id=4069&pdf=1&ysclid=l8aecczv tf167441168 (дата обращения: 15.09.2022).

6. Уздяев М.Ю., Рябинов А.В. Исследование подходов к классификации эмоций в невербальном речевом поведении на основе машинного обучения. – СПб., 2020.

7. Классификация эмоционального состояния диктора по голосу: проблемы и решения [Электронный ресурс]. – Режим доступа: https://www.dialog-21.ru/media/1420/20.pdf (дата обращения: 15.09.2022).

Букина Полина Германовна
Студентка каф. БИС ТУСУРа
Эл. почта: bukina.polina2014@gmail.com

Грунвальд Даниил Валерьевич
Студент каф. КИБЭВС ТУСУРа
Эл. почта: danil.grunvald@rambler.ru

Харченко Сергей Сергеевич
Канд. техн. наук, доцент каф. БИС ТУСУРа
Эл. почта: kss@fb.tusur.ru

УДК 004.934.2

К.Э. Горбунова, С.Г. Букина, С.С. Харченко

Обзор методов анализа речевых сигналов для оценки психоэмоционального состояния человека

Рассматривается современное состояние оценки психоэмоционального состояния на основе анализа речи. Затронуты возможные области применения и важность развития алгоритмов классификации эмоций для различных отраслей производства, маркетинга и транспорта. Выделены современные методы обработки речевых сигналов и классификации естественно выраженных психоэмоциональных состояний.

Ключевые слова: анализ речи, оценка психоэмоционального состояния, алгоритмы кластеризации, алгоритмы классификации, эмоции.

За последние несколько лет системы автоматического распознавания речи получили активное распространение в области современных человеко-машинных интерфейсов, а также в областях, где требуется быстрая оценка ситуации на основе поступающих голосовых данных. Это стало возможным во многом благодаря увеличению количества и доступности высокопроизводительных вычислительных комплексов и активному развитию информационно-коммуникационных технологий.

Однако области человеческой деятельности, такие как медицина, маркетинг, безопасность, контроль персонала на опасных производствах и в различных отраслях транспорта, побуждают исследователей искать новые и эффективные инструменты для автоматического распознавания эмоционального состояния человека по его голосу. При решении этой задачи можно автоматически определять уровень стресса и усталости, распознавать депрессивные состояния, предотвращать переутомление и т.д. Кроме того, автоматическое распознавание эмоций также необходимо для того, чтобы поднять используемые системы взаимодействия человека и компьютера на более высокий уровень.

Значительный положительный эффект от внедрения автоматического распознавания эмоций на основе голоса можно ожидать в отраслях, где общение осуществляется преимущественно с помощью речи, без дополнительных визуальных средств, где важно уменьшить возможные источники риска для людей и имущества посредством непрерывного процесса выявления и мониторинга факторов риска. Эти требования относятся, в первую очередь, к авиации [5] и другим видам транспорта.

Следует отметить, что, несмотря на важность выявленной проблемы, общего подхода, в котором обсуждается связь эмоций говорящего с характеристиками голосового сигнала, в настоящее время не существует [6–8]. Этот факт во многом определяет подход, который следует использовать при решении задачи автоматической классификации эмоций по характеристикам голосового акустического сигнала: необходим поиск новых алгоритмов и глубокая оптимизация уже существующих.

Структура современных исследований

В ходе исследования [9–11] выяснилось, что оценка психоэмоционального состояния человека на основе анализа речи заключается в решении трех основных задач: составление базы необходимых речевых сигналов, выделение информативных параметров и построение паттернов, и реализация алгоритма распознавания психоэмоциональных состояний [12]. Схема, представленная на рис. 1, наглядно показывает, как проводятся современные исследования в этой области.

Рис. 1. Структура современных исследований в области оценки психоэмоционального состояния

Подготовка речевых сигналов к последующему анализу

Распознавание эмоций осуществляется преимущественно посредством распознавания по изоб-

ражению, видеопотоку или звуковой информации за счет простоты получения данных для анализа. В процессе исследования путей анализа аудиосигналов были выделены некоторые методы кластеризации и классификации. Анализ с использованием методов кластеризации заключается в поиске закономерностей распределения данных и выделении групп на основе структуры выборки. В случае классификации типы групп и их количество заранее известны.

Предварительная обработка данных перед подачей выбранному алгоритму, используемому при анализе изображений, включает в себя преобразование входных данных в виде звукового сигнала в спектрограмму данного сигнала, затем идет сглаживание спектра сигнала, разбиение спектра на кадры, определение значимых характеристик кадров. После этого идет определение эмоционального состояния с использованием методов машинного обучения, опираясь на значимые характеристики полученных кадров.

Этапы предварительной обработки данных для методов кластеризации и классификации будут совпадать между собой для задач обработки последовательностей символов, текстовой информации. Целью обработки является преобразование данных в набор векторов, которые впоследствии будут подаваться на вход алгоритмам машинного обучения.

Довольно распространенным методом векторизации является модель Bag of Words. В данном методе порядок не имеет значения. Документы представляются в виде матрицы, где каждая строка соответствует отдельному документу, а столбец – определённому слову (символу). Ячейка представляет собой количество вхождений слова (символа) в документ. Далее документы, представленные в виде векторов своих признаков (атрибутов), используются для обучения. Недостатком является отсутствие семантических связей между словами (символами) в тексте (последовательности). В случае потребности в зависимостях между символами проблему можно решить за счет использования n-грамм.

Алгоритмы кластеризации

После рассмотрения этапа предварительной обработки данных можно перейти к обзору кластеризации и классификации. В рамках рассмотрения кластеризации были выделены агломеративные алгоритмы. Как правило, агломеративным алгоритмам на вход подается конечное множество элементов с определенным на нем индексом различия, иными словами, расстоянием. Принцип работы данных алгоритмов можно описать следующим образом: на первом шаге каждый элемент из входного множества представляет собой отдельный кластер, а на последующих шагах происходит процесс итеративного объединения кластеров. Объединение кластеров осуществляется исходя из выбранного подхода, например, задействовав один из методов на основе связи между центрами кластеров, индивидуальных связей или дисперсионный метод (метод Варда). В случае использования методов на основе индивидуальных связей после включения объекта в кластер все так же учитываются расстояния от этого объекта до других. В методах на основе связи между центрами кластеров после включения объекта рассчитываются и в дальнейшем учитываются в расчетах расстояния от некоторой центральной точки, которая характеризует кластер в целом. Метод Варда выделяется от остальных тем, что происходит объединение не тех кластеров, которые в том или ином понимании максимально близки, а тех, при слиянии которых идет наименьший прирост внутриклассовой дисперсии.

Наиболее простым в использовании и от того довольно распространенным является метод k-ближайших соседей. Используется он для распределения конечного числа наблюдений по конечному числу кластеров таким образом, чтобы каждое наблюдение относилось только к одному кластеру, к центру которого оно в наибольшей степени приближено. Под наблюдениями в задаче определения эмоций по речевому сигналу являются точки на спектрограмме. В недостатках можно выделить невысокую точность.

Для момента остановки агломеративного алгоритма кластеризации имеет значение определение оптимального количества кластеров, выражающегося в выборе конкретного шага или промежутка шагов.

Алгоритмы классификации

При решении задачи классификации часто присутствует сложность в выборе методов, поскольку их количество довольно велико, и в то же время осуществить проверку качества работы каждого отдельного метода не представляется возможным. Наиболее часто используемыми и популярными методами можно считать деревья решений, рекуррентные нейронные сети, сверточные нейронные сети, ансамбли алгоритмов и др. Будут рассмотрены некоторые из методов.

Дерево решений является древовидной структурой, состоящей из узлов, соединенных с помощью ребер. Самый верхний узел (корневой) учится разделять на основе значения атрибута. Дерево разделяется рекурсивным способом, т.е. данные разделяются от корневого до некоторого конечного узла, содержащего целевую переменную, определяющую класс. Деревья решений можно объединить в ансамбль, образуя алгоритм, именуемый случайным лесом (Random Forest). Обучение алгоритма происходит путем построения деревьев решений на случайно выбранных параметрах обучающей выборки. Целевой класс для нового объекта определяется большинством голосов построенных деревьев. Недостатком является долгое построение модели, а так как деревья получаются большими, то и сама модель получается громоздкой.

Сверточные нейронные сети – тип нейронных сетей, которые в своем скрытом слое располагают разные фильтры и области, реагирующие на конкретную особенность входного сигнала. Данный вид нейронных сетей используется при анализе изобра-

жений, например при анализе спектрограмм речевых сигналов в задаче определения психоэмоционального состояния на основе аудиоданных. Дарио Бертеро и Паскаль Фанг [1] в исследовании методов для определения эмоций представили сверточную нейронную сеть, обладающую способностью обнаруживать эмоции радости, грусти и гнева с точностью, превышающей 66%. Нейронная сеть концентрировалась на основных частотах, наиболее соответствующих эмоциям.

В задачах анализа последовательностей присутствуют проблемы, связанные с тем, что последовательности могут различаться по длине, могут потребовать от модели изучения долгосрочного контекста или зависимостей. Для анализа последовательностей были разработаны рекуррентные нейронные сети. Рекуррентная нейронная сеть – тип нейронной сети, в которой выходные данные предыдущих шагов подаются как входные данные для текущего шага, таким образом, запоминается некоторая информация о последовательности, данный алгоритм может быть полезен в случае, когда время имеет весомое значение, например, для задач обработки речи, музыкальных композиций.

Одной из разновидностей рекуррентных нейронный сетей является LSTM, предназначенная для решения проблемы долгосрочной зависимости. Обычно таким нейронным сетям на вход подается информация в виде текстов, аудио- или видеосигналов. В последние годы сети LSTM оказались в центре внимания многих приложений, связанных с событиями временных рядов. К недостаткам можно отнести немалые затраты ресурсов и времени на подготовку и обучение, то есть неэффективность с аппаратной точки зрения.

Заключение

Данная область нуждается в углубленной изучении и разработке. На данный момент существует ряд нерешённых проблем. Таких как: создание базы данных речевых сигналов, полученных в реальных условиях; не установлена четкая взаимосвязь вида эмоционального состояния и объективного набора подходящих параметров речевых сигналов; оптимизация задач регистрации естественной эмоциональной речи, определения подходящего набора параметров и классификации эмоций в реальном времени для системы предоставления оценки.

Литература

1. Bertero D., Fung P. A first look into a Convolutional Neural Network for speech emotion detection // IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). – 2017. – PP. 5115–5119.

2. Богданова Д.Р., Акушев А.Т. // Распознавание эмоций по речевому сигналу // E-Scio. – 2021. – С. 416–425.

3. Belanger D., Maxwell L. Bileschi. Using Deep Learning to Annotate the Protein Universe // Nature Biotechnology. – 2022. – PP.932–937.

4. Modern hierarchical, agglomerative clustering algorithms [Электронный ресурс]. – Режим доступа: https://arxiv.org/abs/1109.2378, свободный (дата обращения: 12.09.2022).

5. International Civil Aviation Organization 2010 Aviation Safety Management Manual: document 9859AN/474. – 2nd edition (Monreal: ICAO). – 150 p.

6. Huang X, Acero A., Hon H. Spoken Language Processing: A guide to theory, algorithm and system development. – New Jersey: Prentice Hall, 2001. – 1008 p.

7. Markel G., Grey A. Linear prediction of speech. – Berlin: Springer, 1980. – 620 p.

8. Gerhard D. Pitch Extraction and Fundamental Frequency: History and Current Technique // Technical Report. – 2003. – 22 p.

9. Schuller B.W., Batliner A.M. Computational Paralinguistics: Emotion, Affect and Personality in Speech and Language Processing. – New York: Wiley, 2013. – 344 p.

10. Schuller B. Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge / B. Schuller, A. Batliner, S. Steidl, D. Seppi // Speech Communication. – 2011. – PP. 1062–1087.

11. Ververidis D. Emotional speech recognition: Resources, features, and methods / D. Ververidis, C. Kotropoulos // Speech Communication. – 2006. – PP. 1162–1181.

12. Оценка психоэмоционального состояния на основе анализа речевых сигналов: современное состояние, проблемы и перспективы / А.К. Алимурадов, А.Ю. Тычков, П.П. Чураков, А.В. Агейкин // Труды междунар. науч.-техн. конф. «Перспективные информационные технологии» (ПИТ–2018) / под ред. С.А. Прохорова. – Самара: Самар. нац. исслед. ун-т им. акад. С.П. Королева, 2018. – С. 677–68.

13. Нежников Р.И. Методы распознавания эмоций человека по речевому сигналу с использованием машинного обучения // Сб. статей междунар. науч.-практ. конф. «Разработка и применение наукоемких технологий в эпоху глобальных трансформаций», г. Оренбург. – Оренбург: ОМЕГА САЙНС, 2022. – С. 5–9.

14. Распознавание эмоций из речевого сигнала / В.В. Видман, А.Я. Видман, В.М. Саклаков, И.С. Ванюшин, С.В. Аксенов // Сб. статей междунар. науч.-практ. конф. «Молодежь и современные информационные технологии», 3–7 декабря 2018 г., г. Томск. – Томск: Изд-во НИ ТПУ, 2019. – С. 140–141.

15. Alonso P., Saini R., Kovacs G. Hate Speech Detection Using Transformer Ensembles on the HASOC Dataset // 22-nd International Conference, SPECOM. – 2020. – PP. 13–21.

16. Avci U. Speech Emotion Recognition Using Spectrogram Patterns as Features // 22nd International Conference, SPECOM. – 2020. – PP. 57–67.

17. Dvoynikova A., Verkholyak O., Karpov A. Emotion Recognition and Sentiment Analysis of Extemporaneous Speech Transcriptions in Russian // 22nd International Conference, SPECOM. – 2020. – PP. 136–144.

Горбунова Кристина Эдуардовна
Студентка каф. КИБЭВС ТУСУРа
Эл. почта: vyt_21624@mail.ru

Букина София Германовна
Студентка каф. БИС ТУСУРа
Эл. почта: sofiabukina1714@gmail.com

Харченко Сергей Сергеевич
Канд. техн. наук, доцент каф. БИС ТУСУРа
Эл. почта: kss@fb.tusur.ru

УДК 004.932.2

С.С. Харченко, К.А. Качалов

Проблемы распознавания схожих изображений

Рассматриваются проблематика определения схожих изображений в потоке контента информации, основные алгоритмы, которые рекомендуются к использованию на сегодняшний день.
Ключевые слова: SIFT, PHASH, гистограммы, схожесть изображений.

Сопоставление изображений является одним из основополагающих концептов в компьютерном зрении и определении объектов. Изображения одного и того же предмета могут быть сделаны под разным углом, при любом освещении или масштабе. Эти проблемы наряду с окклюзией, нарушением распознавания изображения путем перекрытия искомого объекта другим составляют задачи для сопоставления изображений [1, 2]. Благодаря бурному развитию цифровых технологий возникает вопрос обработки огромного количества информации в цифровом виде, особенно видео и изображений [3, 4]. Современными задачами являются: категоризация новостей, внедрение цифровых водяных знаков [5] или сопоставление объектов с камерами наружного наблюдения [6]. Целью работы является выделение актуальных методов для повышения эффективности распознавания схожих изображений.

Существующие решения

Одним из наиболее продвинутых алгоритмов сопоставления изображений является SIFT [7]. SIFT извлекает характерные для изображения неизменяемые признаки, которые могут надежно соотноситься на различных видах изображений. Данные признаки являются стойкими к масштабированию и повороту изображения, а также к различным способам искажения, добавления шума и окклюзии.

С помощью алгоритма SIFT можно добиться высокой точности работы, но он не так быстр, как перцептивный хэш и гистограмма [8]. SIFT – это масштабно-инвариантная трансформация признаков (Scale Invariant Feature Transform). Алгоритм был представлен в 2004 г. Дэвидом Лоу, профессором Университета Британской Колумбии [9]. Основной смысл алгоритма заключается в замене изображения некой моделью, которая представляет собой набор ключевых точек [10]. В качестве ключевой точки используется точка объекта, которую с большей вероятностью можно найти на другом изображении этого объекта [11]. Выявление ключевых точек изображения осуществляется с помощью детектора. С помощью него можно обеспечить нахождение одних и тех же точек при изменении изображения.

Еще одним подходом для решения проблемы распознавания схожих изображений является хеширование. Хеширование изображения используется для преобразования изображения в буквенно-цифровую строку. В качестве примера можно привести снятие отпечатков пальцев со снимков или фотографий.

Хеширование имеет два основных назначения, которые используются в задачах сопоставлении изображений: оно позволяет обнаруживать идентичные или визуально похожие изображения и позволяет сохранять отпечаток изображения.

Также этот метод может использоваться для выявления нарушений в авторских правах, например, при обнаружении идентичных хэшей изображений владельца на других веб-сайтах [12].

Хэш-алгоритм – это класс функций, которые могут генерировать хеш-строки. Их возможно сравнить между собой в процессе работы. Обработка одного и того же изображения с помощью такого алгоритма будет возвращать одно и то же значение хэша. Однако его отличие от криптографических хэш-функций заключается в том, что при малейших изменениях изображения значение хэша не меняется абсолютно и остается близким к первоначальному. Первая и самая популярная реализация алгоритма перцептивного хэширования для изображений называется pHash.

Для определения перцептивного хэша изображения по среднему используется следующий принцип [13, 14] (рис. 1):

1. Изображение уменьшают в размере до разрешения 8 на 8 пикселей, что дает 64 значения.

2. В результате упрощается цветовая гамма изображения, оно становится серым.

3. Затем, в соответствии с этим, рассчитывается среднее значение для 64 цветов.

4. Каждая из четырех картинок сопоставляется с ее средним значением. В случае если оно больше, то сохраняем единицу, в случае если меньше – единица.

5. Результатом является 64-битное значение, которое можно представить в виде числа шестнадцатеричной системы исчисления.

В данном случае критерием оценивания схожести будет являться расстояние Хэмминга между двумя изображениями [15]. Чем оно меньше, тем изображения, соответственно, более схожи между собой.

Геометрический анализ является самым надежным способом оценить идентичность объектов. Однако мы можем распознать стул и на черно-белом изображении. Нам не нужно никаких цветов или текстуры. Хотя это не всегда так. В природе есть много примеров, когда цвет идентифицирует разновидности животных или фруктов, фрукты и минералы. Цветовые обозначения часто используются в товарных знаках или логотипах. Посмотрите на

Google и Facebook, первый идентифицируется как сочетание синего цвета с красным, зеленым и желтым, тогда как второй ассоциируется, в основном, с синим [16].

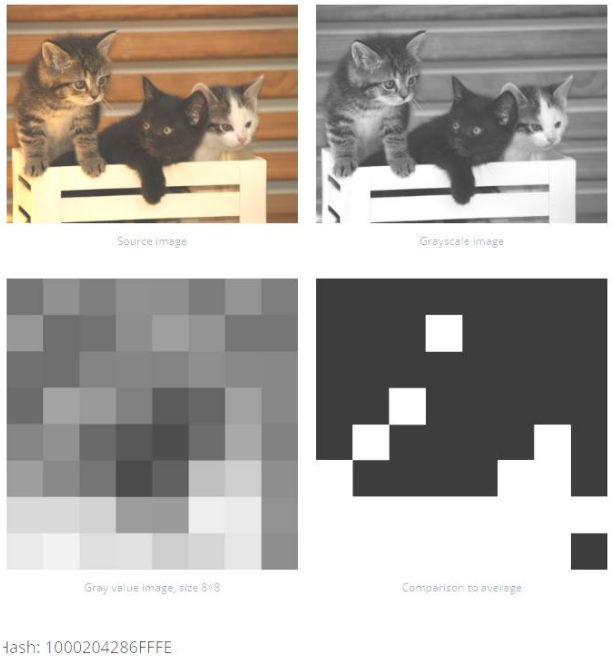


Рис. 1. Пример работы алгоритма на основе pHash

Много различных алгоритмов обрабатывают изображения только в соответствии со своими серыми представлениями. С помощью этих каналов они убирают RGB-каналы, оставляя лишь серые оттенки между белым и черным. Основная причина, по которой изображения в оттенках серого часто используются при обработке вместо непосредственной работы с цветными изображениями, заключается в том, что оттенки серого упрощают алгоритм и снижают требования к вычислениям. Часто информация о цвете не помогает определить важные места объекта или другие особенности, но может являться дополнительным шумом и артефактами [17]. Несомненно, цвет имеет ограниченный эффект для большинства приложений, но добавление ненужных данных может увеличить объем обучающих данных, необходимых для достижения хорошей производительности [18].

В гистограмме отображены распределения интенсивности цветов изображения. Гистограмма изображения показывает количество пикселей в изображении, имеющем определенное значение цвета. Приведем простой пример: обычное серое изображение с интенсивностью пикселей варьируется от 0 до 255. Для создания гистограммы нам нужно только подсчитать количество пикселей, имеющих значение интенсивности 0, а затем 1, затем 2 и т.д. до 255. Рисунок 2 показывает образец изображения размером 5 на 5 пикселей, цветовая интенсивность которого составляет от 0 до 4. После того как создана гистограмма, на первом шаге формируется таблица гистограмм и рассчитывается количество интенсивностей каждого пикселя. Далее можно легко нарисовать гистограмму с помощью таблицы гистограмм, создав ее на основе таблицы гистограмм.

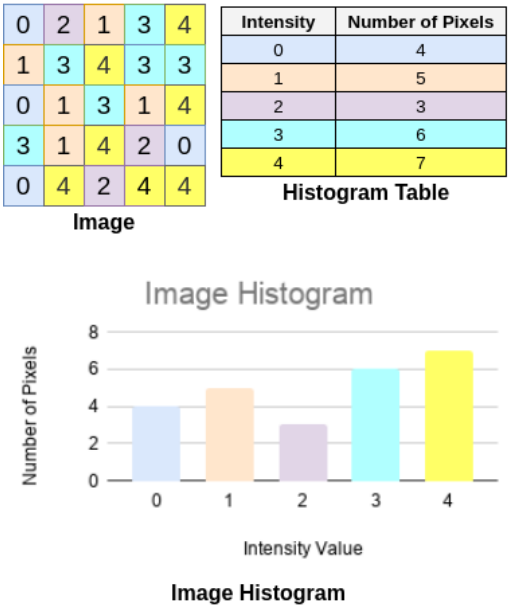


Рис. 2. Пример работы алгоритма на основе гистограмм. Фрагмент изображения (слева сверху) и полученный дескриптор (справа сверху)

Заключение

Область обработки и сравнения изображений нуждается в углубленном изучении и разработке. На данный момент существует ряд нерешённых проблем, таких как отсутствие решений, позволяющих определять схожие изображения в реальном времени для системы предоставления оценки, отсутствие датасетов, позволяющих проводить моделирование и тестирование разрабатываемых алгоритмов, формальное представление какие изображения считать схожими, недостаточная точность определения схожести изображений.

Литература

1. Сопоставление изображений [Электронный ресурс]: сайт Стэнфордского университета. – URL: https://ai.stanford.edu/~syyeung/cvweb/tutorial2.html (дата обращения: 14.03.2022).

2. Качалов К.А. Распознавание схожести изображений в новостном контенте / К.Э. Горбунова, К.А. Качалов // Сборник науч. тр. XVIII Междунар. конф. студентов, аспирантов и молодых ученых. – Томск, 2021. – С. 29–31.

3. Иваненко В.Г., Ушаков Н.В. Защита изображений формата JPEG при помощи цифровых водяных знаков // IT Security. – 2018. – Т. 25, № 2. – С. 106–113.

4. Толеген М.О. Разработка алгоритма построения панорам // Молодежь и современные информационные технологии: сборник тр. XIV Междунар. науч.-практ. конф. студентов, аспирантов и молодых ученых, г. Томск, 7–11 ноября 2016 г.: в 2 т. –Томск: Изд-во ТПУ, 2016. – Т. 2. – С. 191–192.

5. Земцов А.Н., Аль-Макреби И.М. Исследование устойчивости цифровых водяных знаков-логотипов, внедряемых в статические изображения // Инженерный вестник Дона. – 2015. – № 2, ч. 2. – С. 30.

6. Обнаружение и распознавание объектов с камеры [Электронный ресурс]: блог Алексея Буркова. – URL: https://habr.com/ru/post/414991/ (дата обращения: 16.03.2022).

7. Meng Yu Implementing the Scale Invariant Feature Transform (SIFT) Method [Электронный ресурс]: сайт колледжа информационных наук и технологии университета штата Пенсильвания. – URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.102.180&rep=rep1&type=pdf (дата обращения: 21.04.2022).

8. Трубаков А.О., Полякова М.С. Исследование качества работы алгоритмов SIFT и PHASH для задачи поиска нечетких дубликатов изображений // Вестник образовательного консорциума Среднерусского университета. Информационные технологии. – Брянск, 2019. – С. 29–35.

9. Lowe D.G. Distinctive Image Features from Scale-Invariant Keypoints // International Journal of Computer Vision. – 2004. – Vol. 60. – PP. 91–110.

10. Построение SIFT-дескрипторов и задача сопоставления изображений [Электронный ресурс]: портал коллективного блога Хабрахабр. – URL: https://habr.com/ru/post/106302/ (дата обращения: 02.04.2022).

11. Brown M., Winder S., Szeliski R. Multi-image matching using multi-scale oriented patches // Conference on Computer Vision and Pattern Recognition. – 2005. – PP. 510–517.

12. How to use identify visually similar images using hashing [Электронный ресурс]: портал Practical Data Science. – URL: https://practicaldatascience.co.uk/data-science/how-to-use-image-hashing-to-identify-visually-similar-or-duplicate-images (дата обращения: 31.03.2022).

13. Monga V. Perceptually Based Methods for Robust Image Hashing. – PhD thesis, University of Texas, 2005.

14. Как работает перцептивный хэш [Электронный ресурс]: – блог «Секретные записки WEB-программиста». – URL: http://z-i-v.ru/articles/image-hash.htm (дата обращения: 30.03.2022).

15. Christoph Z. Implementation and Benchmarking of Perceptual Image Hash Functions. – PgD thesis, University of Applied Sciences Hagenberg, 2010.

16. Блейхут Р. Теория и практика кодов, контролирующих ошибки. – М.: Мир, 1986. – 576 с.

17. The Simplest Classifier: Histogram Comparison [Электронный ресурс]: сайт блога о машинном обучении Mpatacchiola. – URL: https://mpatacchiola.github.io/blog/2016/11/12/the-simplest-classifier-histogram-intersection.html (дата обращения: 19.04.2022).

18. Grayscale Image Collection Notes for Image Matching [Электронный ресурс]: блог Анкита Дхавана. – URL: https://www.quora.com/Why-do-we-use-grayscale-images-in-most-of-the-image-detection-scripting-in-Python (дата обращения: 11.04.2022).

19. Kanan C., Cottrell G.W. Color-to-Grayscale: Does the Method Matter in Image Recognition // PLoS ONE. – 2012. – Vol. 7(1).

Харченко Сергей Сергеевич
Кан. техн. наук, доцент каф. БИС ТУСУРа
Эл. почта: kss@fb.tusur.ru

Качалов Кирилл Андреевич
Продукт-менеджер ООО «Спутник»
Эл. почта: kachalov@sputnikfund.ru

УДК 004.85

Р.Е. Коломников, К.С. Сарин

Программный комплекс для аутентификации пользователя по динамической рукописной подписи с помощью методов машинного обучения

Представлен программный комплекс для аутентификации пользователя по динамической рукописной подписи. Комплекс извлекает высокоуровневые признаки из сигналов рукописной подписи, проводит выбор информативного подмножества признаков на основе генетического алгоритма и аутентификацию пользователя с помощью алгоритмов классификации машинного обучения. Проведена оценка точности результатов работы комплекса.

Ключевые слова: динамическая подпись, аутентификация, генетический алгоритм, машинное обучение.

Рукописная подпись остается одним из самых распространенных юридически значимых подтверждений личности на документе [1, 2]. Установка подлинности субъекта на основании подписи является актуальной задачей, поскольку средства проверки до сих пор работают в несовершенстве и атаки, такие как подделки, время от времени распознаются как оригинальные подписи [3–6]. Таким образом, актуальна необходимость разработки и реализации программного комплекса, который позволил бы проводить аутентификацию пользователя по рукописной подписи.

Целью настоящей работы является разработка и реализация программного комплекса, позволяющего проводить аутентификацию пользователя по рукописной подписи с помощью методов машинного обучения. Для достижения данной цели было выделено несколько задач:

– разработать архитектуру системы аутентификации;

– провести предобработку сигналов рукописной подписи и извлечь высокоуровневые признаки;

– отобрать из извлеченных признаков подмножество информативных;

– сформировать таблицу наблюдений для создания классифицирующих моделей машинного обучения, осуществляющих аутентификацию;

– построить классифицирующие модели машинного обучения;

– оценить точности созданных классифицирующих моделей.

Архитектура системы аутентификации

Архитектура системы аутентификации представлена на рис. 1. На вход модели подаются две подписи: оригинальная и проверяемая в качестве снятых сигналов. Далее необходимо для каждой подписи вычислить вектор признаков, характеризующий подписи. Для подлинности проверямой подписи блоку принятия решения необходима предобработка признаков путем вычисления абсолютной разницы между соответствующими признаками оригинальной и проверяемой подписи. В результате на основе метода опорных векторов [7–9], деревьев решений или нейронной сети глубокого обучения определяется решение о подлинности подписи. В случае если подпись оригинальна, предоставить доступ, иначе – в доступе отказать).

Предобработка сигналов подписи

При съеме рукописных подписей с планшета сигналы могут содержать разрывы, также ввиду того, что характеристика устройства составляет 200 Гц без интерполяции, что введет к проблеме неравномерного количества отчетов между съемами [10, 11]. Для решения этой проблемы подпись предобрабатывалась с помощью центрирования, масштабирования и интерполирования с помощью кубических сплайнов с количеством отчетов 2048.

Извлечение признаков

Полагаясь на результат работы [12], был сформирован перечень из 113 высокоуровневых признаков, среди которых среднее значение скорости нанесения подписи, максимальное ускорение и рывок, количество отрывов кончика пера от экрана планшета во время нанесения подписи, максимальное расстояние между точками подписи и т.д.

Отбор признаков

Для отбора информативно значимых признаков использовался метаэвристический генетической алгоритм. Чтобы применить его для отбора признаков, необходимо представить кодировку хромосомы следующим образом – значение 0 говорит об отсутствии соответствующего позиции гена признака, а 1 – о присутствии. Помимо отбора признаков, генетический алгоритм применялся также для нахождения гиперпараметров нейронной сети, а именно для нахождения количества слоев сети, количества нейронов на каждом слое и коэффициента скорости обучения. Таким образом, искомая хромосома включает в себя гены, которые отвечают за наличие признаков, и гены, которые содержат значения гиперпараметров нейронной сети. Графическое изображение структуры хромосомы представлено на рис. 2.

Генетический алгоритм был запущен со следующими параметрами: размер популяции – 50, количество поколений – 500, доля новых индивидов в поколении – 0,8, вероятность мутации – 0,05.

Из 113 признаков было отобрано 99. Невыбранными оказались такие признаки, как дисперсия скорости по координате Y, среднеквадратичное отклонение рывка, отношение средней скорости к максимальной скорости, средняя скорость давления, среднее значение рывка по давлению и др.

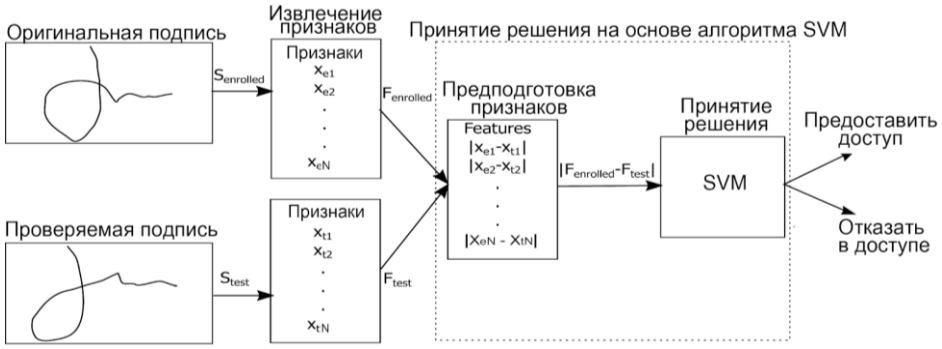


Рис. 1. Архитектура программного комплекса аутентификации

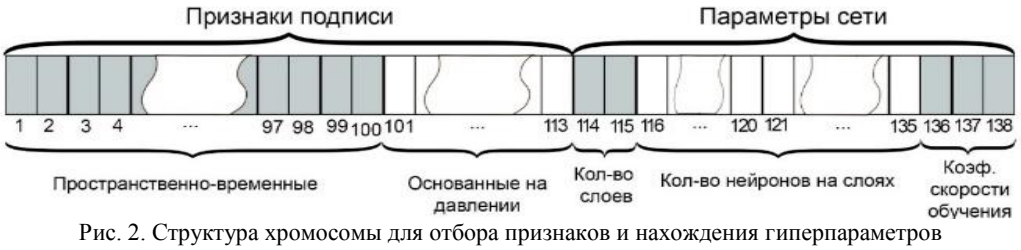


Рис. 2. Структура хромосомы для отбора признаков и нахождения гиперпараметров

Формирование таблицы наблюдений

Для формирования таблицы наблюдений использовалась как собственная база данных, включающая в себя по 30 оригинальных и 30 поддельных подписей для 45 пользователей, так и общедоступная база сигналов DeepSignDB [13], содержащая по 25 оригинальных и 25 поддельных подписей для каждого из 230 пользователей.

Таким образом, был сформирован обучающий комплекс, состоящий из 15 000 подписей в соотношении 2:1:1 (7 500 экземпляров (оригинал:оригинал), 3250 (оригинал:квалифицированная подделка), 3 250 (оригинал:неквалифицированная подделка)).

В результате на основе текущей выборки была создана таблица наблюдений, представляющая собой файл, состоящий из 15 000 и 227 столбцов. За каждой строкой закрепляется одна пара подписей, где в первых 113 столбцах находятся признаки оригинальной подписи, в 114–226 – признаки проверяемой подписи, столбец 227 содержит метку класса, где 0 означает то, что проверяемая подпись оказалась оригинальной, а 1 – если подпись оказалась поддельной.

Программный комплекс

Для работы с комплексом использовался планшет Wacom STU-540. Основными типами сигналов, которые он позволяет снимать, являются координаты (x, y), давление, оказываемое на поверхность, наличие прикосновения, а также выход пера за пределы дисплея.

На рис. 3 представлена модель системы аутентификации согласно методологии функционального моделирования IDEF0.

Данная модель показывает черный ящик системы аутентификации, из которой видно, какие нужны входные параметры, алгоритмы и исполнители. Конечным результатом аутентификации будет сообщение о том, подтвердил ли свою личность субъект или нет.

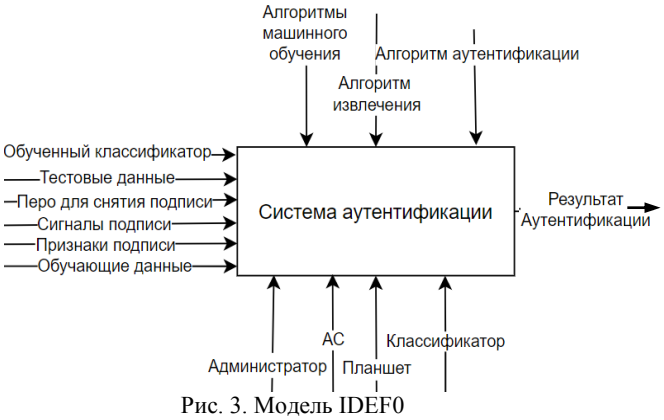


Рис. 3. Модель IDEF0

Далее необходимо заглянуть в модель, а именно сделать декомпозицию черного ящика (рис. 4).

Процесс декомпозиции черного ящика системы аутентификации, исходя из рис. 4, состоит из 4 этапов:

1. Для начала необходимо извлечь подписи у субъектов с помощью планшета и автоматизированной системы. На выходе получаются экземпляры рукописных подписей.

2. Далее идет формирование набора обучающих данных из снятых у субъектов экземпляров рукописных подписей. На выходе системы получается готовый набор, сохраненный в файл с расширением csv.

3. После полученный набор данных необходим для обучения классификаторов, с помощью определенных алгоритмов и генетического алгоритма, позволяющего найти более информативно значимые признаки. На выходе получается обученный классификатор, необходимый для проведения процедуры аутентификации.

4. Для проведения аутентификации необходимы, подаваемая на вход рукописная подпись и непосредственно модель классификации (метод опорных векторов, деревья решения или нейронная сеть). В результате система выдаст сообщение о том, прошел ли субъект аутентификацию.

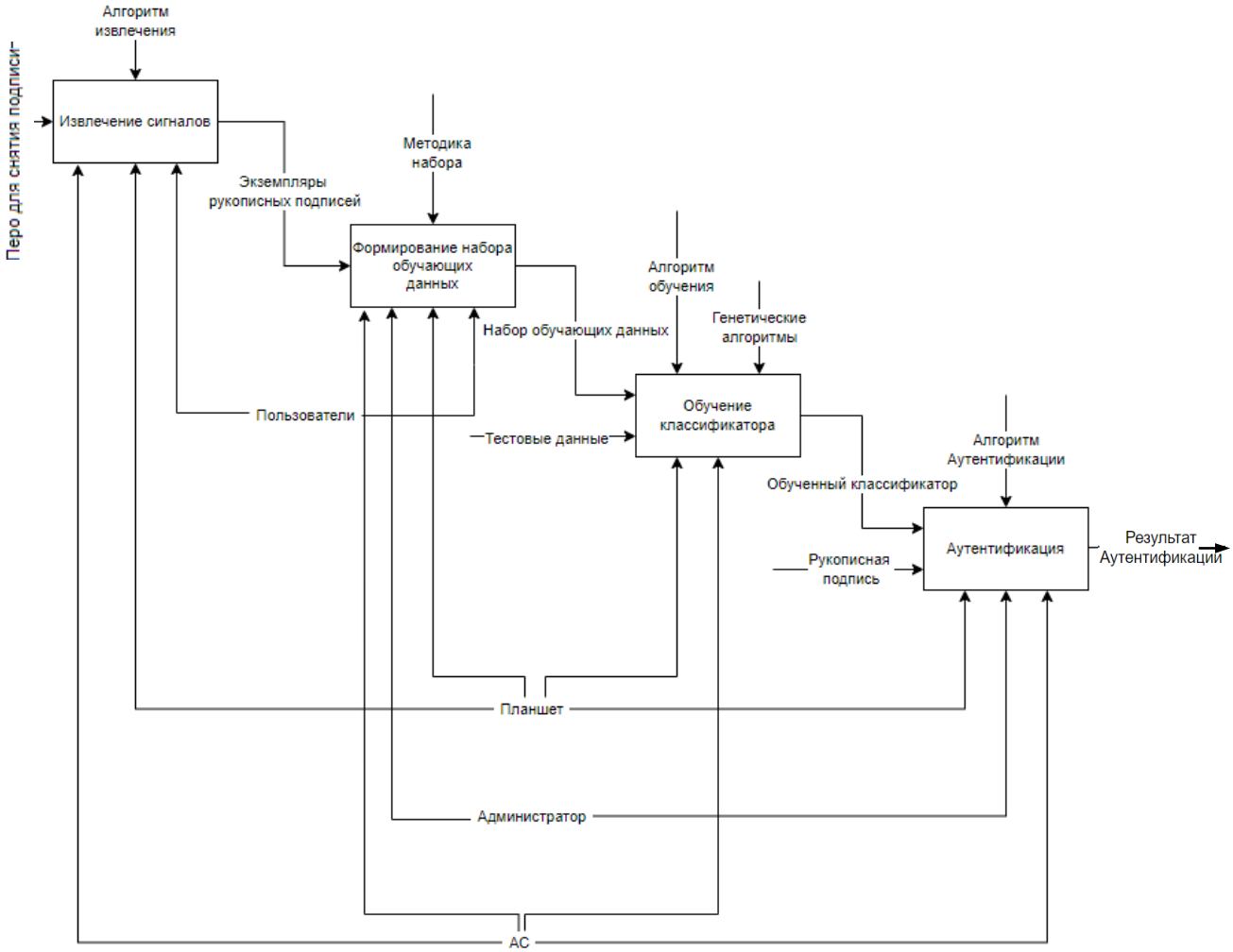


Рис. 4. Декомпозиция системы аутентификации

На рис. 5 представлен фрагмент текстового файла с сигналами подписи. Первые 2 столбца содержат координаты (x, y), далее идет время, давление, признак соприкосновения пера с поверхностью планшета и признак выхода пера за пределы дисплея.

G_R.E. Kolomnikov3_1.txt – Блокнот

Файл Правка Формат Вид Справка

3982	1721	46466	198	1	1
3982	1721	46471	226	1	1
3982	1721	46476	261	1	1
3982	1721	46481	300	1	1
3964	1736	46486	326	1	1
3945	1757	46491	357	1	1
3918	1796	46496	379	1	1
3883	1849	46501	397	1	1
3843	1918	46506	414	1	1
3799	2000	46511	430	1	1
3761	2075	46516	446	1	1
3727	2151	46521	460	1	1
3694	2231	46526	472	1	1
3661	2317	46531	483	1	1
3626	2408	46536	494	1	1
3591	2505	46541	501	1	1
3554	2607	46546	508	1	1
3518	2712	46551	512	1	1
3484	2816	46556	512	1	1
3454	2915	46561	511	1	1

Рис. 5. Текстовый файл с сигналами подписи

В программном комплексе есть возможность построения обучающих данных либо из имеющихся подписей, снятых с планшета, либо из подписей сторонней базы сигналов.

Оценка точности аутентификации

Тестирование классифицированных моделей, построенных как с помощью отобранных признаков, так и без них проводилось на собственной базе данных. Точность аутентификации с отбором информативных признаков оказалась выше. В качестве классифицирующих моделей использовались метод опорных векторов (SVM), деревья решений (DT) и нейронная сеть – многослойный персептрон (NN). Для тестирования использовался метод десятикратной кроссвалидации. В качестве оценок точности аутентификации выступали: точность классификации (ACC), ошибка первого рода (FRR), ошибка второго рода (FAR). Результат тестирования представлен в таблице.

Точность аутентификации

Классификатор	ACC, %	FRR, %	FAR, %
SVM	93,55	17,17	3,68
DT	84,69	30,15	11,59
NN	90,13	22,12	6,61

Заключение

Представлен программный комплекс аутентификации пользователя по динамической рукописной подписи с помощью методов машинного обучения. Комплекс включает в себя возможности сбора сигналов рукописных данных, извлечения высокоуровневых признаков, создания таблицы наблюдений, построения классифицирующих моделей аутентификации на основе методов машинного обучения, осуществления процедуры аутентификации. В качестве классифицирующих моделей использовались метод опорных векторов, деревья решений и нейронные сети.

Создана собственная база сигналов рукописных подписей, содержащая как оригинальные подписи, так и квалифицированные подделки. В сборе сигналов принимали участие 45 человек. На основе собранных сигналов создана таблица наблюдений для построения классифицирующих моделей.

Проведена оценка точности аутентификации. По результатам тестирования лучшие результаты показал классификатор на основе метода опорных векторов с результатом точности классификации 93,55%.

Работа поддержана Российским научным фондом (проект № 22-21-00021).

Литература

1. Ходашинский И.А. Аутентификация пользователя по динамике подписи на основе нечеткого классификатора / И.А. Ходашинский, Е.Ю. Костюченко, К.С. Сарин, А.Е. Анфилофьев, М.Б. Бардамова, С.С. Самсонов, И.В. Филимоненко // Компьютерная оптика. – 2018. – Т. 42, № 4. – С. 657–666.

2. Сарин К.С. Метод баггинга и отбор признаков в построении нечетких классификаторов для распознавания рукописной подписи / К.С. Сарин, И.А. Ходашинский // Компьютерная оптика. – 2019. – Т. 43, № 5. – С. 833–845.

3. Encyclopedia of Biometrics / ed. S.Z. Li, A.K. Jain. – Boston: Springer, 2015. – 1626 p.

4. Vorugunti C.S. OSVFuseNet: Online Signature Verification by feature fusion and depth-wise separable convolution based deep learning / C.S. Vorugunti, V. Pulabaigari, R.K. Gorthi, P. Mukherjee // Neurocomputing. – 2020. – Vol. 409. – PP. 157–172.

5. Song X.Y. Online Signature Verification Based on Stable Features Extracted Dynamically / X.Y. Song, X.H. Xia, F.J. Luan // IEEE Transactions on systems man cybernetics-systems. – 2017. – Vol. 47, No. 10. – PP. 2663–2676.

6. Lai S.X. Recurrent Adaptation Networks for Online Signature Verification / S.X. Lai, L.W. Jin // IEEE Transactions on information forensics and security. – Vol. 14, No. 6. – PP. 1624–1637.

7. Parodi M. Online Signature Verification: Improving Performance through Pre-classification Based on Global Features / M. Parodi, J.C. Gomez // Lecture Notes in Computer Science. – 2013. – Vol. 8158. – PP. 69–76.

8. Jain A. Signature verification using geometrical features and artificial neural network classifier / A. Jain, S.K. Singh, K.P. Singh // Multimedia tools and applications. – 2020. – Vol. 79, No. 27–28. – PP. 19993–20018.

9. Ahrabian K. Usage of autoencoders and Siamese networks for online handwritten signature verification / K. Ahrabian, B. BabaAli // Neural computing and applications. – 2019. – Vol. 31, No. 12. – PP. 9321–9334.

10. Kholmatov A. Identity authentication using improved online signature verification method / A. Kholmatov, B. Yanikoglu // Pattern Recognition Letters. – 2005. – Vol. 26. – PP. 2400–2408.

11. Yang L. Online handwritten signature verification based on the most stable feature and partition / L. Yang, X. Jin, Q. Jiang // Cluster Computing. – 2019. – Vol. 22. – PP. 1691–1701.

12. Fierrez-Aguilar J. An On-line signature verification system based on fusion of local and global information / J. Fierrez-Aguilar, L. Nanni, J. Lopez-Penalba, J. Ortega-Garcia, D. Maltoni // Lecture Notes in Computer Science. – 2005. – Vol. 3546. – PP. 523–532.

13. Tolosana R. DeepSign: Deep On-Line Signature Verification / R. Tolosana, R. Vera-Rodriguez, J. Fierrez, J. Ortega-Garcia // IEEE Transactions on Biometrics, Behavior, and Identity Science. – 2021. – Vol. 3. – PP. 229–239.

Коломников Роман Евгеньевич
Студент каф. КСУП ТУСУРа
Эл. почта: kolomnikov0405@gmail.com

Сарин Константин Сергеевич
Канд. техн. наук, доцент каф. КСУП ТУСУРа
Эл. почта: sarin.konstantin@mail.ru

УДК 004.8

Н.П. Корышев

Построение с помощью алгоритма «китов» нечёткого классификатора вторжений по данным аудита жёсткого диска под управлением операционной системы Linux

Представлены результаты построения классификаторов на основе нечётких правил для обнаружения вторжений по сведениям событий жёсткого диска, находящегося под управлением операционной системы Linux. Настройка его параметров осуществляется с помощью метаэвристического алгоритма «китов». Работоспособность предлагаемого алгоритма построения нечёткого классификатора оценивается с использованием набора Linux_disk из совокупности наборов TON_IoT, полученных с различных конечных устройств интернета вещей.
Ключевые слова: обнаружение вторжений, нечёткий классификатор, метаэвристический алгоритм «китов», набор данных TON-IoT, Linux, обнаружение выбросов.

С распространением интернета вещей (IoT) и промышленных систем интернета вещей (IIoT), количество развернутых устройств постоянно растёт. Обеспечение стабильности компьютерных систем, их защита от киберугроз, достижение их целостности и сохранение их конфиденциальности приобретают всё большую значимость. Устройства IoT не защищены должным образом, а их разработчики часто пренебрегают защитой от атак из-за дополнительных затрат и ограниченных ресурсов интеллектуальных устройств [1].

Увеличение количества атак на системы IoT/IIoT связано с простотой их эксплуатации и возможностью согласованного запуска атак. Промежуточные устройства в этих сетях, как правило, оснащаются операционной системой, которая управляет их аппаратным обеспечением и предоставляет услуги программам. Поэтому с появлением IoT популярность лёгких встраиваемых операционных систем (с разделением времени и реального времени), созданных на базе Linux, резко возросла. Таким образом, исследование киберугроз для систем Linux и разработка методов повышения их безопасности являются важной задачей.

Цель работы – реализация на основе метаэвристического алгоритма «китов» алгоритмов и программных средств для построения нечёткого классификатора, пригодного для решения в том числе задачи обнаружения вторжений в системах под управлением Linux.

Системы обнаружения вторжения в сетях IoT

Из-за многочисленных угроз, которые могут быть нацелены на компьютеры, было разработано множество механизмов для обеспечения безопасности [2]. Одни из механизмов, которые получили большое развитие в последние годы, – это системы обнаружения вторжений (СОВ).

В зависимости от того, в какой части сети применяется СОВ, их можно разделить на две группы: СОВ для сети и СОВ для хоста. Первые фокусируются на сетевом аспекте системы, отслеживая и анализируя входящий и исходящий трафик из стратегических местоположений, где он установлен. Хостовые СОВ устанавливаются на отдельных хостах (компьютерах, узлах) в сети и отслеживают внутреннее состояние устройства. Поскольку данная работа посвящена построению СОВ для обнаружения вторжений по внутренним событиям, обрабатываемым операционной системой, такая СОВ будет являться хостовой.

В зависимости от методологии, используемой для обнаружения атаки, СОВ подразделяются на системы на основе базы данных сигнатур, на основе обнаружения аномалий аномалий или на гибриды того и другого. СОВ на основе сигнатур используют базу данных известных атак и шаблонов, которые их идентифицируют, в то время как СОВ на основе аномалий способны распознавать нормальное поведение законного пользователя, помечая любые отклонения как атаки.

Нечёткий классификатор. Метаэвристический алгоритм «китов»

Работоспособность СОВ может быть увеличена за счёт использования применения методов из области искусственного и вычислительного интеллекта [3]. Одним из способов решения задачи классификации из области вычислительного интеллекта является нечёткий классификатор, который использует нечёткие множества и нечёткую логику для представления знаний о решаемой проблеме. Нечёткая логика позволяет одному объекту принадлежать одновременно к разным классам с некоторой степенью принадлежности. Граница между нормальным и аномальным поведением пользователей в сети и компьютерной системе является размытой, поэтому аппарат нечёткой логики становится отличным выбором для проектирования СОВ.

Главным достоинством нечёткого классификатора является высокая интерпретируемость созданных правил, что повышает надежность классификации и облегчает понимание структуры СОВ, построенной на его основе [4].

СОВ на основе нечёткого классификатора будет относиться к гибридному типу. Модель классификатора будет строиться на основе таблицы наблюдений – базы данных известных сигнатур соединений, и нечёткие правила будут соответствовать известным классам, но благодаря высокой интерпретируемости

правил она может быть способна обнаруживать неизвестные атаки без обновления сведений о сигнатурах вторжений.

Обычно процесс создания нечёткого классификатора включает в себя выполнение 3 этапов. Первый этап заключается в генерации базы нечётких правил, которая может выполняться либо по экстремумам классов, либо с помощью алгоритма кластеризации. Второй этап – в отборе информативных признаков для классификатора, выполняющемся по схеме «фильтра» или «обёртки». Третий этап – в оптимизации параметров в составе нечётких правил классификатора. Алгоритмы решения этих задач могут быть разработаны на основе алгоритмов оптимизации, в том числе и метаэвристик, поскольку решение этих задач можно свести к решению задачи оптимизации некоторой целевой функции. В данной работе для реализации алгоритма построения нечеткого классификатора используется метаэвристический алгоритм «китов» (WOA) [5].

WOA основан на роевом поведении горбатых китов. Здесь используется два поведенческих механизма, которые обновляют векторы решений на каждой итерации: сужающегося окружения добычи и построения спиралевидной пузырьковой сети для атаки на добычу. Первый механизм представляет собой выполнение этапа диверсификации, второй – этапа интенсификации, в котором решение стремится к оптимуму, двигаясь по спиралевидной траектории [6].

Последовательность выполнения WOA состоит из следующих шагов. Вначале случайным образом инициализируется популяция особей, т.е. решений, и определяется наилучшая из них. Затем на каждой итерации все имеющиеся решения меняются в зависимости от типа поведения: на начальных итерациях используется преимущественно первый тип, на последних – второй; в конце итерации определяется лучшее решение. Результат работы метаэвристики – наилучшее решение, найденное за всё время работы алгоритма, и соответствующее ему значение целевой функции.

Описание использованного набора данных

Работоспособность классификаторов существенно зависит от характеристик используемого набора данных. При этом в области обнаружения вторжений актуальна проблема доступности реальных наборов данных по вторжениям для исследователей, работающих с интеллектуальными системами и алгоритмами машинного обучения. Это затрудняет понимание того, насколько адекватны предлагаемые исследователями «интеллектуальные» решения. Кроме того, большинство из наборов данных, предназначенных для создания СОВ, фокусируются исключительно на системных вызовах [7], игнорируя следы вторжений в памяти (RAM), жёстком диске (HD) и в работе процессора.

Однако исследователи [7] предложили набор данных TON_IoT, разработанный на основе взаимодействующих разнообразных сетевых элементов и систем IoT/IIoT. В лабораторных условиях они попытались сымитировать реальные процессы в сетях IoT/IIoT. Данные в этом наборе собраны из разных источников: данные из компьютерной сети, данные из операционных систем и данные телеметрии различных датчиков. Для создания тестового стенда для сбора набора данных с систем Linux ими была разработана отдельная архитектура сетевого взаимодействия систем Linux и интернета вещей.

Набор данных Linux был создан благодаря выполнению аудита системных событий на машинах с установленным дистрибутивом Ubuntu. Источниками отслеживаемых событий служили оперативная память, выполняемые процессы и жёсткий диск; по этой причине набор данных Linux состоит из трёх поднаборов, соответствующих источнику событий. Однако их можно объединить в один по признаку PID – идентификатору процесса.

Для того чтобы иметь возможность сравнить работоспособность предлагаемого классификатора по обнаружению вторжений в системах Linux с другими, эксперименты выполняются только на наборе Linux_disk, взятом из репозитория [8]. Именно поэтому данная работа посвящена определению вторжений только по записям аудита жёсткого диска, находящегося под управлением Linux-системы.

Экземпляры набора распределены на один класс нормальных соединений (из 100 000 экземпляров) и семь классов вторжений в операционную систему: сканирование (Scanning), отказ в обслуживании (DoS), распределённая атака типа «Отказ в обслуживании» (DDoS), инъекционная атака (Injection), атака перебором паролей (Password), атака «человек посередине» (Man-In-The-Middle, MITM), применение межсайтовых скриптов (XSS). Класс MITM состоит из 112 экземпляров, тогда как остальные классы вторжений включают по 10 000 экземпляров. Таким образом, Linux_disk, как и большинство других реальных наборов данных, содержащих записи о сетевом трафике (данные телеметрии TON_IoT, KDD-Cup-99, UNSW-NB15 и др.), является несбалансированным.

Экземпляры Linux_disk обладают 6 признаками, среди которых идентификатор процесса, который активен в ядре Linux, объёмы считанных / записанных данных, количество данных, которые были записаны, но были изъяты, процент занятости диска и имя процесса, который активен в ядре Linux (не был использован, так как признак номинален и соответствует числовому идентификатору PID).

Эксперимент и полученные результаты

При построении классификаторов в данной работе WOA использовался только для выполнения оптимизации параметров нечётких термов. Генерация базы правил нечёткого классификатора (НК) осуществлялась по экстремумам классов, однако функцией принадлежности оставалась гауссоида.

Также было принято решение не включать в процесс построения этап отбора информативных признаков, поскольку количество признаков относительно мало.

Усреднённые значения точности классификаторов атак на Linux (по жёсткому диску)

Классификаторы	Средняя точность распознавания по классам вторжений, %						
	DDoS	DoS	Injection	Scanning	Password	XSS	MITM
HK+WOA	0,00	**75,56**	**61,55**	3,80	17,20	20,85	43,47
k-NN	**100,00**	53,50	61,03	**84,94**	23,56	**89,44**	**99,10**
LOF	38,60	50,39	53,47	50,72	33,18	57,64	74,10
COF	8,10	68,19	80,86	63,82	32,57	70,56	94,60
aLOCI	9,58	42,43	39,71	34,83	**40,38**	41,99	29,50
INFLO	8,84	42,75	50,04	25,33	38,57	47,17	58,90
LoOP	16,41	42,02	44,04	43,46	33,40	58,51	51,80
CBLOF	**100,00**	32,09	21,07	44,78	35,71	18,15	50,00
CMGOS	**100,00**	50,94	21,17	46,54	33,52	22,19	50,00
LDCOF	**100,00**	32,10	21,09	44,81	35,74	18,18	50,00
RPCA	**100,00**	46,35	23,73	47,83	35,73	25,32	48,02
HBOS	**100,00**	64,57	51,63	66,22	23,57	78,77	66,07

Эксперимент выполнялся по схеме 5-кратной кроссвалидации; количество прогонов равнялось 3. Максимальное количество итераций WOA равнялось 250, количество особей в WOA – 30.

В таблице приводятся средние показания точности распознавания отдельных классов нечётким классификатором, создаваемого с помощью предлагаемого алгоритма. Для сравнения в таблицу включены аналогичные показания других классификаторов: *k*-ближайших соседей (*k*-NN), алгоритмов определения локального уровня выброса (LOF), уровня выброса на основе подключения (COF), локального корреляционного интеграла (aLOCI), влияния выбросов (INFLO), локальной оценки вероятности выброса (LoOP), локального уровня выброса на основе кластеризации (CBLOF), многомерного показателя выбросов по Гауссу на основе кластеризации (CMGOS), уровня выброса на основе локальной плотности на основе кластера (LDCOF), а также алгоритма надёжного анализа основных компонентов (RPCA) и оценки выбросов на основе гистограммы (HBOS). Точности аналогов были получены исследователями в работе [7] с помощью инструмента Rapid Miner.

Заключение

Результаты построения нечёткого классификатора для обнаружения вторжений на наборе Linux_disk из TON_IoT в таблице показывают, что настраиваемые с помощью WOA нечёткие классификаторы могут быть использованы в СОВ для определения легитимности сетевого трафика и выполняемых процессов в Linux-системе, и их работоспособность находится на уровне известных аналогов. Наивысшая по сравнению с аналогами точность классификации была достигнута при распознавании атак «Injection» (61,55%) и «DoS» (75,76%). Кроме того, точность распознавания «нормальных» событий составила 100%; эти сведения не были включены в таблицу, так как эта точность для аналогов не была известна. Однако точность нечёткого классификатора в классах «DDoS», «Scanning» и «Password» оказалась наихудшей (0, 3,8 и 17,2%, соответственно). Среди проверенных классификаторов можно выделить *k*-NN, чьи результаты работы в целом успешнее относительно всех остальных.

В дальнейшем планируется проведение исследований, направленных на улучшение работоспособности нечёткого классификатора на наборах данных TON_IoT. Такими направлениями могут служить дальнейшая модернизация алгоритмов WOA, применение алгоритмов предобработки данных для преодоления несбалансированности данных, а также разработка алгоритмов для обучения нечёткого классификатора онлайн при рассмотрении данных в виде непрерывного потока, так как современные СОВ предполагают мониторинг событий и принятие решений в режиме реального времени.

Литература

1. Iot security vulnerability: A case study of a web camera / Y. Seralathan, T.T. Oh, S. Jadhav, J. Myers, J.P. Jeong, Y.H. Kim, J.N. Kim // 2018 20th International Conference on Advanced Communication Technology (ICACT). – IEEE, 2018. – PP. 172–177.

2. Moustafa N. Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models / N. Moustafa, G. Creech, J. Slay. – Data analytics and decision support for cybersecurity. – Springer, 2018. – PP. 127–156.

3. Cavusoglu U. A new hybrid approach for intrusion detection using machine learning methods // Applied Intelligence. – 2019. – №. 49. – PP. 2735–2761.

4. Bardamova M. Construction of Fuzzy Classifiers by a Brain Storm Optimization Algorithm / M. Bardamova, I. Hodashinsky, M. Svetlakov // Advances in Swarm Intelligence. ICSI 2022. Lecture Notes in Computer Science. – Cham: Springer, 2022. – Vol. 13344.

5. Корышев Н.П. Оптимизация параметров нечёткого классификатора на основе алгоритма «китов» // Сб. избран. статей научной сессии ТУСУРа, Томск, 22–24 мая 2019 г.: в 2 ч. – Томск: В-Спектр, 2019. – Ч. 2. – С. 66–68.

6. Mirjalili S. The Whale Optimization Algorithm / S. Mirjalili, A. Lewis // Advances in Engineering Software. – 2016. – № 95. – PP. 51–67.

7. Moustafa N. Data Analytics-enabled Intrusion Detection: Evaluations of ToN IoT Linux Datasets / N. Moustafa, M. Ahmed, S. Ahmed // 2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). – IEEE, 2020. – PP. 727–735.

8. UNSW Sydney. The TON_IoT Datasets [Электронный ресурс]. – Режим доступа: https://research.unsw.edu.au/projects/toniot-datasets, свободный (дата обращения: 18.09.2022).

Корышев Николай Павлович
Аспирант каф. КСУП ТУСУРа
Эл. почта: koryshev1997@gmail.com

УДК 51-77

Р.О. Остапенко, И.А. Ходашинский

Обнаружение сетевых атак с помощью нечёткого классификатора, построенного с помощью метаэвристического алгоритма «саранчи»

Приведены описание алгоритма построения нечёткого классификатора с использованием метаэвристики, а также результаты, полученные на наборах данных, специализированных на сетевых атаках при проведении тестирования алгоритма.

Ключевые слова: нечёткий классификатор, K-средние, CICIDS2017, UNSW-NB15, алгоритм «саранчи».

Сетевые атаки

С каждым годом объём сетевого трафика возрастает. Данный факт подтверждается увеличением объёма мобильного трафика. По данным компании Statista, каждый год происходит увеличение объёма интернет-трафика мобильных устройств [1]. При анализе сетевого трафика возможно выявить закономерности передачи данных по сети, собрать статистику об использовании веб-приложений, провести мониторинг и исследовать уровень сетевой нагрузки, обнаружить сетевые атаки, определить потенциальные вредоносные программные средства и т.д. [2].

При выполнении анализа трафика возникает следующая проблема: злоумышленник может получить доступ к содержимому пакетов данных, которые передаются по сети. С развитием сетевых технологий анализ сетевых пакетов усложнился, из-за этого многие подходы анализа сетевого трафика стали устаревшими. Поэтому возникает задача фильтрации сетевого трафика, которую можно привести к задаче классификации сетевого трафика.

Целью работы является повышение точности обнаружения сетевых атак нечётким классификатором, построенным с помощью метаэвристического алгоритма «саранчи».

Классификация и нечеткий классификатор

Классификация – важная составляющая научного направления «Машинное обучение». В данной статье под понятием «классификация» понимается процесс построения классификатора – разделение множества объектов (наблюдений) на группы (классы) на основе анализа их признакового описания;

В статье классификация – это обучение с учителем, обучение на помеченных данных ($\mathbf{x}_i$, c_j).

Далее приведена постановка задачи классификации.

Пусть <**X, A, C**> – набор данных,

$\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x}_{|X|}\}$ – непустое конечное множество объектов (наблюдений);

$\mathbf{A} = \{a_1, \ldots, a_n\}$ – непустое конечное множество признаков (атрибутов);

$C = \{c_1, c_2, \ldots, c_m\}$ – непустое конечное множество меток классов;

$c_j \in C$ – значение метки класса для в j-го наблюдения.

На множестве объектов **X** и множестве меток классов C построить алгоритм (функцию или классификатор) c: $\mathbf{X} \rightarrow C$, способный сопоставить метку класса для произвольного объекта из исходного множества; $c = c(\mathbf{a}; \boldsymbol{\theta})$ – метка, соответствующая вектору признаков **a**; $\boldsymbol{\theta}$ – вектор параметров классификатора.

Среди различных типов классификаторов нечеткий классификатор (НК) имеет преимущество в виде возможности интерпретации как полученного результата, так и процесса его получения [3, 4].

Процесс построения НК состоит из трех этапов: формирование базы нечетких правил [5, 6], отбор признаков [7, 8], оптимизация параметров НК [9, 10]. Данные этапы возможно привести к задачам оптимизации, для решения которых возможно использовать метаэвристические алгоритмы.

В [11] описан популяционный алгоритм оптимизации, который вдохновлен поведением роя саранчи и описано его применение для решения задач поиска оптимума сложных математических функций, а также для проектирования оптимальных форм консольной балки и ферм различной конфигурации.

В [12, 13] алгоритм приведён для отбора информативных признаков. Результаты применения алгоритма показали его высокую эффективность в решении задач с аналитически незаданными пространствами поиска.

Алгоритм «саранчи»

Множество личинок саранчи передвигаются подобно катящимся цилиндрам. На своем пути они едят почти всю растительность. Во взрослом возрасте они образуют рой в воздухе. Так саранча может передвигаться на большие расстояния. Основные характеристики роя в фазе личинок – медлительность, короткие движения. С другой стороны, во взрослом состоянии рой характеризуется резкими, дальними движениями.

Алгоритм учитывает обе фазы роя. Все особи пытаются найти зону комфорта (этап интенсификации). Существует сила стремления попасть в зону комфорта и сила отталкивания, что позволяет каждой отдельной особи искать лучшее решение (этап диверсификации). Новая позиция особи определяется на основе её текущего положения, лучшего решения на данный момент и положения всех других особей [11].

Далее представлен собственно алгоритм «саранчи».

Алгоритм «саранчи»

Вход: N – размер популяции, $\max_{iter}$ – максимальное количество итераций;
Выход: **T** – лучшее решение.
1: Случайным образом сгенерировать N решений.
2: Найти лучшее решение **T**.
3: **while** $k < \max_{iter}$ **do**
4: Расчёт коэффициента c по формуле (1);
5: **while** $i < N$ **do**
6: Поиск i-го решения по формуле (2);
7: **end while**;
8: Найти лучшее решение **T** в новой популяции.
9: $k = k + 1$;
10: **end while**

Коэффициент c, который отвечает за соблюдение баланса диверсификация-интенсификация, рассчитывается следующим образом:

$$c = c_{\max} - k\frac{c_{\max} - c_{\min}}{\max_{\text{iter}}}, \quad (1)$$

где $c_{\max}$, $c_{\min}$ – максимальное и минимальное значение коэффициента c, $\max_{\text{iter}}$ – максимальное количество итераций, k – текущая итерация.

$$\mathbf{x}_i = c\cdot\left(\sum_{\substack{j=1\\ j\neq i}}^{N} c\cdot\frac{ub-lb}{2}\cdot s(\rho_{ij})\cdot\frac{\mathbf{x}_j-\mathbf{x}_i}{\rho_{ij}}\right)+\mathbf{T}, \quad (2)$$

где ub, lb – верхняя и нижняя границы поиска; $\rho_{ij} = |\mathbf{x}_i - \mathbf{x}_j|$ – расстояние между i-м и j-м решением; $s(\rho) = f\exp(-\rho/v)$ – сила притяжения; f, v – константы.

Для отбора информативных признаков используется алгоритм «саранчи» и S-образная трансформационная функция, которая представлена ниже.

$$f(X_{ij}) = \begin{cases} 1, \text{ если } r < \dfrac{1}{1+e^{-X_{ij}}}, \\ 0, \text{ иначе,} \end{cases} \quad (3)$$

где r – случайное число непрерывного равномерного распределения из отрезка [0, 1], X_{ij} – j-й признак i-го вектора отбираемых признаков.

Гибрид алгоритмов «саранчи» и «K-средние»

Гибрид алгоритмов «саранчи» и «K-средние» используется для генерации базы правил. Первоначальные координаты кластеров в гибридном алгоритме формируются с помощью алгоритма K-средние. Далее в цикле выполняется одна итерация алгоритма «саранчи», использующего функцию пригодности, и одна итерация алгоритма K-средние; выбирается лучшее решение. Рассчитанные координаты центроидов кластеров используются для формирования базы правил НК.

Популяция состоит из единственного решения, которое представлено в виде матрицы $\mathbf{T} = (\mathbf{q}_1, \ldots, \mathbf{q})$, где $\mathbf{q}_i = (q_{i1}, \ldots, q_{in})$, q_{il} – l-я координата i-го центроида в n-мерном пространстве признаков.

В качестве функции пригодности выбрана суммарная дисперсия F_1 исходя из [14]:

$$F_1 = \sum_{k=1}^{K}\sum_{x_i\in C_k}\|\mathbf{x}_i - \mathbf{c}_k\|^2 \to \min, \; i=\overline{1,N}, \; k=\overline{1,K}, \quad (4)$$

где $\mathbf{x}_i$ – i-й экземпляр таблицы наблюдений; $\mathbf{c}_k$ – координаты центроида k-го кластера; N – количество экземпляров в наборе данных; K – количество кластеров.

Гибридный алгоритм представлен ниже.

Алгоритм кластеризации

Вход: K – количество кластеров,, $\max_{\text{iter}}$ – максимальное количество итераций, $f(\mathbf{X})$ – функция пригодности, **X** – текущее решение;
Выход: **T** – координаты центроидов кластеров.
1: Сгенерировать **T** с помощью алгоритма K-средние.
2: $fit_T = f(\mathbf{T})$.
3: **while** $k < \max_{\text{iter}}$ **do**
4: Формирование $\mathbf{X}_k$ с помощью последовательного выполнения одной итерации алгоритма «саранчи» и одной итерации алгоритма K-средние;
5: $fit_k = f(\mathbf{X}_k)$.
6: **if** $fit_T > fit_k$ **then** $fit_T = fit_k$, $\mathbf{T} = \mathbf{X}_k$.
7: $k = k + 1$;
8: **end while**

Наборы данных

Один из наборов данных для тестирования обнаружения сетевых атак – «UNSW-NB15». Первоначально сетевые пакеты, на которых основан данный набор данных, были собраны в лаборатории Cyber Range университета Нового Южного Уэльса в Канберре [15]. В наборе представлено девять видов атак.

Набор данных «CICIDS2017» подготовлен Канадским институтом кибербезопасности по результатам анализа сетевого трафика изолированной среды, в которой производилось моделирование действия злоумышленников и обычных пользователей [16]. В рамках данной статьи использовался только файл «Thursday-WorkingHours-MorningWebAttacks», который содержит в себе обычный трафик и распространенные сетевые атаки трех видов.

Предобработка наборов данных

Набор данных «UNSW-NB15» содержит экземпляры 10 классов: fuzzers, analysis, backdoors, DoS, exploits, generic, reconnaissance, shellcode, worms и нормальный трафик. Помимо этого, присутствует вторая классификация: 1 – присутствует сетевая атака, 0 – нормальное соединение. В эксперименте использовалась именно вторая классификация, так как важно повышение точности обнаружения сетевых атак, а не распознавание конкретной атаки.

Проведена предобработка набора данных «CICIDS2017». Метки класса заменены на 1 – если есть атака, и на 0 – если атаки нет. Неопределённые значения (NaN) и бесконечные значения заменены на –1. Удалён дважды присутствующий признак «FWD Header Length».

Исходный набор не является сбалансированным, так как доля нормального трафика составляет 98,73%, а класс сетевой атаки имеет долю 1,27%. Использовался метод undersampling для устранения несбалансированности данных. В результате достигнуто соотношение экземпляров классов 70 / 30.

Эксперимент

Проведён эксперимент с использованием НК, построенного на основе алгоритма «саранчи», установлены следующие параметры: популяция 40, количество итераций генерации базы правил 150, количество итераций отбора информативных признаков 150, количество итерация алгоритма оптимизации нечётких термов 150, количество правил 20, количество повторений эксперимента 5. Средняя точность классификации на обучающей и тестовой выборках, количество отобранных информативных признаков представлены в табл. 1.

Таблица 1

Средние точности и среднее количество признаков на наборе данных «UNSW-NB15»

	Значение
Обучение до, %	80,02
Тест до, %	86,54
Обучение после, %	82,42
Тест после, %	84,15
Отобрано признаков, количество	25,5

Из табл. 1 видно, что после применения алгоритма «саранчи» точность увеличилась.

Для оценки пригодности построенного классификатора было проведено сравнение с другими моделями классификации, обученными на наборе данных «UNSW-NB15» для обнаружения сетевых атак.

Результаты НК (GS-F1) сопоставлены со следующими моделями [17]:

– DNN – глубокая нейронная сеть;

– GBT – дерево градиентного бустинга;

– RF – случайный лес;

– NB – байесовский классификатор.

Результаты, полученные различными классификаторами на тестовой выборке, приведены в табл. 2.

Таблица 2

Результаты, полученные на наборе данных «UNSW-NB15»

Алгоритм	Точность, %
DNN	99,16
GBT	97,83
RF	97,49
GS-F1	84,15
NB	74,19

Из табл. 2 видно, что наилучший результат показала модель DNN. Худший результат показала модель NB. НК, построенный с помощью алгоритма «саранчи», показал четвертый результат.

Проведён эксперимент при использовании набора данных «CICIDS2017». Параметры для построения НК: популяция 40, количество итераций генерации базы правил 150, количество итераций отбора информативных признаков 150, количество итерация алгоритма оптимизации нечётких термов 150, количество правил 10, количество повторений эксперимента 5. Средняя точность классификации на обучающей и тестовой выборках, количество отобранных информативных признаков представлены в табл. 3.

Таблица 3

Средние точности и среднее количество признаков на наборе данных «CICIDS2017»

	Значение
Обучение до, %	97,11
Тест до, %	97,52
Обучение после, %	99,11
Тест после, %	97,5
Отобрано признаков, количество	28,8

Из табл. 3 видно, что после применения алгоритма «саранчи» точность увеличилась.

Для оценки пригодности классификатора, построенного нечеткого классификатора на основе метаэвристического алгоритма «саранчи» было проведено сравнение с другими моделями классификации, обученных на наборе данных «CICIDS2017» для обнаружения сетевых атак.

Произведено сравнение с ранее рассмотренными моделями [18, 19]. Результаты, полученные различными классификаторами на тестовой выборке, приведены в табл. 4.

Таблица 4

Результаты, полученные на наборе данных «CICIDS2017»

Алгоритм	Точность, %
GBT	99,81
DNN	97,72
GS-F1	97,5
RF	97,1
NB	72,2

Из табл. 4 видно, что наилучший результат показала модель GBT. Худший результат показала модель NB. НК, построенный с помощью алгоритма «саранчи», показал третий результат.

Заключение

В работе был исследован нечеткий классификатор, построенный с помощью метаэвристического алгоритма «саранчи» в рамках задачи обнаружения сетевых атак.

На рассмотренных наборах данных точность классификации после применения метаэвристического алгоритма «саранчи» повысилась.

На наборе данных «UNSW-NB15» был достигнут четвертый результат по точности на тестовой выборке. На наборе данных «CICIDS2017» был достигнут третий результат по точности.

Достигнутые результаты: на наборах данных «UNSW-NB15» – 84,15% и «CICIDS2017» – 97,5%, что является достаточными результатами для построения системы обнаружения вторжений на основе нечеткого классификатора, построенного на основе метаэвристического алгоритма «саранчи».

Литература

1. Global mobile data traffic from 2017 to 2022 [Электронный ресурс]: сайт Statista. – URL: https://www.statista.com/statistics/271405/global-mobile-data-traffic-forecast/ (дата обращение: 16.06.2022).

2. Анализ зашифрованного сетевого трафика на основе вычисления энтропии и применения нейросетевых

классификаторов / В.А. Буковшин, П.А. Чуб, Д.А. Короченцев, Л.В. Черкесова, Н.В. Болдырихин, О.А. Сафарьян // Известия ЮФУ. Технические науки. – 2020. – № 6 (216).

3. Explainable Fuzzy Systems: Paving the Way from Interpretable Fuzzy Systems to Explainable AI Systems / J.M.A. Moral, C. Castiello, L. Magdalena, C. Mencar. – Cham: Springer, 2021. – 253 p.

4. Ходашинский И.А. Параметрическая идентификация нечетких моделей на основе гибридного алгоритма муравьиной колонии / И.А. Ходашинский, П.А. Дудин // Автометрия. – 2008. – Т. 44, № 5. – С. 24–35.

5. Корышев Н.П. Алгоритм формирования базы правил нечёткого классификатора на основе алгоритма кластеризации k-средних и метаэвристического алгоритма «китов» / Н.П. Корышев, И.А. Ходашинский // Доклады ТУСУР. – 2021. – Т. 24, № 1. – С. 42–47.

6. Бардамова М.Б. Формирование структуры нечёткого классификатора комбинацией алгоритма экстремумов классов и алгоритма «прыгающих лягушек» для несбалансированных данных с двумя классами / М.Б. Бардамова, И.А. Ходашинский // Автометрия. – 2021. – Т. 57, № 4. – С. 54–64.

7. Ходашинский И.А. Отбор классифицирующих признаков с помощью популяционного случайного поиска с памятью / И.А. Ходашинский, К.С. Сарин // Автоматика и телемеханика. – 2019. – № 2. – С. 161– 172.

8. Ходашинский И.А. Отбор классифицирующих признаков: сравнительный анализ бинарных метаэвристик и популяционного алгоритма с адаптивной памятью / И.А. Ходашинский, К.С. Сарин // Программирование. – 2019. – № 5. – С. 3– 9.

9. Ходашинский И.А. Идентификация нечетких систем на базе алгоритма имитации отжига и методов, основанных на производных // Информационные технологии. – 2012. – № 3. – С. 14– 20.

10. Ходашинский И.А. Оптимизация параметров нечетких систем на основе модифицированного алгоритма пчелиной колонии / И.А. Ходашинский, И.В. Горбунов // Мехатроника, автоматизация, управление. – 2012. – № 10. – С. 15–20.

11. Saremi S. Grasshopper Optimisation Algorithm: Theory and application / S. Saremi, S. Mirjalilia, A. Lewis, S. Saremietal // Advances in Engineering Software. – 2017. – Vol. 105. – PP. 30–47.

12. Zakeri A. Efficient feature selection method using real-valued grasshopper optimization algorithm / A. Zakeri, A. Hokmabadi //Expert Systems with Applications. – 2019. – Vol. 119. – PP. 61– 72.

13. Kamel S.R. Feature selection using grasshopper optimization algorithm in diagnosis of diabetes disease / S.R. Kamel, R. Yaghoubzadeh // Informatics in Medicine Unlocked. – 2021. – Vol. 26. – P. 100707.

14. Остапенко Р.О. Формирование базы правил нечёткого классификатора с помощью метаэвристического алгоритма «саранчи» / Р.О. Остапенко, И.А. Ходашинский // Доклады ТУСУР. – 2022. – Т. 25, № 2. – 2022. – С. 31–36.

15. The UNSW-NB15 Dataset [Электронный ресурс]: сайт UNSW Research. – URL: https://research.unsw.edu.au/projects/unsw-nb15-dataset (дата обращения: 13.06.2022).

16. Sharafaldin I., Lashkari A.H., Ghorbani A.A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization // Proc. of the 4th International Conference on Information Systems Security and Privacy (ICISSP). – 2018. – № 1. – PP. 108–116.

17. Mustapha Belouch, Salah El Hadaj, Mohamed Idhammad Performance evaluation of intrusion detection based on machine learning using Apache Spark // Procedia Computer Science. – 2018. – № 127. – PP. 1–6.

18. Горюнов М.Н., Мацкевич А.Г., Рыболовлев Д.А. Синтез модели машинного обучения для обнаружения компьютерных атак на основе набора данных CICIDS–2017 // Труды ИСП РАН. – 2020. – № 32(5). – С. 81–93.

19. Faker O., Dogdu E. Intrusion Detection Using Big Data and Deep Learning Techniques // ACM SE '19 Proceedings of the 2019 ACM Southeast Conference. – 2019. – PP. 86–93.

Остапенко Роман Олегович
Студент каф. КИБЭВС ТУСУРа
Эл. почта: romanOstpub@mail.ru

Ходашинский Илья Александрович
Д-р техн. наук, профессор каф. КСУП ТУСУРа
ORCID: https://orcid.org/0000-0002-9355-7638
Эл. почта: hodashn@gmail.com

УДК 004.02

А.А. Паскаль

Модификации алгоритма Future Search Algorithm для отбора признаков в нечетком классификаторе

Рассмотрены и реализованы шесть методов модификации метаэвристического алгоритма Future Search Algorithm для задачи отбора признаков в нечетком классификаторе в режиме обертки. Проведено сравнение эффективности реализованных методов для метаэвристики Future Search на 18 наборов данных из репозитория KEEL [1]. Наилучшим методом по точности классификации и количеству отобранных признаков стал метод, основанный на модифицированных алгебраических операциях и операции слияния.

Ключевые слова: FSA, нечеткий классификатор, отбор признаков, оптимизация, бинаризация, классификация.

Этап отбора признаков является одним из самых эффективных и полезных способов улучшения модели классификации. Этот этап предполагает выбор из всего множества признаков такого подмножества, содержащего меньшее количество признаков по сравнению с полным множеством и позволяющее получить большую или такую же точность классификации. Отобранное подмножество признаков должно быть релевантным и представлять все классы, присутствующие в обучающей выборке.

Метаэвристики всё чаще используются в различных задачах, в том числе для отбора признаков [2].

В данной работе был разработан метаэвристический алгоритм Future Search Algorithm, функционирующий в режиме обёртки в бинарном пространстве поиска.

Постановка задачи

При отборе признаков возникают два состояния: 0 – признак не используется; 1 – признак используется. Поэтому главным условием, позволяющим применить алгоритм, является его способность работать в бинарном пространстве поиска.

Так как метаэвристический алгоритм Future Search Algorithm [3] работает в непрерывном пространстве поиска, необходимо осуществить его бинаризацию – модификацию с целью организации поиска глобального оптимума в бинарном пространстве.

Future Search Algorithm

Данный алгоритм имитирует поведение человека. Человек, который добивается лучших результатов, является оптимальным локальным решением среди других людей. Каждый год (каждую итерацию) это решение может измениться на другого человека. Наилучшее из них признаётся глобальным решением.

Шаг 1. Инициализация популяции:

$$X_i = Lb + (Ub - Lb) * rand, \quad (1)$$

где X_i – особь популяции, Lb – границы нижнего предела, Ub – границы верхнего предела, $rand$ – равномерно распределенные псевдослучайные числа.

Шаг 2. Определение X_{L_i} и X_{G_i}:

$$X_{L_i}(t+1) = (LS(t) - x_i(t)) * rand, \quad (2)$$

где $LS(t)$ – все решения;

$$X_{G_i}(t+1) = (GS(t) - x_i(t)) * rand, \quad (3)$$

где $GS(t)$ – наилучшее решение.

Шаг 3. Инициализация улучшенной популяции:

$$X_i(t+1) = x_i(t) + X_{L_i}(t+1) + X_{G_i}(t+1). \quad (4)$$

Шаг 4. Обновление $LS(t)$ и $GS(t)$.

Шаг 5. Обновление случайного начального значения:

$$Y_i(t+1) = GS(t) + (GS(t) - LS(t)) * rand. \quad (5)$$

Шаг 6. Обновление $LS(t)$ и $GS(t)$.

Шаги 2–6 выполняются в зависимости от количества заданных итераций.

Для задачи отбора признаков структура алгоритма была изменена, а именно были удалены шаги 5 и 6. Это позволило производить вычисления быстрее без потери эффективности.

Метод трансформационных функций

Метод трансформационных функций [4] основывается на преобразовании непрерывного вектора к бинарному виду с помощью трансформационной функции, формирующей промежуточное решение.

Были реализованы три версии алгоритма FSA бинарного вида с использованием следующих трансформационных-функций. Одна S-образная

$$f_1 = \left| \frac{1}{\sqrt{1 + x_i^2}} \right| \quad (6)$$

и две V-образные

$$f_2 = \left| \frac{x_i}{\sqrt{1 + x_i^2}} \right|, \quad (7)$$

$$f_3 = \left| \tanh(x_i) \right|. \quad (8)$$

Для трансформации вещественного значения в бинарное использовано следующее правило:

IF $rand\ (0; 1) < f(x_i)$,
Then $s = 1$, Else $s = 0$.

Метод модифицированных алгебраических операций

В методе модифицированных алгебраических операций все арифметические операторы заменяются логическими, что позволяет алгоритму работать без выходов из бинарного пространства.

В соответствии с этим методом формулы FSA были заменены следующими:

$$X_{L_i}(t+1)=\left(LS(t)\oplus x_i(t)\right)\wedge rand\ binvector, \quad (9)$$

$$X_{G_i}(t+1)=\left(GS(t)\oplus x_i(t)\right)\wedge rand\ binvector, \quad (10)$$

$$x_i(t+1)=x_i(t)\vee X_{L_i}(t+1)\vee X_{G_i}(t+1). \quad (11)$$

Метод бинаризации, основанный на операции слияния (MERGE)

В работе [5] введена авторская операция слияния (MERGE). Функция MERGE (далее – *M*) сравнивает два бинарных вектора. Если значение элемента в обоих векторах совпадает, то в результирующем векторе на эту позицию запишется совпадающее значение. В ином случае будет сгенерировано случайное число от 0 до 1. Если оно окажется меньше или равно 0,5, то элементу нового вектора будет присвоено значение из старого вектора. Если число окажется больше 0,5, то в новый вектор запишется значение из лучшего вектора.

Для этого метода формулы FSA были заменены следующими:

$$X_{L_i}(t+1)=M\left(M\left(LS(t),x_i(t)\right),rand\ binvector\right), \quad (12)$$

$$X_{G_i}(t+1)=M\left(M\left(GS(t),x_i(t)\right),X_{G_i}(t+1)\right), \quad (13)$$

$$X_i(t+1)=M\left(x_i(t),M\left(X_{L_i}(t+1),X_{G_i}(t+1)\right)\right). \quad (14)$$

Комбинация методов модифицированных алгебраических операций и слияния

Этот авторский метод, предложенный в [5], является синтезом модифицированных алгебраических операций и операции слияния. Отличие от метода модифицированных алгебраических операций заключается в том, что операция сложения заменяется на операцию MERGE.

В соответствии с этим методом формулы FSA приведены ниже:

$$X_{L_i}(t+1)=\left(LS(t)\oplus x_i(t)\right)\wedge rand\ binvector, \quad (15)$$

$$X_{G_i}(t+1)=\left(GS(t)\oplus x_i(t)\right)\wedge rand\ binvector, \quad (16)$$

$$X_i(t+1)=M\left(x_i(t),M\left(X_{L_i}(t+1),X_{G_i}(t+1)\right)\right). \quad (17)$$

Описание эксперимента

Вычислительный эксперимент был проведён на 18 различных сбалансированных наборах данных для классификации из репозитория KEEL (табл. 1) и был разделен с помощью процедуры 10-кратной перекрёстной проверки.

В качестве классификатора использовался нечеткий классификатор, оценка качества классификации производилась с помощью численной метрики точности (accuracy), использовались треугольные функции принадлежности, в качестве алгоритма генерации правил был использован алгоритм экстремумов классов. Для оптимизации параметров термов был применён Future Search Algorithm, работающий в непрерывном пространстве поиска. В качестве алгоритма отбора признаков использовался Future Search Algorithm, бинаризованный шестью различными способами, описанными выше. Аббревиатуры представлены в табл. 2.

Использовались следующие параметры алгоритма для всех наборов: 25 частиц популяции, 250 итераций. Эксперимент был проведен 10 раз для каждого набора данных. Результаты были усреднены.

В ходе эксперимента была произведена оценка двух параметров: точности классификации на тестовых данных и количества признаков, выбранных алгоритмом. Каждый раз применялась оптимизация параметров термов. Результаты эксперимента представлены в табл. 3, 4.

Таблица 1

Наборы данных

Dataset	Признаки	Экземпляры	Классы
australian	14	690	2
ionosphere	33	351	2
page-blocks	10	5472	5
segment	19	2310	7
sonar	60	208	2
spambase	57	4597	2
spectfheart	44	267	2
winequality-red	11	1599	11
yeast	8	1484	10
movement_libras	90	360	15
optdigits	64	5620	10
ring	20	7200	2
satimage	36	6435	7
texture	40	5500	11
thyroid	21	7200	3
twonorm	20	7400	2
vehicle	18	846	4
wdbc	30	569	2

Таблица 2

Обозначение методов бинаризации

Обозначение	МЕТОД
NO SELECT	Без отбора признаков
OPER	Метод модифицированных алгебраических операций
MERGE	Метод, основанный на операции MERGE
OPMERGE	Гибридный метод, основанный на методе модифицированных алгебраических операций и операции MERGE
SIGM	Метод трансформационных функций с функцией трансформации f_1
V	Метод трансформационных функций с функцией трансформации f_2
TANH	Метод трансформационных функций с функцией трансформации f_3

Обобщив результаты, можно сделать вывод, что метод OPMERGE позволил найти минимальное подмножество признаков при наибольшей точности классификации.

Результаты тестирования были усреднены по значению точности по всем наборам, а также по количеству отобранных признаков. Наглядный график представлен на рис. 1. Звездочкой отмечен наилучший метод.

Таблица 3

Среднее значение точности, полученное по каждому методу

DATASET	NO SELECT	OPER	MERGE	OPMERGE	SIGM	V	TANH
australian	79,00	72,09	71,86	71,86	71,91	71,42	71,74
ionosphere	88,33	74,65	74,76	76,21	74,34	74,88	75,35
page-blocks	90,12	89,90	90,34	90,14	89,70	89,57	89,26
segment	84,57	36,13	38,27	36,82	35,52	35,82	35,47
sonar	66,10	59,87	62,02	63,50	60,25	62,99	62,58
spambase	39,42	68,91	68,26	69,19	70,59	70,46	70,24
spectfheart	78,30	77,93	78,67	78,85	78,82	78,14	78,86
winequality-red	46,58	45,74	47,02	47,53	47,03	46,41	45,57
yeast	39,16	32,64	31,93	32,26	32,08	32,15	32,41
movement_libras	51,60	12,11	14,06	12,43	11,11	12,99	12,36
optdigits	19,68	17,78	19,06	19,14	18,46	19,17	18,13
ring	76,74	67,43	66,38	66,32	67,45	66,71	67,18
satimage	64,53	43,86	47,19	46,45	44,18	47,96	48,13
texture	70,66	23,81	26,37	27,65	24,92	27,14	26,48
thyroid	31,75	92,64	93,31	93,09	92,93	93,02	93,14
twonorm	96,31	76,57	76,56	77,45	75,47	76,66	77,15
vehicle	39,40	37,71	36,27	35,75	35,42	34,89	35,23
wdbc	94,42	87,28	90,09	90,69	85,00	88,71	89,58

Таблица 4

Среднее количество признаков, отобранных каждым методом

DATASET	NO SELECT	OPER	MERGE	OPMERGE	SIGM	V	TANH
australian	14,00	5,21	4,72	4,28	6,53	5,53	5,59
ionosphere	33,00	17,87	14,85	14,03	20,43	17,94	18,08
page-blocks	10,00	0,86	2,05	1,95	3,76	3,71	3,76
segment	19,00	10,73	5,73	5,58	9,94	7,56	7,89
sonar	60,00	33,79	3,02	3,09	36,62	3,91	4,61
spambase	57,00	21,19	3,19	3,36	20,04	4,53	4,79
spectfheart	44,00	31,20	22,05	18,61	29,15	20,70	21,24
winequality-red	11,00	5,57	3,18	3,15	4,51	4,46	4,45
yeast	8,00	4,14	3,85	3,75	4,50	4,50	4,50
movement_libras	90,00	52,54	22,24	19,18	52,53	28,90	27,98
optdigits	64,00	31,64	13,14	12,73	33,83	23,58	23,78
ring	20,00	6,76	2,02	1,95	3,80	2,75	2,80
satimage	36,00	16,58	4,10	4,08	13,37	4,20	4,40
texture	40,00	21,92	4,40	4,48	19,57	5,33	5,40
thyroid	21,00	10,04	3,92	3,63	8,83	6,20	6,63
twonorm	20,00	20,00	20,00	20,00	18,73	19,55	19,00
vehicle	18,00	8,86	3,62	3,70	8,20	5,60	5,73
wdbc	30,00	14,44	4,20	4,15	13,27	5,58	6,00

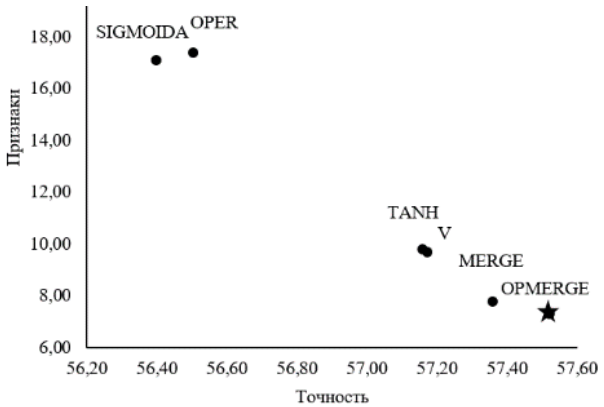


Рис. 1. График зависимости количества признаков от точности

Сравнение реализованных методов бинаризации с аналогами

Было проведено сравнение бинарных вариантов FSA с аналогами, такими как Rock Hyraxes Swarm (RHA) и Harris Hawks Optimization (HHO). Тестирование всех алгоритмов производилось со следующими параметрами: размер популяции 25, количество итераций 250. Для каждого набора данных эксперимент повторялся 10 раз, после чего результаты усреднялись. Эксперименты проводились по схеме 10-кратной кроссвалидации. Были выбраны 18 наборов данных, представленных в репозитории KEEL (см. табл. 1).

Сравнение алгоритмов производилось по критерию Фридмана по всем реализованным методам. Результаты представлены в табл. 5. Определена следующая H_0: распределения FSA, RHA и HHO одинаковы.

Таблица 5

Двухфакторный ранговый дисперсионный анализ Фридмана для связанных выборок

p-value	α
0,002	0,05

Полученное значение *p*-value меньше установленного уровня значимости. Следовательно, нулевая гипотеза о равенстве значений между методами отклоняется.

Для статистического сравнения был использован критерий Уилкоксона. Сравнение было проведено по лучшему методу в каждом алгоритме. Была выдвинута нулевая гипотеза: статистическая значимость между результатами отсутствует. Принятый уровень значимости $\alpha = 0{,}05$. В табл. 6 приведены значения *p*-value, ССК – стандартизированная статистика критерия.

Таблица 6

Критерий Уилкоксона для сравнения методов отбора признаков

Название	Обучение		Тестирование	
	ССК	*p*-value	ССК	*p*-value
OPMERGE FSA – MERGE HHO	3,724	<0,001	3,462	<0,001
OPMERGE FSA – OPMERGE RHA	1,851	0,064	0,936	0,349

При сравнении FSA на обучении и тестировании с HHO можно наблюдать статистическое различие, так как *p*-value меньше принятого уровня значимости.

При сравнении FSA на обучении и тестировании с RHA можно заметить, что различия между выборками являются статистически незначимыми.

Положительное значение ССК говорит о том, что методы HHO и RHA показали значения ошибки больше, чем метод FSA.

Заключение

В данной работе было рассмотрено шесть подходов к бинаризации Future Search Algorithm для решения задачи отбора признаков при классификации 18 наборов данных. Во всех реализованных методах бинаризации удалось добиться значительного уменьшения количества признаков.

Во всех реализованных методах бинаризации удалось добиться значительного уменьшения количества признаков. В 4 случаях значение точности классификации удалось увеличить при помощи отбора нерелевантных признаков. В остальных случаях (в 13) точность при отборе и оптимизации оказалась не выше, чем точность только при оптимизации. Однако в данных случаях количество признаков было достаточно сильно сокращено, что позволило уменьшить сложность модели и производить вычисления быстрее.

Лучшие способности к отбору признаков и наибольшую точность классификации показал гибридный метод, основанный на методе модифицированных алгебраических операций и операции MERGE. Данный метод бинаризации рекомендуется к использованию в задаче отбора признаков.

Худшие результаты показали метод трансформационных функций с функцией трансформации f_1 и метод модифицированных алгебраических операций. Они отобрали наибольшее количество признаков и показали наименьшую точность классификации.

Литература

1. Datasets for classification – KEEL [Электронный ресурс]. – Режим доступа: https:// sci2s.ugr.es/keel/datasets.php

2. Ходашинский И.А., Бардамова М.Б. Модификации алгоритма прыгающих лягушек для отбора признаков в нечетком классификаторе при аутентификации пользователя по рукописной подписи // Информационные и математические технологии в науке и управлении. – 2020. – № 4 (20). – С.75–83. DOI: 10.38028/ESI.2020.20.4.007

3. Elsisi M. Future search algorithm for optimization // Evolutionary Intelligence. – 2019. – Vol. 12, No. 1. – PP. 21–31.

4. Mirjalili S., Lewis A. S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization // Swarm and Evolutionary Computation. – 2013. – Vol. 9. – PP. 1–14.

5. Слезкин А.О., Ходашинский И.А., Шелупанов А.А. Методы бинаризации алгоритма стаи ласточек для решения задачи отбора признаков // Программирование. – 2021. – № 5. – С. 44–59.

Паскаль Анна Александровна
Студентка каф. БИС ТУСУРа
Эл. почта: paskalanna@mail.ru

УДК 004.912, 004.656

Е.Ю. Печенина, Н.В. Рузанов, В.А. Печенин

Разработка информационной системы подбора научных статей с использованием глубокого обучения

Разработана информационная система, позволяющая автоматизировать скачивание ключевой информации по статьям. Отличительной особенностью системы является дальнейший подбор подходящих конкретному автору статей, в соответствии с областью его научных интересов, на основании использования алгоритмов глубокого обучения.

Ключевые слова: информационная система, обработка естественного языка, глубокое обучение, вектор слова, долгая краткосрочная память.

В последние десятилетия количество публикаций отечественных учёных в международных журналах растёт, о чем свидетельствуют данные международной базы цитирования Scopus. Согласно данным аналитической платформы InCites от 11.03.2019 г., исследователи с российскими аффилиациями опубликовали за 2013–2018 гг. 382 935 работ, что составило 2,4 % от общемирового массива научных статей (16 051 825) и обеспечило России 15-е место среди 223 стран, представленных в WoS CC. Особенно велика доля публикаций у учёных, работающих в ведущих вузах страны, входящих в международные коллаборации, например, CMS, ATЛAC.

Ученым, у которых среди соавторов отсутствуют китайские или европейские коллеги, довольно сложно опубликовать свои работы в высокорейтинговых журналах, входящих в первый квартиль. Есть даже журналы уровня Q1 из России, которые могут подходить данным исследователям. Стоит отметить, что ученых с потенциалом исследований для публикации в ведущих журналах достаточно много, часть их работ могла бы повысить долю российской науки на международном уровне.

Известно, что поиск подходящего журнала может занять много времени. Кроме того, много времени занимает поиск литературы по теме своих исследований, ее изучение, формирование необходимого и достаточного литературного обзора по теме исследования. Возможно, разрабатываемый подход или метод был уже создан в относительно недавнем времени.

Интернет в данном случае предоставляет как большие возможности, так и большие трудности: на анализ такого большого объема информации ученый или даже группа исследователей зачастую не имеют достаточно много времени. Решением может стать информационно-аналитическая система, которая сможет гибко подготавливать всю необходимую информацию за короткий срок. Достаточную эффективность и необходимую гибкость системе (в отличие от существующих, например, researchgate) возможно придать с использованием методов машинного обучения, направленных на обработку естественного языка (NLP). Можно выделить три крупные задачи NLP: классификация области текста (определение части речи, эмоциональной окраски); выполнение связи нескольких частей в тексте; воспроизведение отсутствующей информации (пропущенное слово) по контексту [1].

Целью работы является разработка информационной системы, в которой для подбора статей, соответствующих профилю автора, используется глубокая нейронная сеть. Таким образом, нейронная сеть будет классифицировать статьи на два класса: подходят / не подходят автору.

Ход работы

Система была реализована на языке Python. Остальной стэк технологий в модуле получения данных разработанной системы следующий: библиотека pybliometrics – обёртка над Scopus API, позволяющая автоматизировать поиск по базе; библиотека pyodbc – для реализации доступа к базам данных при помощи протокола ODBC; библиотеки pickle, json – используются для хранения промежуточных данных.

Первая функция системы – выполнять парсинг статей из базы Scopus. Пользователь задает временной период и журнал. Соответственно, модель данных изображена на рис. 1. Полученная информация сохраняется в базе данных, в качестве системы управления реляционными базами данных использована Microsift SQL Server.

В базе данных, помимо информации о статье, хранится информация об авторе, журнале и месте работы автора. После – сохранение всех статей из интересующих журналов за определенный период.

Для интеллектуального поиска подходящих статей используется глубокое обучение, позволяющее классифицировать статьи по их содержанию (из аннотаций) согласно степени сходства со статьями исследователя.

Сохраненная в базе данных информация (таблица Articles) группируется по статьям, для проведения экспериментов по определению подходящих статей используется поле аннотации (DescriptionExtention1).

Как правило, слова в тексте преобразуются при помощи нейронных сетей в векторы относительно небольшой размерности. Стандартом в данной области является программное обеспечение Word2vec [1]. Для каждого слова в аннотации с использованием Word2vec был получен вектор длиной в 300 значений. Каждая аннотация либо усекалась, либо дополнялась нулями до *N* токенов.

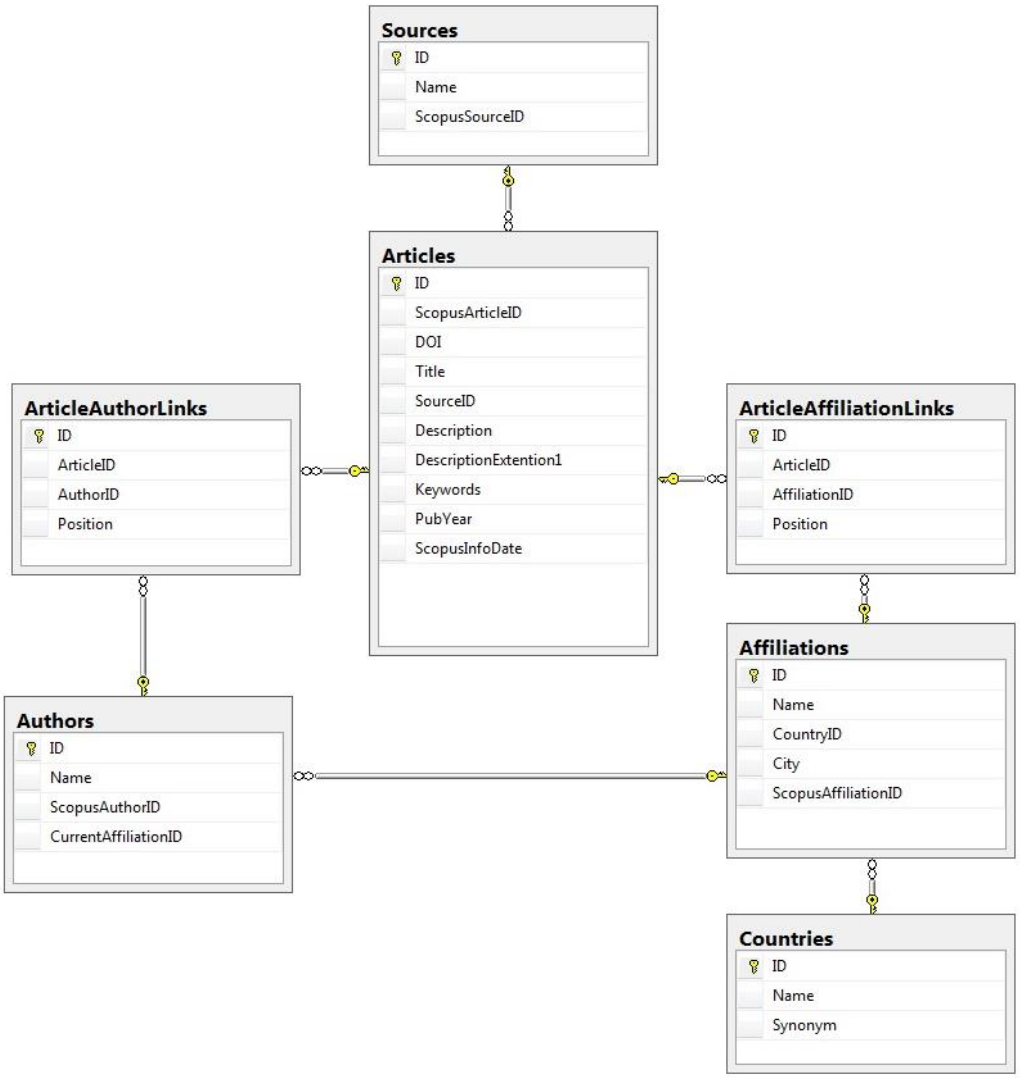


Рис. 1. Диаграмма модели данных сущность–связь, сохраняемых системой

Используемая глубокая нейронная сеть является рекуррентной и состоит из ячейки долгой кратковременной памяти (LSTM) [2] с M нейронами и выходным слоем с одним нейроном и сигма-функцией активации. Между LSTM и выходным слоем располагается слой дропаута [3] с параметром вероятности, с которой нейроны выбрасываются из слоя p = 0,2. Количество LSTM нейронов зависит от количества токенов, подаваемых на входе в нейронную сеть. Архитектура используемой нейронной сети приведена на рис. 2.

Для оценки ошибок в задаче классификации был использован коэффициент точности $\delta_{\text{клас}}$, вычисляемый как отношения количества правильно классифицированных объектов $N_{\text{прав}}$ к общему количеству объектов $N_{\text{общ}}$ выборки:

$$\delta_{\text{клас}} = N_{\text{прав}} / N_{\text{общ}}. \quad (1)$$

Соответственно если коэффициент равен 1, то получается 100%-ная эффективность работы сети.

Кроме того, для оценки качества обучения и тестирования была применена функция потерь L_f – бинарная кроссэнтропия.

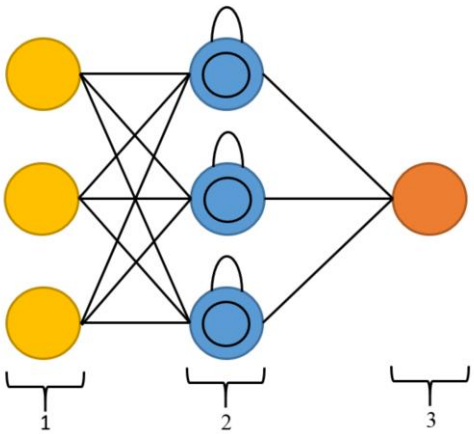


Рис. 2. Архитектура используемой нейронной сети:
1 – входной слой, *2* – слой LSTM,
3 – выходной слой с сигма-функцией

Всего для проведения обучения и тестирования было скачано 30 000 статей за 2016–2020 гг. В качестве профиля ученого, для которого выполняется поиск, был выбран профиль автора с ID 56401223700. В соответствии с данным профилем были размечены подходящие и не подходящие по

темам исследования автора статьи. Величина обучающей выборки составила 24 000 статей (80%), остальные были использованы для тестирования. Были проведены эксперименты по исследованию влияния количества входных токенов и нейронов в LSTM на точность работы сети.

В первом эксперименте для расчета величины входного вектора была ориентация на максимальное количество токенов (слов), используемых в аннотациях. Максимальная величина в выборке составила 312 токенов, минимальная – 8, средняя величина – 123. Размер N, по которому аннотация либо усекалась, либо дополнялась нулями, был выбран 300 токенов. Количество нейронов M в LSTM было взято 50, каждый нейрон имел 351 весовой коэффициент. Слой с LSMT имел всего 105 300 весовых коэффициентов, последующий слой 15 000 весовых коэффициентов.

На рис. 3 приведен график зависимости коэффициента $\delta_{класс}$ от количества эпох, на рис. 4 – график функции величины функции потерь L_f от количества эпох обучения.

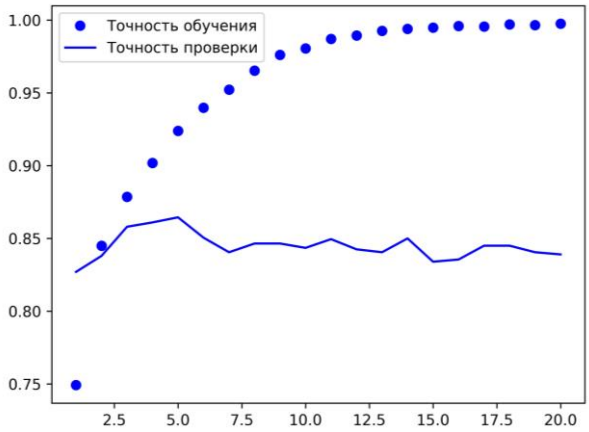


Рис. 3. График изменения точности по эпохам

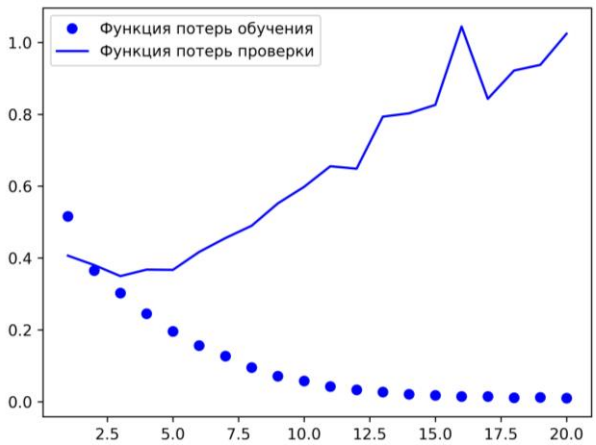


Рис. 4. График изменения L_f по эпохам

Оптимальное количество эпох обучения, согласно графикам, видится 7. Достигаемая точность классификации на тестовой выборке перестала улучшаться с 6-й эпохи, а функция потерь начала расти, что говорит о переобучении.

Во втором эксперименте величина N токенов была снижена до средней величины длины аннотации и составила 120. Количество весовых коэффициентов слоя с LSMT стало равно 42 120, у последующего слоя 6 000 весовых коэффициентов. На рис. 5 и 6 изображены графики зависимости коэффициента $\delta_{класс}$ и функции потерь L_f от количества эпох при уменьшенном количестве используемых на входе в сеть токенов.

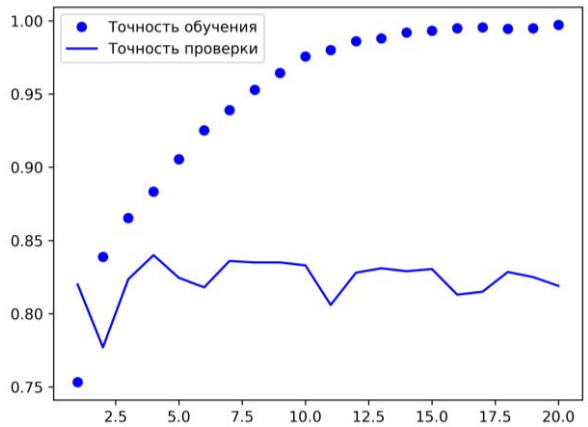


Рис. 5. График изменения точности для 120 токенов

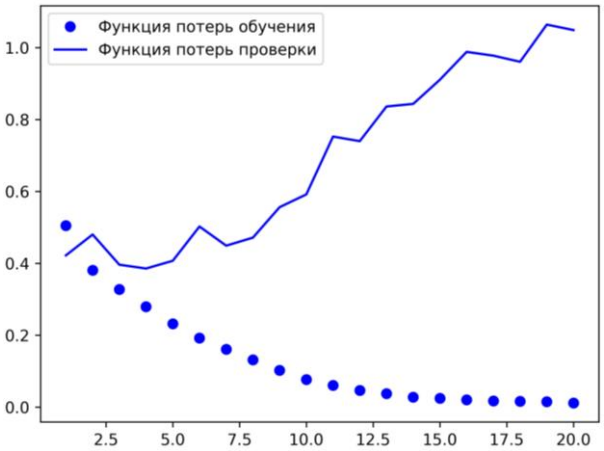


Рис. 6. График изменения L_f по эпохам для 120 токенов

В третьем эксперименте количество токенов было выбрано 120, количество нейронов в LSTM – 25. В четвертом эксперименте количество токенов 300 при 25 нейронов в LSTM.

В таблице приведены результаты по точности и функции потерь для всех проведенных экспериментов.

Точность функции потерь обучающих и тестовых выборках

Эпоха обучения	Параметры	1-й эксперимент		2-й эксперимент		3-й эксперимент		4-й эксперимент	
		Обучение	Тест	Обучение	Тест	Обучение	Тест	Обучение	Тест
5	$\delta_{класс}$	0,924	0,865	0,883	0,840	0,889	0,852	0,901	0,860
	L_f	0,200	0,367	0,279	0,385	0,276	0,341	0,244	0,348
20	$\delta_{класс}$	0,998	0,839	0,997	0,819	0,990	0,829	0,993	0,847
	L_f	0,010	1,024	0,011	1,049	0,034	0,741	0,027	0,789

Согласно полученным результатам, можно сделать вывод, что лучшая точность для тестовой выборки достигается в 1-м эксперименте (86,5%). Уменьшение количества токенов, как и уменьшение нейронов в LSTM не дает существенного улучшения, и можно сделать вывод, что брать количество токенов на основании максимальной длины аннотации является лучшей стратегией.

Заключение

Проведенное исследование является решением задачи подбора статей, соответствующих профилю выбранного автора, с использованием рекуррентной нейронной сети с ячейкой LSTM. Достигнута точность подбора в 86,5%. Создана программная система автоматизации предложенного решения. Дальнейшим развитием работы будет создание для системы своего легковесного фреймворка глубокого обучения, следующего последним тенденциям в разработке фреймворков, имеющий свой программный интерфейс и функциональные возможности для разрабатываемой системы. Конкретно данный фреймворк будет реализовывать обратное распространение ошибки и коррекцию весов в создаваемых нейронных сетях.

Разработанный фреймворк будет использован в создаваемой информационно-аналитической системе, обладающей следующим функционалом:

1) анализ текущей публикационной активности касающейся интересующей отрасли знаний. Загрузка информации по журналам Q1 за последние 5 лет, определение ключевых слов, выявление публикаций по заданным критериям;

2) определение ключевых направлений исследований. Анализ заголовков, аннотаций и определение направлений исследований. Определение актуальных направлений;

3) определение ведущих специалистов и центров разработки в интересующей отрасли знаний. Выявление исследователей и университетов с наибольшим рейтингом, отметка их на карте;

4) определение потенциальных кандидатов на сотрудничество. Выявление исследователей, работающих с исследователями из других университетов, городов, стран;

5) отслеживание новых публикаций, соответствующих различным критериям. Возможность оповещения на почту о выходе новой статьи;

6) отслеживание текущего взаимодействия с другими исследователями. Отслеживание ссылок на публикации интересующих авторов, организация взаимодействия по новым статьям.

Работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации в рамках гранта Президента Российской Федерации (номер МК-2156.2022.1.6).

Литература

1. Efficient Estimation of Word Representations in Vector Space / T. Mikolov, K. Chen, D. Corrado, J. Dean // ArXiv. – 2013. – URL: https://arxiv.org/abs/1301.3781 (дата обращения: 08.08.2022).

2. Траск Э. Грокаем глубокое обучение. – СПб.: Питер, 2019. – 352 с.

3. Николенко С.И. Глубокое обучение / С.И. Николенко, А. Кадурин, Е. Архангельская. – СПб.: Питер, 2018. – 480 с.

Печенина Екатерина Юрьевна
Аспирант каф. технологий производства двигателей (ТПД) Самарского университета
Эл. почта: ek-ko@list.ru

Рузанов Николай Владимирович
Канд. техн. наук, доцент, вед. программист каф. ТПД Самарского университета
Эл. почта: kinform_@mail.com

Печенин Вадим Андреевич
Канд. техн. наук, доцент, доцент каф. ТПД Самарского университета
Эл. почта: vadim.pechenin2015@gmail.com

УДК 004.02

Д.С. Савельев

Отбор признаков с помощью метаэвристического алгоритма пустынных канюков

Представлен алгоритм оптимизации пустынных канюков, который с помощью методов бинаризации алгоритма проводит отбор признаков в классификаторе, построенном на нечеткой системе.
Ключевые слова: алгоритм пустынных канюков, нечеткий классификатор, отбор признаков.

Метаэвристические алгоритмы широко применяются для оптимизации в различных задачах, в том числе с целью отбора информативных подмножеств признаков при построении модели машинного обучения. Удаление лишних признаков помогает избежать переобучения модели и уменьшить её сложность.

При использовании метаэвристик в качестве инструмента отбора признаков необходимым условием является способность алгоритма осуществлять поиск в бинарном пространстве.

На вход бинарной версии метаэвристического алгоритма пустынных канюков подается популяция, состоящая из векторов вида $\mathbf{S} = (S_1, S_2, \ldots, S_n)$. Каждый элемент вектора соответствует признаку из таблицы наблюдения, n – количество признаков. Значение элемента вектора $S_i = 0$ показывает, что i-й признак не участвует в классификации, $S_i = 1$ означает, что i-й признак используется классификатором. Алгоритм должен найти такой вектор **S**, который позволил бы максимально улучшить фитнес-функцию при уменьшении количества признаков.

Целью работы является отбор признаков в нечетком классификаторе с помощью метаэвристического алгоритма пустынных канюков, его тестирование в задаче классификации данных и сравнение с результатами других метаэвристик.

Методы бинаризации алгоритма

В данной работе будут рассмотрены следующие методы бинаризации:

1. Метод модифицированных алгебраических операций. Данный метод позволяет алгоритму оперировать бинарными входными векторами, так как все арифметические операторы заменяются на логические [1, 2]. В соответствии с этой идеей, операции умножения, сложения и вычитания заменяются на конъюнкцию, дизъюнкцию и строгую дизъюнкцию соответственно.

2. Метод слияния. Предполагает использование оригинальной функции слияния для осуществления бинарного поиска [3]. Функция поэлементно сравнивает два бинарных вектора:

– если значение элемента на одной и той же позиции совпадает, то в результирующем векторе на эту позицию запишется данное значение,

– иначе осуществляется генерация случайного числа из интервала от 0 до 1. Если оно меньше или равно 0,5, то в соответствующую позицию нового вектора записывается элемент из худшего вектора. В противном случае на этом месте будет выставлен элемент из лучшего вектора.

3. Комбинация методов модифицированных арифметических операций и слияния. Данная модификация представляет собой совмещение оператора, использующего логические операции, и функции слияния двух векторов.

Эксперимент и полученные результаты

Эксперимент проводится по схеме кросс-валидации с соотношением: 90% – обучающая выборка, 10% – тестовая на наборах данных репозитория KEEL. Отбор признаков с помощью алгоритма пустынных канюков оценивается с помощью набора данных, общая информация которых представлена в табл. 1, где Tr.B и Te.B – точность классификации до применения алгоритма на обучающей и тестовой выборке.

Таблица 1

Набор данных для эксперимента

Набор	Признак	Объект	Класс	Tr.B	Te.B
Australian - DS1	14	690	2	50,52	49,13
Ionosphere – DS2	33	351	2	79,61	79,76
Page-blocks – DS3	10	5472	5	50,4	51,17
Segment– DS4	19	2310	7	80,26	80,48
Sonar – DS5	60	208	2	57,37	57,24
Spambase – DS6	57	4597	2	39,42	39,42
Spectfheart – DS7	44	267	2	80,44	80,88
Vehicle – DS8	18	846	4	29,87	29,9
Winequality-red – DS9	11	1599	11	19,86	19,7
Yeast – DS10	8	1484	10	24,18	23,52
Movement_libras – DS11	90	360	15	47,62	48,06
Optdigits – DS12	64	5620	10	10	10,02
Ring – DS13	20	7400	2	49,55	49,53
Satimage – DS14	36	6435	7	60,26	60,36
Texture – DS15	40	550	11	69,78	69,96
Thyroid – DS16	21	7200	3	7,05	7,11
wdbc– DS17	30	569	2	92,58	91,91

Результаты работы методов алгоритма на семнадцати наборах данных представлены в табл. 2, где Tr.A и Te.A – точность классификации на обучающей и тестовой выборке после применения алгоритма, Feaut. – число отобранных признаков, Optimization – оптимизация без применения отбора, Merge – отбор методом слияния, Operations – метод модифицированных алгебраических операций, M+O – комбинация методов модифицированных арифметических операций и слияния.

Условия эксперимента для всех алгоритмов были одинаковы: количество выполнений – 10 раз, численность популяции – 25, число итераций – 250.

Как можно заметить из табл. 2, алгоритм работает на разных наборах данных, он производит отбор признаков и вычисляет точность классификации. Результаты зависят от набора данных, в каком-то наборе не стоит применять отбор, а следует оставить только оптимизацию, так как уменьшается точность классификации, соответственно, лучше применять оптимизацию без отбора. В каких-то наборах можно применить отбор признаков, мы уменьшаем наше количество признаков и получаем большую точность классификации после применения отбора и оптимизации. Также на некоторых наборах данных можно применить отбор признаков с уменьшением точности классификации на пару процентов, это позволит улучшить интерпретируемость.

Были построены графики зависимости среднего значения всех наборов точности классификации в зависимости от количества признаков для отбора (рис. 1).

Таблица 2

Результаты оптимизации алгоритма без применения отбора и с отбором признаков методами Merge, Operations и M+O

DataSet	Optimization			Merge			Operations			MergeOperation		
	Tr.A	Te.A	Feaut.	Tr.A	Te.A	Feaut.	Tr.A	Te.A	Feaut.	Tr.A	Te.A	Feaut.
DS1	69,5	69,13	14	67,46	64,49	5	68,02	68,12	5,8	70,81	69,57	5,6
DS2	79,77	78,05	33	74,04	75,78	22	72,71	70,92	23,1	72,37	70,68	20,5
DS3	82,65	82,47	10	90,1	89,93	4,4	88,35	88,36	3,8	89,65	89,71	4,2
DS4	82,88	81,63	19	25,85	25,97	9,7	31,4	31,17	10,5	27,25	27,4	10,2
DS5	70,14	64,4	60	63,89	64,38	33,4	60,31	58,19	31,7	59,14	50,95	32,3
DS6	39,42	39,42	57	62,22	61,91	32,4	60,6	60,5	32,8	60,71	60,63	31,3
DS7	81,4	78,05	44	79,36	79,05	21,5	78,98	78,66	21	78,94	79,42	22
DS8	35,54	34,07	18	34,41	34,04	8,1	32,65	31,45	8,8	34,88	33,34	8,9
DS9	40,23	39,72	11	42,75	42,53	6,1	43,64	43,66	4,5	42,07	41,34	6,2
DS10	32,03	31,67	8	29,13	29,72	4,1	31,79	31,74	4,4	29,78	29,99	4,8
DS11	50,28	49,96	90	12,04	10	45,9	10,96	11,11	45,9	11,64	10	45,8
DS12	12,45	12,26	64	16,68	16,62	35,4	15,05	15,05	41	17,37	17,37	34,9
DS13	53,28	53,12	20	62,59	62,49	6,7	57,94	57,51	6,9	61,03	61,72	6,9
DS14	60,55	60,61	36	33,07	33,83	17,9	31,15	31,31	18	33,75	33,47	16,7
DS15	70,26	70,37	40	17,29	17,58	19,8	16,92	16,96	24,2	19,05	19,58	21,1
DS16	7,3	7,31	21	92,58	92,58	9,1	83,82	83,83	11,9	92,63	92,64	10,9
DS17	93,71	92,43	30	75,53	75,03	14,7	68,15	67,66	13,1	77,95	77,49	14,8
Среднее значение	**56,21**	**55,37**	33,82	51,71	51,53	**17,42**	50,14	49,78	18,08	51,71	50,90	17,48

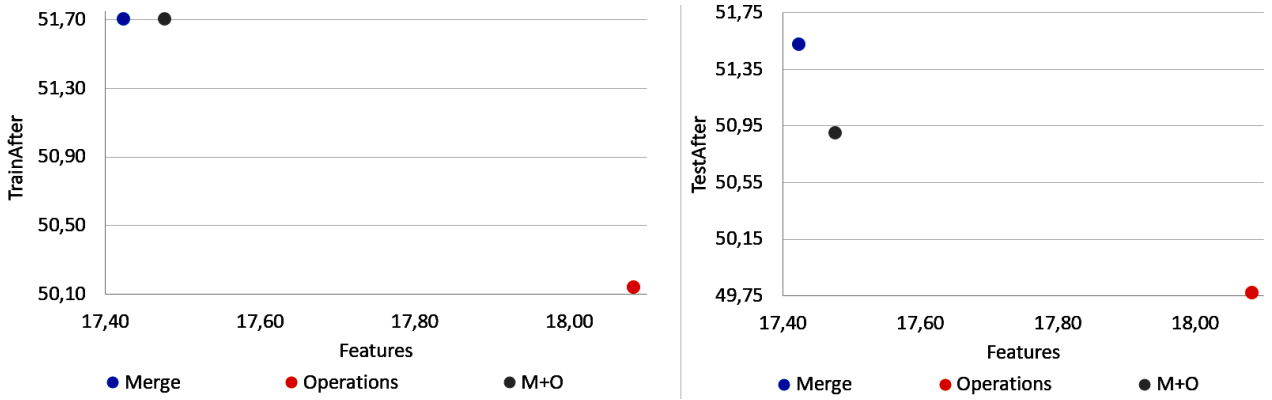


Рис. 1. Зависимость методов отбора алгоритма – количество признаков от точности на обучающем и тестовом наборе

Исходя из рис. 1, можно сказать, что метод отбора Merge в среднем справляется лучше всего с данными наборами данных, так как у него наименьшее количество отобранных признаков и наибольшая точность классификации по сравнению с двумя другими методами.

Сравнение с аналогами

Было проведено сравнение методов отбора с алгоритмами Future Search algorithm [4] и Rock Hyraxes algorithm [5].

Сравнение проводилось на одной вычислительной машине, в качестве входных параметров было решено взять $N = 25$, $T = 250$, где N – количество популяции, T – число итераций.

Тестирование проводилось на одинаковых наборах данных, было решено взять среднюю точность классификации на тестовых наборах по всем методам отбора, результаты занесены в табл. 3.

Таблица 3

Результаты сравнения с аналогами

Метод	Harris Hawks	Future Search	Rock Hyraxes
Merge	52,32	57,36	53,80
Operation	50,30	56,50	54,20
M+O	51,65	57,52	55,37

Полученные результаты переведены в ранги и занесены в табл. 4.

Таблица 4

Ранги алгоритмов

Метод	Harris Hawks	Future Search	Rock Hyraxes
Merge	3,00	1,00	2,00
Operation	3,00	1,00	2,00
M+O	3,00	1,00	2,00
Средний ранг	3	1	2

Выводы по таблицам: как можно заметить, наилучшими методами отбора для алгоритма пустынных канюков является метод Merge, для двух других алгоритмов наилучшим методом является метод M+O. По итогу можно сказать, что при данных наборах данных и при данных входных параметрах алгоритм пустынных канюков справляется хуже, так как точность меньше по сравнению с двумя другими алгоритмами.

Сравнение с аналогами

Дополнительно, в программе IBM SPSS Statistics, по критерию Уилкоксона были проведены сравнения: среди лучших – метод отбора, для Harris Hawks – метод Merge, для Future Search и Rock Hyraxes – метод M+O, результаты приведены в табл. 5.

Таблица 5

Ранги алгоритмов

Алгоритм	ССК	Асимптотическая значимость	Нулевая гипотеза
Train. HHO – Train. FSA	–3,724	<0,001	Отклоняется
Train. HHO – Train. RHO	–3,201	<0,001	Отклоняется
Test. HHO – Train. FSA	–3,462	<0,001	Отклоняется
Test. HHO – Test. RHO	–3,29	<0,001	Отклоняется

Асимптотическая значимость (p-value), полученная по результатам программы, меньше уровня значимости (0,05), значит, нулевая гипотеза отклоняется.

Нулевая гипотеза гласит об отсутствии статистически значимых различий, следовательно, между результатами есть различия.

Отрицательное значение стандартизированной статистики критерия (ССК) свидетельствует о превосходстве результатов аналогов над результатами алгоритма пустынного канюка (HHO).

Заключение

По итогам работы можно сделать выводы о работоспособности алгоритма оптимизации пустынных канюков в системе нечеткого классификатора. Как можно заметить, для некоторых наборов данных можно применить метод отбора признаков с оптимизацией, так как при этом увеличивается точность классификации, а для других достаточно применить оптимизацию, так как с отбором точность классификации становится хуже.

Литература

1. Jia D., Duan X., Khan M. K. Binary artificial bee colony optimization using bitwise operation // Computers and Industrial Engineering. – 2014. – Vol. 76. – PP. 360–365.

2. Kiran M.S., Gunduz M. XOR-Based Artificial Bee Colony Algorithm for Binary Optimization // Turkish Journal of Electrical Engineering and Computer Sciences. – 2013. – Vol. 21. – PP. 2307–2328.

3. Hodashinsky I., Sarin K., Shelupanov A., Slezkin A. Feature selection based on swallow swarm optimization for fuzzy classification // Symmetry. – 2019. – Vol. 11, No. 11. – PP. 1423.

4. Elsisi, M. Future search algorithm for optimization. // Evolutionary Intelligence. – 2019. – Vol. 12. – PP. 21–31.

5. Jain M., Maurya S., Rani A., Singh V. Owl search algorithm: A novel nature-inspired heuristic paradigm for global optimization // Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology. – 2018. – Vol. 34, Iss. 32018. – PP. 1573–1582.

Савельев Денис Сергеевич
Студент Факультета безопасности ТУСУР гр.747
Эл. почта: savelyev.denis7@gmail.com

УДК 004.852

М.О. Светлаков

Применение нечеткого классификатора для аутентификации пользователя на основе данных электроэнцефалограммы

Представлены результаты исследования по применению данных электроэнцефалограммы (ЭЭГ) для аутентификации пользователей с использованием дискретного вейвлет-преобразования и нечеткого классификатора. В исследовании использовался общедоступный набор данных PhysioNet EEG Motor Movement / Imagery Dataset (PEEGMIMDB), а в качестве признаков – среднее значение, стандартное отклонение и среднеквадратичное значение коэффициентов дискретного вейвлет-преобразования. Для отбора признаков использованы критерий взаимной информации и бинарный метаэвристический алгоритм Bird Swarm. Для классификации используется нечеткий классификатор, построенный с помощью кластеризации и оптимизированный непрерывным метаэвристическим алгоритмом Bird Swarm. Эффективность построенного классификатора была проверена с помощью процедуры кроссвалидации. Лучший построенный нечеткий классификатор достиг точности 79,37%.

Ключевые слова: ЭЭГ, Bird Swarm, метаэвристический алгоритм, отбор признаков, нечеткий классификатор, кластеризация, K-means, биометрия.

Аутентификация на основе биометрии считается более удобной и надежной по сравнению с аутентификацией на основе паролей [1]. Биометрические системы аутентификации основываются на уникальных биологических характеристиках человека, например, используется радужка глаза, рисунок вен, походка, а также в качестве подобной уникальной характеристики может применяться активность мозга пользователя, наблюдаемая через сигнал электроэнцефалограммы. Использование ЭЭГ для биометрии привлекает интерес ученых, так как в настоящий момент биометрические сигналы ЭЭГ не могут быть легко воспроизведены злоумышленником [2]. Данные ЭЭГ также многофункциональны, применимы для распознавания эмоций, изучение сна и здоровья. Из-за того, что данные ЭЭГ могут меняться в зависимости от эмоционального состояния пользователя [3], злоумышленник с меньшей вероятностью может принудительно заставить легального пользователя предоставить доступ. Сигнал ЭЭГ также не получится использовать для слежки за пользователем, как в случае с изображением лиц и системами распознавания на их основе [4].

Интерпретируемость моделей машинного обучения является одной из важнейших проблем сегодняшнего дня. Для биометрической аутентификации эта проблема особенно важна в аспекте повышения прозрачности работы системы для пользователя. является важной задачей машинного обучения. Одним из методов решения этой проблемы является определение частей входных данных, которые вносят наибольший вклад в конечный результат модели [5]. В данной работе предлагается использовать нечеткий классификатор, основанный на правилах вида ЕСЛИ-ТО, принцип работы и принятия решения которых понятен для пользователя и администратора системы аутентификации.

В данной работе предложено использование нечеткого классификатора, построенного методами кластеризации и оптимизированного метаэвристическим алгоритмом Bird Swarm. Для отбора признаков применялся бинарный метаэвристический алгоритм Bird Swarm, в качестве признаков использовались статистические характеристики, вычисленные на основе коэффициентов дискретного вейвлет-преобразования. Построенные классификаторы были протестированы на общедоступном наборе данных PhysioNet EEG Motor Movement/Imagery Dataset (PEEGMIMDB) с достижением точности классификации 79,37%. Основным преимуществом предложенного подхода является интерпретируемость итогового классификатора, присущая системам, работающим на основе нечетких правил.

Набор данных и препроцессинг

Набор данных сигналов ЭЭГ Physionet EEG Motor Movement/Imagery Dataset [6] содержит записи 109 человек. Первоначально записи ЭЭГ представляют собой наборы по 64 временные серии (соответственно числу электродов), записанные с частотой дискретизации 160 Гц. Участники совершали двигательные задания и задачи представления, пока велась запись 64-канального ЭЭГ-сигнала с использованием системы BCI2000. Данные были разделены на эпохи длительностью 1,5 с (рис. 1). В данной работе использованы данные только 8 каналов (O1, O2, P3, P4, C3, C4, F3, F4) и данные ЭЭГ только для состояний «глаза открыты» и «глаза закрыты». После обработки имеем следующие размеры набора данных – 8 715 образцов, 8 каналов. Каждый образец представляет собой временной ряд из 144 точек, поэтому представим размерность в следующем виде – 8 715 образцов, 144 точка, 8 каналов.

Извлечение признаков

Вейвлет-преобразование позволяет получить частотно-временное представление сигнала в различных масштабах, при этом выделяя различные особенности/признаки. Дискретное вейвлет-преобразование можно описать следующим образом: сигнал одновременно пропускается через низкочастотные и высокочастотные фильтры, в результате получаются коэффициенты аппроксимации и детализирующие коэффициенты соответственно. Это разложение можно повторять несколько раз для получения сигнала в различных масштабах [7].

Для преобразования исходного сигнала был применен базисный вейвлет Добеши 4-го порядка, для декомпозиции сигнала на 5 уровней, соответствующих ритмам Альфа (8–16 Гц), Бета (16–32 Гц), Гамма (32–64), Дельта (0,5–4 Гц), Тета (4–8 Гц), ЭЭГ-ритмам. Наиболее высокая частота соответствует максимальной по теореме Найквиста–Шеннона.

На основе коэффициентов вычислены признаки: среднее значение, стандартное отклонение и среднеквадратичное значение. Схема выделения признаков представлена на рис. 2. После извлечения признаков имеем следующую размерность набора данных – 8 715 образцов, 144 признака.

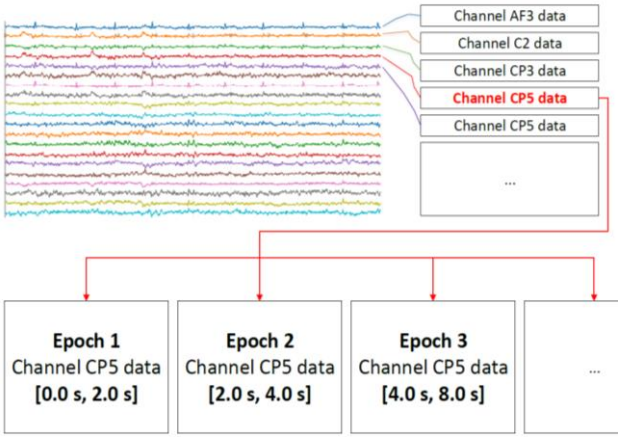


Рис. 1. Деление данных на эпохи

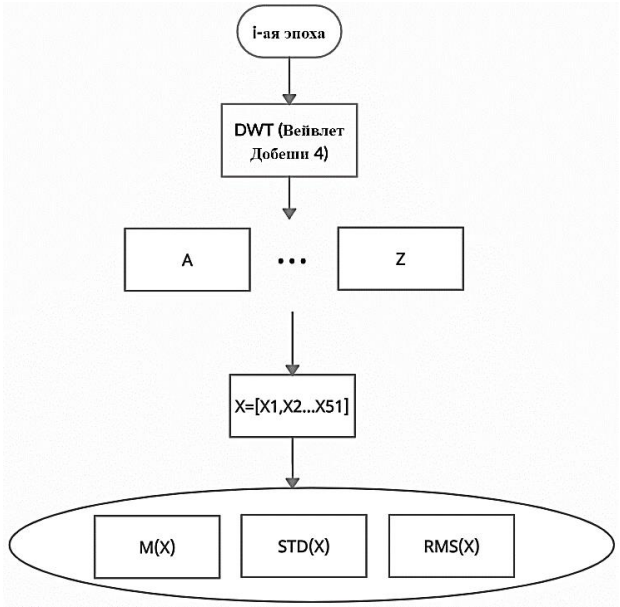


Рис. 2. Схема выделения признаков из сигнала ЭЭГ

На основе полученного набора данных были сгенерированы позитивные (владелец векторов один и тот же, метка класса – 1) и негативные (различные владельцы векторов, метка класса – 0) пары векторов признаков, суммарно 10 000 пар. Таким образом, задача аутентификации сводится к бинарной классификации. Каждая пара векторов также была заменена на вектор абсолютной разности их значений.

Нечеткий классификатор

Нечеткие классификаторы состоят из базы продукционных правил, оценивающих значение признаков объекта $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ нечеткими термами следующего вида:

R_j: ЕСЛИ $p_1 \wedge x_1 = T_{1j}$, И $p_2 \wedge x_2 = T_{2j}$, И …,
И $p_n \wedge x_n = T_{nj}$, ТО class = c_j,

где T_{ij} – нечеткое множество (лингвистический терм), характеризующее i-й признак в правиле R_j, $j = 1, \ldots, R$, R – количество правил; c_j – индекс класса j-го правила, p_i – бинарное значение, полученное в результате отбора признаков.

Процесс нечеткого вывода для входного вектора **x** (классификации вектора) выполняется следующим образом:

1) вычисление степени принадлежности вектора **x** к перечню термов в каждом правиле. Для j-го правила степень принадлежности $\mu_j(\mathbf{x})$ определяется формулой, представленной ниже:

$$\mu_j(\mathbf{x}) = \prod_{i=1}^{n} \mu_{T_{ij}}(x_i),$$

где $\mu_{T_{ij}}(x_i)$ – степень принадлежности признака x_i нечеткому терму T_{ij};

2) вычисление степени принадлежности входного вектора **x** каждому имеющемуся классу. Для k-го класса ($k = 1, \ldots, M$, M – число классов) степень уверенности $\beta_k(x)$ рассчитывается путем сложения степени принадлежности объекта правилам, консеквент которых соответствует k-му классу:

$$\beta_k(\mathbf{x}) = \sum_{(j|c_j=k)} \mu_j(\mathbf{x}).$$

3) для **x** определяется класс с максимальной уверенностью в принадлежности:

$$class = \arg \max_{k=1,\ldots,M} \{\beta_k(\mathbf{x})\}.$$

Методика построения нечеткого классификатора включает в себя несколько этапов – построение базы правил, отбор признаков, оптимизация параметров классификатора.

Наиболее простой способ построить начальную базу правил – алгоритм экстремальных значений признаков класса (АЭПК) [8]. В данной работе также использовались и другие методы генерации, в частности, с использованием алгоритмов K-Means и комбинации алгоритмов Autonomous Data Partitioning (ADP) вместе с K-Means из работы [9].

Также был применен метод генерации базы правил с использованием метаэвристической кластеризации. В этом случае генерировалась случайным образом популяция векторов $\mathbf{CS} = [\mathbf{CS}_1, \ldots, \mathbf{CS}_i, \ldots, \mathbf{CS}_{\text{size}}]$ (size – размер популяции, $|\mathbf{CS}_i| = R * n * pl$, pl – количество параметров выбранной функции принадлежности), каждый из которых представляет собой совокупность центроид-правил. Псевдокод генерации нечеткой системы приведен ниже.

Алгоритм 1. Генерация нечеткой системы с помощью метаэвристической кластеризации

Вход: таблица наблюдений $(\mathbf{X},\mathbf{Y})$, $\mathbf{X}=[\mathbf{x}_1,...,\mathbf{x}_i,...,\mathbf{x}_l]$, $\mathbf{Y}=[y_1,...,\ y_i\ ,...,y_l]$, *size, n, R, pl,* **params** – вектор параметров метаэвристического алгоритма, *type* – тип функции принадлежности, *metaheuristic* – функция, выполняющая оптимизацию метаэвристическим алгоритмом, *ffclust* – фитнес-функция.

Выход: $\mathbf{FS}_{best}$ – сгенерированная база правил.

1: $\mathbf{CS} \leftarrow generate(type, size, pl, R, n)$ – генерация популяции баз правил;

2: $\mathbf{CS} \leftarrow metaheuristic(\mathbf{params}, \mathbf{CS}, \mathbf{X}, \mathbf{Y}, ff)$

3: Выбор лучшей базы правил $\mathbf{CS}_{best}$ по вычисленному лучшему значению $ffclust(\mathbf{CS}_j)$

4: Генерация базы правил $\mathbf{FS}_{best}$ на основе вектора $\mathbf{CS}_{best}$

Алгоритм 2. Фитнес-функция *ffclust*

Вход: таблица наблюдений $(\mathbf{X},\mathbf{Y})$, $\mathbf{X}=[\mathbf{x}_1,...,\mathbf{x}_i,...,\mathbf{x}_l]$, $\mathbf{Y}=[y_1,...,\ y_i\ ,...,y_l]$, *size, n, R, pl, type,* $\mathbf{CS}_j=[\mathbf{c}_{j1},...,\mathbf{c}_{ji},...,\mathbf{c}_{jR}]$.

Выход: *ffvalue*.

1: Вычисление матрицы евклидовых расстояний $\mathbf{D}_{R\times l}=(d_{km})$ между центроидами-правилами $\mathbf{c}_{ji}$ и векторами данных $\mathbf{x}_i$ из таблицы наблюдений $\mathbf{X}$;

2: $Clusters \leftarrow \varnothing$

3: **ЦИКЛ** m=1..R

4: $Clusters \leftarrow Clusters \cup \varnothing$

5: **ЦИКЛ** m=1..l

6: $nearest \leftarrow \underset{k\in 1..R, m\in 1..l}{\arg\min_i} d_{km}$

7: $Cluster_{nearest} \leftarrow Cluster_{nearest} \cup \mathbf{x}_m$

8: $FS \leftarrow \varnothing$

8: **ЦИКЛ** m=1..R

10: **ЕСЛИ** $|Cluster_m| > 0$

11: Сгенерировать правило *Rule* на основе минимальных и максимальных значений признаков в $Cluster_m$

12: **ИНАЧЕ**

13: Сгенерировать правило *Rule* на основе минимальных и максимальных значений признаков в $\mathbf{X}$

14: $FS \leftarrow FS \cup Rule$

15: $ffvalue \leftarrow acc(FS, \mathbf{X}, \mathbf{Y})$

Отбор признаков производился с помощью фильтра на основе критерия взаимной информации, использовалось 60% лучших по значению критерия признаков согласно работе [10], а также на основе бинарного метаэвристического алгоритма Bird Swarm Optimization (BSO) [11] с числом итераций, равным 300. Для оптимизации параметров нечеткого классификатора также применялась непрерывная версия алгоритма BSO [12] с числом итераций, равным 500. Варианты построения нечеткого классификатора описаны в табл. 1.

Эксперимент

Для проверки обученных моделей была использована методика кроссвалидации (5 непересекающихся разбиений в соотношении 20% на тестовую выборку и 80% на обучающую). Процедура StratifiedKFold (с k = 5) из пакета scikit-learn [13] использовалась для реализации подобной кроссвалидации. Также использовалась модель машинного обучения LinearSVC из пакета scikit-learn для сравнения с нечетким классификатором.

Таблица 1

Перечень вариантов построения нечеткого классификатора

Обозначение	Генерация	*R*
EX-BS	АЭПК	2
ADP-KMEANS	ADP+Kmeans	20
BSO-GEN	Метаэвристическая кластеризация BSO	20

Исходя из результатов, приведенных в табл. 2, лучшим по точности классификации является метод на основе метаэвристической кластеризации. Он не только превосходит линейную машину опорных векторов по точности, но и более интерпретируем.

Таблица 2

Процент правильной классификации на тестовой выборке

Обозначение	*R*	Число признаков	Процент правильной классификации
EX-BS	2	101	75,94%
ADP-KMEANS	20	98	74,01%
BSO-GEN	20	98	79,37%
LinearSVC	-	86	78,85%

Однако полученный классификатор имеет недостатки в аспекте интерпретируемости – большое число правил и признаков, несмотря на то, что после отбора их количество сократилось на треть. Классификатор EX-BS может быть более предпочтителен для интерпретации, но также использует довольно значительное количество признаков.

Заключение

В данной работе предложено использование нечеткого классификатора, построенного методами кластеризации и оптимизированного метаэвристическим алгоритмом Bird Swarm. Для отбора признаков применялся бинарный метаэвристический алгоритм Bird Swarm, в качестве признаков использовались статистические характеристики, вычисленные на основе коэффициентов дискретного вейвлет-преобразования. Построенные классификаторы были протестированы на общедоступном наборе данных PhysioNet EEG Motor Movement/Imagery Dataset (PEEGMIMDB) с достижением точности классификации 79,37% нечеткой системой. Основными преимуществами предложенного подхода являются интерпретируемость итогового классификатора, присущая системам, работающим на основе нечетких правил.

Работа выполнена в рамках программы стратегического академического лидерства «Приоритет 2030».

Литература

1. Review A. Biometric Authentication / A. Review, S.R. Kodituwakku // International Journal of Trend in Research and Development. – 2015. – Vol. 2 (4). – PP. 113–123.

2. Gui Q. и др. A Survey on Brain Biometrics // ACM Comput. Surv. – 2019. – Vol. 51, No. 6. – PP. 1–38.

3. Lee Y.-Y., Hsieh S. Classifying Different Emotional States by Means of EEG-Based Functional Connectivity Patterns // PLoS ONE. – 2014. – Vol. 9, No. 4. – P. e95415.

4. Mason J. et al. An Investigation of Biometric Authentication in the Healthcare Environment // Array. – 2020. – Vol. 8. – P. 100042.

5. Sundararajan M., Taly A., Yan Q. Axiomatic Attribution for Deep Networks [Электронный ресурс]. – Режим доступа: https://arxiv.org/abs/1703.01365 (дата обращения: 22.08.2022).

6. Schalk G. et al. BCI2000: A General-Purpose Brain-Computer Interface (BCI) System // IEEE Trans. Biomed. Eng. – 2004. – Vol. 51, No. 6. – PP. 1034–1043.

7. Mallat S.G. A theory for multiresolution signal decomposition: the wavelet representation // IEEE Trans. Pattern Anal. Machine Intell. – 1989. – Vol. 11, No. 7. – PP. 674–693.

8. A Fuzzy Classifier with Feature Selection Based on the Gravitational Search Algorithm / M. Bardamova, A. Konev, I. Hodashinsky, A. Shelupanov // Symmetry. – 2018. – Vol. 10. – PP. 609.

9. Svetlakov M.O. Clustering-based rule generation methods for fuzzy classifier using Autonomous Data Partitioning algorithm / M.O. Svetlakov, I.A. Hodashinsky // Journal of Physics: Conference Series: 18, Tomsk, 27–30 apr. 2021. – Tomsk, 2021. – P. 012032. – DOI 10.1088/1742-6596/1989/1/012032.

10. Feature Selection Methods Comparison for EEG-based Classifier Constructed Using Discrete Wavelet Transform Features / Tsybenov B., Svetlakov M., Hodashinsky I. // J. Phys.: Conf. Ser. – 2022. – Vol. 2291, No. 1. – P. 012003.

11. Светлаков М.О. Отбор признаков для нечеткого классификатора с использованием метаэвристического алгоритма «bird swarm» // Сборник избр. статей научной сессии ТУСУР. – 2021. – Ч. 2. – С. 230–233.

12. Meng X.-B. et al. A new bio-inspired optimisation algorithm: Bird Swarm Algorithm // Journal of Experimental & Theoretical Artificial Intelligence. – 2015. – Vol. 28, No. 4. – PP. 673–687.

13. Pedregosa F. et al. Scikit-learn: Machine Learning in Python // arXiv. 2012.

Светлаков Михаил Олегович
М.н.с. лаб. съема, анализа и управления биологическими сигналами (ЛСАУБС) (ИСИБ) ТУСУРа
Эл. почта: svetlakov.m4@gmail.com

УДК 004.02

В.П. Турнаева

Отбор признаков с помощью метаэвристического алгоритма Rock Hyraxes Swarm / Алгоритм каменных даман

Рассматривается отбор информативных признаков с помощью метаэвристического алгоритма Rock Hyraxes Swarm / Алгоритма каменных даман. Эксперимент по оценке исследуемого алгоритма в задачи классификации с помощью нечёткого классификатора был проведен на наборах данных из репозитория KEEL, а также, было проведено статистическое сравнение с другими алгоритмами: Future Search Algorithm, Harris Hawks Optimization.
Ключевые слова: нечёткий классификатор, отбор признаков, метаэвристический алгоритм, Rock Hyraxes Swarm.

Задача классификации относится к области машинного обучения. Классифицировать объект – присвоить ему метку класса, к которому относится данный объект. На данный момент существует большое количество алгоритмов классификации – линейная и логистическая регрессии, метод k-средних, метод опорных векторов и многие другие.

В отличие от представленных методов нечеткий классификатор, благодаря своим не жестким границам между соседними классами, может определить объект к нескольким классам с разной степенью уверенности, что дает более точную картину об исследуемом объекте.

Выбор признаков как стратегия предварительной обработки данных доказал свою эффективность и действенность при подготовке данных для различных задач интеллектуального анализа данных и машинного обучения. Суть отбора признаков состоит в построении более простых и понятных моделей, улучшении интеллектуального анализа данных, повышении производительности. Таким образом, была поставлена цель: рассмотреть отбор признаков с помощью метаэвристического алгоритма Rock Hyraxes Swarm/Алгоритм каменных даман.

Нечёткий классификатор

По большей части объекты в реальной жизни не имеют точно определенного критерия принадлежности. Любой объект может быть отнесен как к одному, так и к другому классу. Например, к классу «животные» можно отнести следующие элементы: собаки, птицы, лошади, кошки и т.д., но очевидно, что данный класс исключает камни, деревья, реки и т.д. А вот, например, такой объект, как морская звезда, имеет уже более неопределенный класс по отношению к классу животных.

Для решения таких задач существует метод классификации объектов, цель которого состоит в том, чтобы сопоставить входному набору данных одну из меток класса. Существует довольно много методов классификации, в данной работе будет рассмотрен нечёткий классификатор.

Суть системы в том, что в консекветной части содержится метка – название класса. Его основная задача состоит в том, чтобы описать предполагаемый кластер нечетким прототипом, размерность которого задана размерностью пространства данных, которые будут исследоваться [1, 2].

Построение нечеткого классификатора подразумевает под собой решение ряда задач:

1. Отбор признаков.
2. Генерация базы нечетких правил.

Чаще правила строятся на основе алгоритма кластеризации.

3. Оптимизация параметров антецедентов.

На этапе оптимизации параметров антецедентов правил происходит более «тонкая» настройка, данный этап, как правило, выполняется с помощью роевых алгоритмов. Это необходимо для определение неизвестных параметров антецедентов и консеквентов нечетких правил путем оптимизации работы нечеткой системы по заданному критерию [3]. В данной работе для оптимизации параметров был использован Rock Hyraxes Algorithm/Алгоритм каменных даман.

Отбор признаков

Эффективность алгоритмов классификации существенно зависит от набора признаков, используемых для описания классифицируемых объектов.

Существует как минимум три причины отрицательного влияния большого количества неинформативных признаков на качество работы классификатора. Во-первых, при увеличении числа признаков уменьшается статистическая надежность работы алгоритма. Во-вторых, чем больше признаков, тем больше необходимо объектов обучения для того, чтобы была надежная классификация. И, в-третьих, чем больше признаков – тем больше время работы алгоритма [4].

Существует три метода отбора признаков:

1. Фильтр. Данный метод основан на выявлении обобщающих свойств данных и не ориентируются на конкретные алгоритмы интеллектуального анализа, основан на теории вероятностей и статистических подходах.

2. Обертка. Особенность этого метода – поиск всех возможных подмножеств признаков и оценка их качества путем «прогонки» через модель. Данный метод включается в процесс построения классификатора, поэтому он использует меру оценки эффективности классификатора для того, чтобы оценить отобранное множество.

3. Встроенный метод. Позволяет не разделять отбор признаков и обучение классификатора, а производит отбор внутри процесса расчета модели [7].

Рассмотрен отбор признаков по методу обертки. Поскольку задача отбора признаков может быть сформулирована как задача оптимизации, для ее решения было решено использовать метаэвристические методы поиска оптимального или субоптимального подмножества признаков. Однако для решения задачи отбора признаков алгоритмы, работающие в непрерывном пространстве поиска, необходимо адаптировать для работы в бинарном пространстве, сохранив при этом изначальную идею алгоритма.

Методы преобразования алгоритма к бинарному виду

Выделяют следующие методы:

1. Модифицированные алгебраические операции. Суть в том, что все арифметические операции заменяются на логические аналоги [8, 9]:

«+» заменяется на логическое «ИЛИ» / «OR»; «*» заменяется на логическое «И» / «AND»; «–» заменяется на исключающее «ИЛИ» / «XOR».

Пространство поиска меняется на бинарное, но основная идея алгоритма остается.

2. Трансформационные функции. Идея метода заключается в том, чтобы преобразовать непрерывное решение к бинарному с помощью трансформационной функции, которая показывает вероятность изменения отдельной позиции. Чаще всего применяются следующие функции [10]:

S-образная функция:

$$F_1(x) = \frac{1}{(1+e^{-x})}, \quad (1)$$

где x – некоторое входное вещественное число.

V-образная функция:

$$F_2(x) = \left|\tanh(x)\right|. \quad (2)$$

Еще один вид V-образной функции:

$$F_3(x) = \left|\frac{z}{\sqrt{1+x^2}}\right|. \quad (3)$$

3. Операция MERGE. Суть состоит в объединении двух бинарных векторов: при совпадении элементов – сохраняется, при несовпадении – случайно выбирается из несовпадающих. В целом, метод очень похож на модифицированные алгебраические операции, только в данном случае все операции заменяются на операцию MERGE [1, 11].

4. Комбинация методов модифицированных алгебраических операций и операции MERGE. Операция сложения заменяется на операцию MERGE, остальное – в соответствии с первым методом.

Эксперимент

Реализованный ранее алгоритм в задаче поиска минимума функции Rock Hyraxes Swarm / Алгоритм каменных даман [12] был преобразован в бинарный с применением вышеописанных методов.

Для тестирования были выбраны наборы данных из репозитория KEEL, представленные в табл. 1, где TestBefore – точность классификации до применения нечёткого классификатора.

Таблица 1

Наборы данных

Dataset	Признаки	Экземпляры	Классы	TestBefore
australian	14	690	2	49,13
ionosphere	33	351	2	79,76
movement_libras	90	360	15	48,06
optdigits	64	5620	10	10,02
page-blocks	10	5472	5	51,17
ring	20	7200	2	49,53
satimage	36	6435	7	60,36
segment	19	2310	7	80,48
sonar	60	208	2	57,24
spambase	57	4597	2	39,42
spectfheart	44	267	2	80,88
texture	40	5500	11	69,96
thyroid	21	7200	3	7,11
twonorm	20	7400	2	96,09
vehicle	18	846	4	29,9
wdbc	30	569	2	91,91
winequality-red	11	1599	11	19,7
yeast	8	1484	10	23,52

Результаты эксперимента

Результаты применения метаэвристического алгоритма Rock Hyraxes Swarm / Алгоритм каменных даман в задаче отбора признаков и классификации данных представлены в табл. 2, 3.

Таблица 2

Результаты точности классификации

Dataset	NoSelect	Operations	Merge	Opmerge	Sigm	Tanh	Vsharped
australian	73,33	72,03	72,75	75,07	71,88	73,04	71,01
ionosphere	79,19	82,64	83,21	80,66	80,66	84,07	82,06
movement_libras	48,89	15,56	9,44	9,72	9,72	7,78	9,17
optdigits	13,40	19,29	17,99	23,95	19,22	18,63	21,37
page-blocks	89,66	90,19	90,11	90,30	90,77	90,08	89,64
ring	67,01	61,47	63,68	63,51	63,03	61,69	65,49
satimage	60,47	37,64	42,11	39,88	40,30	39,16	39,39
segment	80,17	33,16	28,83	32,94	30,91	32,25	29,35
sonar	64,88	58,55	62,02	60,52	58,60	58,52	59,12
spambase	45,71	72,44	71,05	75,53	75,24	69,63	69,06
spectfheart	79,83	77,91	77,93	83,14	76,41	69,27	74,53
texture	70,31	21,85	20,45	22,56	21,87	19,20	19,84
thyroid	36,03	91,15	93,14	92,94	92,78	92,72	92,50
twonorm	96,09	70,99	74,05	71,53	70,14	70,84	71,70
vehicle	34,41	34,63	30,03	34,99	32,16	35,11	36,86
wdbc	92,62	85,23	80,13	84,69	77,32	81,71	82,60
winequality-red	45,34	40,83	42,40	42,53	40,84	40,59	41,03
yeast	31,61	30,66	29,25	34,39	29,12	29,78	29,72

Таблица 3

Количество отобранных признаков с помощью метаэвристического алгоритма при разных методах преобразования алгоритма

Dataset	NoSelect	Operations	Merge	Opmerge	Sigm	Tanh	Vsharped
australian	14	5,60	4,80	5,30	6,20	5,40	5,20
ionosphere	33	13,40	19,20	14,00	19,00	18,60	18,10
movement_libras	90	25,70	44,80	18,30	44,60	53,80	57,10
optdigits	64	21,10	33,00	12,80	31,80	32,20	32,20
page-blocks	10	2,10	3,70	2,30	4,10	0,40	0,40
ring	20	1,90	2,60	2,30	5,30	6,70	6,40
satimage	36	4,00	8,70	4,00	15,70	17,30	17,40
segment	19	5,50	8,80	5,60	9,20	10,60	11,00
sonar	60	15,80	30,30	5,70	31,50	37,80	36,30
spambase	57	15,60	19,50	4,00	21,00	21,80	22,80
spectfheart	44	19,40	24,20	16,40	24,70	32,90	35,90
texture	40	4,70	15,50	5,20	19,20	20,10	22,30
thyroid	21	4,00	8,20	5,40	7,80	8,50	10,30
twonorm	20	20,00	19,60	19,70	20,00	20,00	20,00
vehicle	18	10,10	6,10	4,40	8,10	9,50	8,10
wdbc	30	4,40	12,10	5,60	13,00	13,60	14,10
winequality-red	11	4,10	4,50	3,70	4,20	5,10	6,10
yeast	8	3,90	4,50	4,00	4,50	4,10	4,10

Сравнение с аналогами

Следующим этапом было проведено сравнение результатов с алгоритмами-аналогами: Future Search Algorithm (FSA), Harris Hawks Optimization (HHO), которые были реализованы коллегами в рамках научно-исследовательской работы. Тестирование проводилось на тех же 18 наборах данных с параметрами N=24, где N – количество особей в популяции, и T=250, где T – количество итераций. Эксперимент проводился по схеме десятикратной кроссвалидации, полученные результаты усреднялись.

Сравнение алгоритмов было сделано по критерию Фридмана – непараметрический аналог дисперсионного анализа. Итогом данного критерия является получение рангов для каждого из алгоритмов. В табл. 4. представлены результаты сравнения.

Таблица 4

Результаты сравнения алгоритмов

Метод	RHA		FSA		HHO	
	Точность	Ранг	Точность	Ранг	Точность	Ранг
Merge	53,80	2	57,36	1	52,32	3
Operations	54,20	2	56,50	1	50,30	3
Opmerge	55,37	2	57,52	1	51,65	3
Sigma	53,46	2	56,40	1	50,78	3
Tanh	53,16	2	57,16	1	49,78	3
Vsharped	53,69	2	57,17	1	50,03	3

По итогу данного сравнения можно сделать вывод, что алгоритм Rock Hyrax Algorithm / Алгоритм каменных даман показал себя не хуже своих аналогов и оказался по всем методам на втором месте.

После была проверена статистическая значимости с помощью критерия Уилкоксона для лучшего метода при отборе признаков:

– RHA – модифицированные алгебраические операции + операция MERGE;

– FSA – модифицированные алгебраические операции + операция MERGE;

– HHO – операция MERGE.

Была выдвинута нулевая гипотеза H_0 – статистическая значимость между результатами отсутствует. Было проведено попарное сравнение с уровнем значимости $\alpha = 0{,}05$, результаты представлены в табл. 5.

Таблица 5

Результаты статистической значимости

Сравнение	α	p-value	H_0
RHA – HHO	0,05	<0,001	Отклоняется
RHA – FSA	0,05	0,349	Принимается

Тогда можно сделать вывод, что алгоритм Rock Hyraxes Swarm показал результаты, близкие к алгоритму Future Search.

Заключение

С помощью реализованного алгоритма Rock Hyraxes Swarm / Алгоритм каменных даман получилось добиться увеличения точности, однако не на всех протестированных наборах. В некоторых случаях значения точности с оптимизацией параметров оказались лучше, чем при отборе признаков. Это может быть связано с тем, что в данных наборах убирались информативные признаки, либо признаки, которые по отдельности являются неинформативными, но в совокупности могли бы показать большую точность. Также это может быть связано с тем, что в некоторых наборах наблюдается большее количество экземпляров, что также может повлиять на точность классификации.

Литература

1. Ходашинский И.А. Отбор признаков и построение нечеткого классификатора на основе алгоритма прыгающих лягушек // Искусственный интеллект и принятие решений. – 2018. – № 1. – С. 76–84.

2. Горбунов И.В. Методы построения трехкритериальных Парето-оптимальных нечетких классификаторов / И.В. Горбунов, И.А. Ходашинский // Искусственный интеллект и принятие решений. – 2015. – № 2. – С. 75–87.

3. Ходашинский И.А., Мех М.А. Построение нечеткого классификатора на основе методов гармонического поиска // Программирование. – 2017. – № 1. – С. 54–65.

4. Цурко В.В. Метод контрастирования для отбора информативных признаков по эмпирическим данным / В.В. Цурко, А.И. Михальский // Автоматика и телемеханика. – 2016. – № 12. – С. 136–154.

5. Ходашинский И. А. Отбор признаков для нечеткого классификатора с использованием алгоритма паукообразных обезьян / И.А. Ходашинский, М.М. Немирович-Данченко, С.С. Самсонов // Бизнес-информатика. – 2019. – Т. 13, № 2. – С. 29–42.

6. Zhang P. Feature selection considering weighted relevancy / P. Zhang, W. Gao, G. Liu // Applied Intelligence. – 2018. – Vol. 48, No. 12. – PP. 4615–4625.

7. Бардамова М.Б. Алгоритмы построения нечетких классификаторов несбалансированных данных на основе метаэвристик «гравитационный поиск» и «прыгающие лягушки»: дис. … канд. техн. наук. – Томск, 2021. – 126 с.

8. Jia D. Binary artificial bee colony optimization using bitwise operation / D. Jia, X. Duan, M.K. Khan // Computers and Industrial Engineering. – 2014. – Vol. 76. – P. 360–365.

9. Kiran M.S. XOR-Based Artificial Bee Colony Algorithm for Binary Optimization // Turkish Journal of Electrical Engineering and Computer Sciences. – 2013. – Vol. 21. – PP. 2307–2328.

10. Ходашинский И.А. Модификации алгоритма прыгающих лягушек для отбора признаков в нечетком классификаторе при аутентификации пользователя по рукописной подписи / И.А. Ходашинский, М.Б. Бардамова // Информационные и математические технологии в науке и управлении. – 2020. – № 4 (20). – С. 75–83.

11. Papa J.P. Feature selection through binary brain storm optimization // Computers & Electrical Engineering. – 2018. – Vol. 72. – PP. 468–481.

12. Al-Khateeb B. Rock Hyraxes Swarm Optimization: A New Nature-Inspired Metaheuristic Optimization Algorithm // CMC-computers materials & Continua. – 2021. – Vol. 68, No. 1. – PP. 643–654.

Турнаева Вера Павловна
Студентка каф. БИС, ТУСУРа
Эл. почта: turnaevavera@gmail.com

УДК 004.8

В.С. Ваняшин

Анализ существующих актуальных библиотек для выполнения задачи распознавания рукописного ввода

Анализируются сведения и информация об актуальных библиотеках и их использование в целях выполнения задач распознавания рукописного ввода.

Ключевые слова: машинное обучение, рукописный ввод, библиотека, TensorFlow, PyTorch, Keras, Scikit-Learn.

В последнее время машинное обучение и наука о данных стали очень популярными в современном обществе. Постоянно появляются новые библиотеки, и для обучения моделей машинного обучения может потребоваться довольно много кода. В такой ситуации не должны забывать, что машинное обучение – это не самоцель, а инструмент для решения проблемы. Недостаточно создать рабочую модель; не менее важно качественно представить результаты анализа или создать рабочий продукт.

В статье обновляется тема машинного обучения. В результате будет получен анализ, какая библиотека лучше всего подходит для выполнения задач распознавания рукописного ввода. Полученные данные будут представлены в виде дисперсии и точности каждой библиотеки в этой задаче машинного обучения.

В современном обществе машинное обучение в компьютерной сфере считается активно развивающейся областью научных исследований. Развитие обусловлено тем, что работа с данными приобрела более доступный характер, а новые методы взаимодействия различных законов в научных сферах позволили ускорить данный процесс. Благодаря этому в сложных научных задачах и случилось непосредственное внедрение глубокого машинного обучения для решения и раскрытия различных законов.

Глубокое машинное обучение взаимодействует с различными способами моделирования сложных абстракций при помощи трансформаций, представляющих собой нелинейную последовательность данных, преобразующих искусственные нейронные сети [1].

На данный момент развития технологий, различные нейросети используются для решения разного рода задач, таких как прогнозирование, распознавание визуальных образов и проч.

Достаточное количество новых инструментов глубокого машинного обучения было выявлено в последнее время, что способствует осознанию актуальности данной тематики развития [2]. Это и привело к данному сравнительному анализу актуальных библиотек, используемых для машинного обучения. Взаимодействие нейронных сетей происходит за счет многих аспектов, а не только из-за уникальных библиотек, также важнейшим аспектом являются язык программирования и различные дополнительные расширения, относящиеся к нему, для доступного старта нейросетей и ее моделей. Все перечисленное обладает обширным и сложным функционалом, тем самым требуя от пользователя навыков, способных преодолеть трудности на этапе подготовки системы и ее поддержки. Тем самым важным аспектом является выбор нужной библиотеки, для того чтобы достичь нужного результата, не потеряв важных ресурсов.

Основные рассматриваемые библиотеки, о которых пойдет речь и будут анализироваться некие особенности, следующие: TensorFlow, PyTorch, Keras и Scikit-Learn. Стандартные настройки и возможности будут приоритетными в данном анализе. Очевидным и логичным решением будет осмотр скорости работы и качества каждой библиотеки в задаче с рукописным вводом цифр. При конструировании была использована выборка, основанная на MNIST, для более корректной проверки данных библиотек и оформления их статистик.

Рабочие данные для библиотек при помощи MNIST

Для работы с данными библиотеками было использовано изображение, основанное на базе рукописных цифр MNIST. Данная база является набором образцов рукописного написания цифр. Изначальный вариант является официальным и общепринятым стандартом, признанным в США с целью взаимодействия средств машинного обучения и имеющихся данных для их последующей работы, основанной на нейронной сети. Так, была построена переделанная база с уже имеющимися изображениями.

Рис. 1. Пример изображения для обучения сети

Стандартные данные изображения имеют разрешение 28×28 и преобразованы в виде оттенков серого с последующим форматированием их в виде данных. Все данные в виде чисел отцентрированы на изображении. Этот пул данных поделен на две части, а именно, тренировочную, состоящую из 50 000 изображений, и тестовую – 10 000 изображений.

Выбор библиотек для решения задач

На данный момент представлено большое количество программных средств для решения тех или иных задач в области глубокого обучения. Всегда имеется возможность выбрать общее, объединить и уже после сравнить различный программный потенциал рассматриваемых сред.

На сегодняшний день многие разработчики предлагают такой подход к разработке нейронных сетей, что пользователь может себя ни в чем не ограничивать по типу создания самых различных методов машинного, а также глубокого обучения. Возможность основания нейронных сетей – это лишь малая доля потенциала рассматриваемых библиотек в современном мире. Но даже имея такой пласт потенциала, находятся индивидуумы в области машинного обучения, не способные предложить те функции, которые уже стали классическим набором при работе с глубоким обучением. Главной причиной данной проблемы является желание добиться большей производительности за счет уменьшения набора функционала [3].

Библиотеки для распознавания рукописного ввода рассматривались по различным критериям и подробно изучались для выявления сильных и слабых сторон. При изучении был осмотрен такой параметр, как справочная информация по библиотеке. Этим критерием начинался каждый анализ библиотеки, так как для ознакомления следует использовать справочники и руководства, представленные официально разработчиками данных библиотек.

Также были изучены их возможности функционального плана. Этот критерий сразу же даёт признак отсева при выборе библиотеки, так как, несмотря на различные плюсы и минусы, представленные разработчиками, сухие данные и возможности говорят гораздо больше, тем самым мотивируя остановить свой взгляд на том или ином представителе [4].

В данном анализе одним из важнейших критериев была операционная система. Несмотря на популярность в программном мире разработчиков операционной системы Linux, в данном анализе преимущество было у Windows, так как на этом и основывалась ключевая задача данного анализа. Язык же программирования не имел такого важного значения, но все равно тот же язык Python, на котором уже долгие годы люди развертывают различные нейронные сети, имеет статусное преимущество перед другими языками программирования [5].

Изучение тестовых конфигураций сетей также склонило к определенному вые попала библиотека Scikit-Learn, самая старая библиотека из выбранных на анализ. Несмотря на это, она все еще получает обновления, работает на языке программирования Python и поддерживает кроссплатформу среди операционных систем.

Библиотеки для работы с данными

Представленные далее библиотеки в том или ином исполнении похожи друг на друга. Пересечения происходят в сравнении языков программирования, везде наблюдается кроссплатформенность фреймворков, все это заслуга настоящего времени и развития современного общества в сфере программирования. Но при более детальном разборе каждой системы становится очевидным, что каждый из представленных кандидатов имеет те или иные критические преимущества или недостатки, с которыми в будущем следует столкнуться каждому разработчику той или иной системы.

Библиотека Scikit-Learn, вышедшая в 2007 г., несмотря на свой возраст, получает актуальные обновления, так, последнее вышло в августе 2022 г. Благодаря всеобъемлющему потенциалу язык программирования Python считается отличным, а для многих и самым лучшим фреймворком для работы с данными в сфере искусственного интеллекта: поддерживает разработку кластеров и различных моделей классификаций, умеет уменьшать размеры моделей. За восемь лет на GitHub имеет отличные показатели популярности [6]. Более 2,4 тыс. контрибьюторов и 51 тыс. звезд в системе.

Зависимая библиотека Keras, изначально основанная на Theano, а теперь уже полноценно рабочая через фреймворк TensorFlow, базируется на языке программирования Python и использует модуль Starlark для автоматизации создания и тестирования созданных систем. Вышедшая в 2015 г,. она имеет ряд полезных функций по типу комфортного и быстрого препроцессинга различного рода текстов и вложений самых разных типов данных. За пять лет GitHub уже получил 40 обновлений и пользуется большой популярностью в сфере машинного обучения, даже на сложных задачах, несмотря на свою простоту [7]. Более 1 тыс. контрибьюторов и 56 тыс. звезд в системе.

Библиотека, или полноценная среда машинного обучения PyTorch, вышедшая в 2016 г., несмотря на основу в лице языка программирования Python, имеет глубокую интеграцию с кодом C++, тем самым при активации нужного расширения есть возможность многоязыкового программирования. Следует выделить большие функции, а именно расчет тензора и возможность развертывания сетей, построенных на ленточной автоградной системе. За шестилетний период GitHub уже получил более 40 обновлений [8]. Более 2,4 тыс. контрибьюторов и 59 тыс. звезд в системе.

Самая популярная библиотека и по совместительству полноценный фреймворк для работы с задачами глубокого обучения TensorFlow, разработанный командой Google Brain в 2015 г., по сей день представляет собой мощнейший инструмент в индустрии. Как и PyTorch, имеет API для взаимодействия и программирования на C и C++, но в сравнении имеет куда более гибкую систему взаимодействия между ними. Имеется возможность запуска модели на мобильном устройстве. За счет мощнейшего включения в разработку энтузиастов имеет огромнейший потенциал для развития. За шестилетний период GitHub уже получил более 176 обновлений

[9]. Более 3,2 тыс. контрибьюторов и 168 тыс. звезд в системе, что является рекордом среди конкурентов.

В итоге данные библиотеки и стали основой анализа, после которого появилась возможность приступить к дальнейшему получению данных и сравнению их в задаче считывания рукописного ввода. Данные библиотеки представлены в таблицах для полного конкретного понимания преимуществ при решении поставленной задачи.

Анализ библиотек для работы с данными

Рассматривая статистические данные, различные спичи специалистов и многие информационные ресурсы, была основана следующая выборка библиотек, с которыми и проведен анализ [10]. При анализе было обращено внимания на технические возможности библиотек для того, чтобы сформировать первичное понимание каждой из систем (табл. 1). Благодаря этому можно в виде таблицы представить каждую из них для последующего взаимодействия с данными.

Для начала было проведено сравнение каждой библиотеки по четырем критериям. Во-первых, полностью подключенная нейронная сеть (FCNN) работает на плоском входе, где каждый вход подключен ко всем нейронам [11]. Если таковые имеются, уровни FC обычно находятся ближе к концу архитектур CNN и могут использоваться для оптимизации таких целей, как оценки в классе.

Второй критерий – сверточная нейронная сеть (CNN). Это класс искусственных нейронных сетей, наиболее часто применяемых для анализа визуальных образов. CNN также известны как инвариантные к сдвигу или пространственно-инвариантные искусственные нейронные сети, основанные на архитектуре ядер свертки с общим весом или фильтром, которые скользят по входным объектам и обеспечивают переводно-эквивариантные ответы, известные как карты объектов.

Третий критерий – автоэнкодер (AE). Это особый вид неконтролируемой искусственной нейронной сети, которая обеспечивает сжатие и другие функциональные возможности в области машинного обучения. Конкретное применение автоэнкодера заключается в использовании прямого подхода для восстановления выходных данных из входных.

И последнее – ограниченная машина Больцмана (RBM). Это генеративная стохастическая искусственная нейронная сеть, которая может изучать распределение вероятностей по своему набору входных данных. RBM – это вариант машин Больцмана с ограничением, заключающимся в том, что их нейроны должны образовывать двудольный граф: пара узлов из каждой из двух групп единиц, обычно называемых «видимыми» и «скрытыми» единицами соответственно, может иметь симметричную связь между ними; и между ними нет никаких связей узлов внутри группы.

В результате выделена важнейшая характеристика систем машинного обучения, а именно точность (табл. 2). Они будут распределены в соответствии с критериями. Первый – CNN, а второй – многослойный персептрон (MLP) – это искусственная нейронная сеть с прямой связью, которая генерирует набор выходных данных из набора входных данных. MLP характеризуется несколькими слоями входных узлов, соединенных в виде ориентированного графа между входным и выходным слоями. MLP использует обратное распространение для обучения сети.

Таблица 1

Технологические особенности выбранных библиотек для глубокого обучения

№	Название	Язык	ОС	FCNN	CNN	AE	RBM
1	Tensor-Flow	Python, C++	Windows, Linux, MacOS, JavaScript, Android	+	+	+	+
2	PyTorch	Python, C++	Windows, Linux, MacOS	+	+	+	+
3	Keras	Python	Windows, Linux	+	+	+	+
4	Scikit-Learn	Python, C++	Windows, Linux	+	+	+	+

Таблица 2

Результаты разбора библиотек, выраженные в точности MLP и CNN

Название	Точность MLP, %	Точность CNN, %
TensorFlow	97,1	97,4
PyTorch	96,9	98,1
Keras	95,1	96,2
Scikit-Learn	95,3	95,1

Несмотря на преимущество со стороны популярности библиотеки TensorFlow, есть моменты, где она не смогла показать результат выше, чем могло бы иметься на самом деле. В целом ровные данные, показывают, что результат един. Несмотря на это, популярное значение в 95% точности у Scikit-Learn наблюдается и в подобной задаче. Но в любом случае имеются преимущественные стороны у TensorFlow и PyTorch как со стороны практичной популярности и поддержки, так и в целом состояние точности на подобной и простой задаче говорит о библиотеке достаточно много и уже можно показать, как она ведет себя в различных задачах.

Заключение

В итоге следует выделить, что в современном обществе глубокое машинное обучение в сфере компьютерных технологий постоянно развивается в области научных исследований. Так всегда нужно использовать развивающиеся и актуальные средства в области нейронных сетей.

Тем самым был произведен сравнительный анализ четырех известных на сегодняшний день библиотек для решения задач рукописного ввода чисел, благодаря которому были получены данные.

Также следует отметить, что библиотеки, представляющие собой кроссплатформенный фреймворк, уже вовсю используются сотнями тысяч различных разработчиков во многих сферах машинного

обучения. Представленные результаты показывают, что, несмотря на популярность операционной системы Linux в программном мире, данные библиотеки отлично показали свою работу с операционной системой Windows.

Следующим этапом потенциально станет возможность использования одной из изученных библиотек в задаче программирования модуля для самостоятельного управления мышью в сложных задачах.

Литература

1. Сравнение библиотек глубокого обучения на примере задачи классификации рукописных цифр [Электронный ресурс]. – Режим доступа: https://habr.com/ru/company/intel/blog/254747/, свободный (дата обращения: 20.09.2022).

2. Бовырин А.В. Образовательный курс «разработка мультимедийных приложений с использованием библиотек OPENCV и IPP» / А.В. Бовырин, П.Н. Дружков, В.Л. Ерухимов, Н.Ю. Золотых, К.В. Корняков, В.Д. Кустикова, И.Д. Лысенков, И.Б. Мееров, В.Н. Писаревский, А.Н. Половинкин, А.В. Сысоев // Матер. XIV междунар. конф. «Высокопроизводительные параллельные вычисления на кластерных системах», г. Пермь, 10–12 ноября 2014 г. – 2014. – Т. 26, вып. 1. – С. 9–15.

3. Кустикова В.Д. Сравнение некоторых методов глубокого обучения для решения задачи классификции автомобилей на изображениях / В.Д. Кустикова, Д.А. Кручинин, П.Н. Дружков // Матер. 26-й междунар. конф. GraphiCon2016, Н. Новгород: ННГАСУ, 2016. – 2016. – Т. 1, вып. 1. – С. 424–429

4. Педро Д. Главный алгоритм: как поиски совершенной обучающей машины изменят наш мир. – 1-е изд. – М.: Манн, Иванов и Фербер, 2016. – 333 с.

5. Hastie T. The Elements of Statistical Learning / T. Hastie, R. Tibshirani, J. Friedman. – Second Edition. – Springer, 2017. – 653 p.

6. Scikit-Learn GitHub [Электронный ресурс]. – Режим доступа: https://github.com/scikit-learn/scikit-learn, свободный (дата обращения: 20.09.2022).

7. Keras GitHub [Электронный ресурс]. – Режим доступа: https://github.com/keras-team/keras, свободный (дата обращения: 20.09.2022).

8. PyTorch GitHub [Электронный ресурс]. – Режим доступа: https://github.com/pytorch/pytorch/, свободный (дата обращения: 20.09.2022).

9. TensorFlow GitHub [Электронный ресурс]. – Режим доступа: https://github.com/tensorflow/tensorflow, свободный (дата обращения: 20.09.2022).

10. Бурков А. Машинное обучение без лишних слов. – 1-е изд.. – СПб.: И.Д. Питер, 2020. – 188 с.

11. Митчел Т.М. Машинное обучениел. – 1-е. изд. – СПб.: И.Д. Питер, 1997. – 432 с.

Ваняшин Владислав Сергеевич
Аспирант, инженер каф. экономической математики, информатики и статистики (ЭМИС) ТУСУРа
Эл. почта: mr.vaniaschin@yandex.ru

УДК 004.31

Д.А. Жуков, М.Ю. Катаев

Разработка программы анализа естественно-языковых ответов в программных системах тестирования

Представлены диаграммы программной библиотеки, выполняющей функции анализа естественно-языковых ответов в программных системах тестирования, на языке UML, а также их обзор.
Ключевые слова: обработка естественного языка, смысл текста, семантика, семантический анализ, системы тестирования.

Актуальность данной работы обусловлена все возрастающей ролью компьютеризации сферы образования, а также автоматизацией различных задач процесса обучения. Среди всех направлений процесса обучения важными являются разработка и применение систем контроля знаний, которые связаны с отслеживанием уровня усвоения обучающимся знаний через мониторинг качества обучения (оценок).

Одним из вариантов развития образования является внедрение информационных технологий в учебный процесс, что способствует скорости, надежности, точности решения разнообразных задач обучения, в том числе и контроля знаний. Программы тестирования применяются в различных направлениях человеческой деятельности. Качество проверки ответов на тестовые вопросы в большой степени зависит от вариантов фиксации ответов в виде использования знаков (например, «+»), подчеркивания или текста. Для отвечающего наиболее естественной формой ответа является запись текста, смысл которого связан с вопросом. Для проверяющего ответы не составляет труда воспринимать такие ответы и оценивать их на основе соответствующих знаний. Проблемной стороной проверки тестов является их количество или число тестовых вопросов, что заставляет думать об автоматизированном анализе текстов ответов.

Целью работы является разработать программу для распознавания информации, содержащейся в тексте. и представления функций программы в качестве программной библиотеки. Для достижения поставленной цели была разработана архитектура, представленная на рис. 1. Важным является обеспечение интегрирования функции распознавания смысла текста в сторонние программы, которые выполняют решение задач тестирования или других приложений, где необходимо понимание смысла текста.

Для реализации программы анализа естественно-языковых ответов в программных системах тестирования предлагается реализовать программную библиотеку, направленную на решение общих проблем области естественной обработки языка.

Основные требования к такой библиотеке: простота использования и возможность реализации задач, требуемых для создания системы анализа ответов.

Программа состоит из четырёх основных модулей: машинный перевод, анализ тональности, реферирование текста и поисковая система. Такой подход позволяет многократно упростить разработку и сопровождение системы путём выделения основных типов задач в обособленные модули программной библиотеки, которые скрывают большую часть тонкостей программной реализации, оставляя пользователю лишь самые необходимые функции через программный интерфейс [1–3].

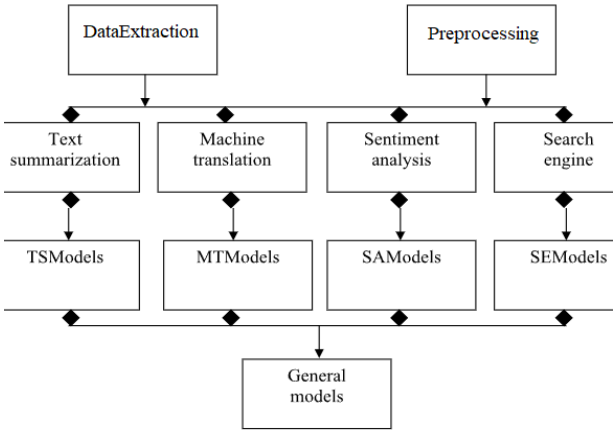


Рис. 1. Диаграмма классов архитектуры программной библиотеки

Машинный перевод

Функционал модуля машинного перевода, несмотря на его название, не ограничивается лишь переводом текста с одного естественного языка на другой. Данный класс может быть использован для разного рода задач, предполагающих трансляцию текста одного типа в соответствующий текст другого типа. Например, это может быть вопросно-ответная система, которая используется в чат ботах или системах автоматического тестирования. Также данный модуль можно использовать для реализации алгоритма трансляции естественного языка в машинный код и т.п. Гибкий функционал класса, однако, предполагает меньший уровень специализации, а следовательно, менее точные результаты при эксклюзивном использовании. Для устранения этой проблемы следует обработать результат работы данного модуля с использованием других классов для достижения более высокой точности на узкоспециализированной задаче.

Основной механизм модуля машинного перевода реализован с помощью encoder-decoder модели, изображённой на рис. 2 [4].

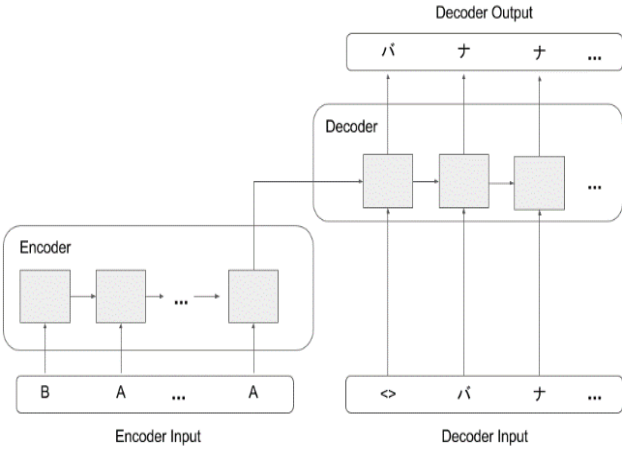


Рис. 2. encoder-decoder архитектура

В распространённых задачах трансляции одной последовательности символов или слов в другую входная и выходная последовательность имеют разную длину. Стандартный подход к решению проблем подобного типа – это архитектура encoder-decoder, состоящая из двух главных компонентов: кодировщик, который берёт последовательность произвольной длины в качестве входных данных, и декодировщик, который действует как условная языковая модель, предсказывая последовательный токен в исходном предложении на основе закодированных входных данных и контекста исходного предложения. Данная архитектура была реализована с помощью рекуррентных нейронных сетей, архитектура которых изображена на рис. 3, с механизмом внимания и трансформеров [5].

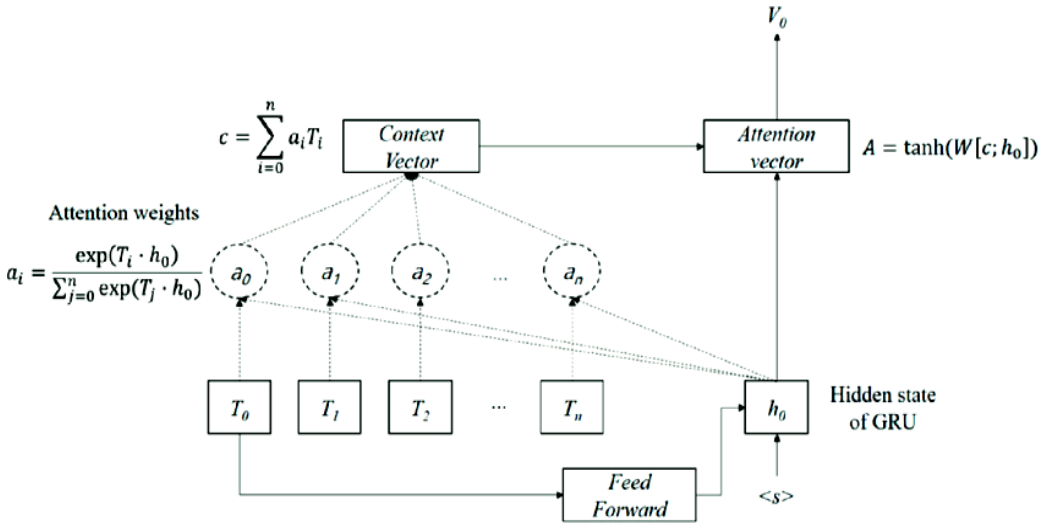


Рис. 3. Рекуррентная нейронная сеть с механизмом внимания

Механизм внимания обучает множество весов внимания, которые отображают отношение между закодированным вектором и скрытыми нейронами декодера, для того чтобы сгенерировать вектор контекста путём взвешивания суммы всех скрытых состояний кодировщика. Таким образом, декодировщик будет иметь доступ ко всей входной последовательности с фокусом на наиболее релевантную информацию при генерации выходных данных.

Статья «Attention is all you need», опубликованная в 2017 г., представила encoder-decoder архитектуру, основанную на слоях внимания, под названием трансформеры [6], представленную на рис. 4. Одним из главных отличий является то, что входная последовательность может быть обработана параллельно так, что графический процессор будет использован куда более эффективно, и скорость обучения возрастёт.

Тестирование модели производилось на специально собранном небольшом наборе данных. На рис. 5 изображено, как модели удалось правильно перевести предложение, содержащееся в тренировочной выборке, что является достаточным условием для утверждения о её корректной работе.

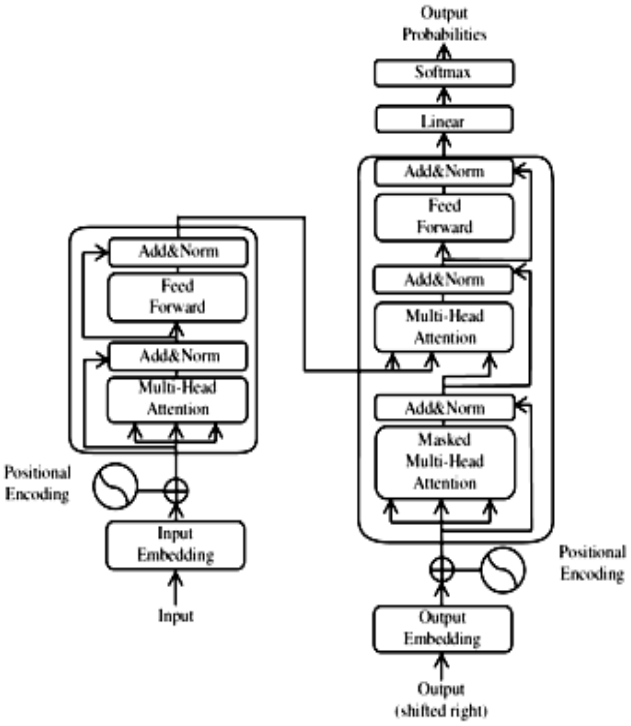


Рис. 4. Архитектура трансформера

```
1/1 [==============================] - 0s 332ms/step - loss: 0.
[['my', 'phone', 'is', 'laying', 'on', 'the', 'table']]
[[['my', 'phone', 'is', 'laying', 'on', 'the', 'table']]]
Training finished!
```

Рис. 5. Результат работы модуля Machine Translation

Анализ тональности текста

Модуль анализа тональности отвечает за круг задач, направленных на определение эмоционального тона отрывка текста. Для организаций это популярный способ определить и категорировать мнения о продукте, сервисе или идее. Данный подход предполагает использование сборов данных (data mining), машинного обучения и искусственного интеллекта для извлечения из текста настроения и субъективной информации.

Системы анализа тональности помогают организациям получить полезную информацию из неорганизованных и неструктурированных данных, когда речь идёт об онлайн-сервисах вроде электронной почты, интернет-блогов, веб-чатов, социальных сетей, форумов, комментариев и т.д. Алгоритм заменяет ручную обработку данных, применяя основанные на правилах, автоматические или гибридные методы. Методы, основанные на правилах, выполняют анализ тональности исходя из предопределённых, основанных на лексиконе, правил, в то время как автоматические системы обучаются с помощью алгоритмов машинного обучения.

Модуль анализа тональности предполагается для использования при создании автоматических систем с применением нейронных сетей, в частности рекуррентных нейронных сетей, а также сетей прямого распространения.

Тестирование модуля производилось на стандартном наборе данных, содержащем отзывы на американскую лётную компанию в Twitter, часто использующемся для целей тестирования, впервые опубликованном на Crowdflower's Data for Everyone library [7]. Модулю удалось достичь точности в 84% при относительной простоте используемой модели и малых затратах вычислительных мощностей, что является допустимым результатом для соответствующей архитектуры.

Для работы с текстовой частью, вводится функция суммаризации, которая связана с получением краткого содержания анализируемого текста, имеющего лишь его основной смысл. Существует два основных типа суммаризации: абстрагирующая и экстрактивная. Абстрагирующие методы выбирают слова, основываясь на семантическом понимании, в том числе слова, которые даже не появлялись в исходном документе. Они интерпретируют и исследуют текст, используя продвинутые техники обработки естественного языка для того, чтобы сгенерировать новый, более короткий текст, содержащий наиболее важную информацию из оригинального текста.

Данный метод похож на то, как люди читают статью и пересказывают основное содержание её своими словами.

Экстрактивные методы предпринимают попытку сгенерировать краткое содержание текста путём выбора наиболее значимого подмножества слов. Этот подход взвешивает наиболее важные части предложений и использует их для формирования резюме текста. Экстрактивные методы зачастую дают лучшие результаты при практическом применении по сравнению с абстрагирующими методами. Это является следствием того, что абстрагирующие методы вынуждены решать такие проблемы, как семантическая репрезентация и генерация естественного языка, которые являются более сложными относительно подходов, основанных на данных вроде вычленения предложений.

По этой причине модуль суммаризации текста использует исключительно экстрактивные методы. В их основе лежит pagerank алгоритм, изображённый на рис. 6, различные модификации которого современные браузеры используют для поиска наиболее релевантной веб-страницы [8].

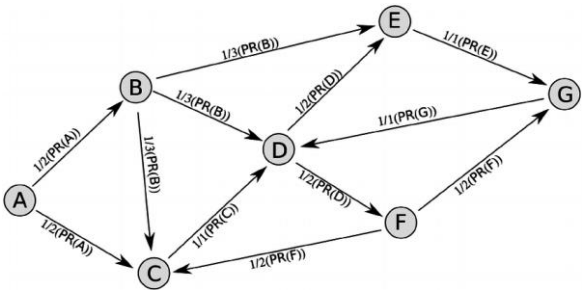


Рис. 6. Pagerank алгоритм

Тестирование модели производилось на статье [9], где результатом работы алгоритма стала разработка направления «Intelligent Cloud Hub», в рамках которого студентам будет обеспечен доступ к облачным сервисам и сервисам ИИ. Кроме этого, будут предоставлены средства разработки ИИ и программные сервисы Azure AI, такие как Microsoft Cognitive Services, Bot Services и Azure Machine Learning [https://azure.microsoft.com/en-us/products/machine-learning], что позволит выполнять сравнительный анализ с разрабатываемой моделью и анализ основного содержания текста.

Поисковая система

Модуль поисковой системы выполняет функции общего назначения и позволяет вычленять из текста нужную пользователю информацию, основываясь на различных направлениях лингвистического анализа. Данный модуль использует как детерминированные алгоритмы вроде регулярных выражений для вычленения информации по шаблону, так и алгоритмы машинного обучения для поиска, основанного на семантическом анализе.

Архитектура

Каждый модуль программной библиотеки содержит соответствующий класс, реализующий модели для данной задачи, который, в свою очередь, содержит класс, реализующий модели общего назначения, которые могут быть использованы при

конструировании более специализированных архитектур. Также каждый модуль содержит в себе класс предобработки текста, позволяющий привести его в подходящий вид для использования выбранной модели.

Важным последствием стремления к удобству также является модуль DataExtraction. Для продвинутого пользователя такой класс являлся бы практически полностью бесполезным, так как значительно сокращает количество опций в обмен на незначительное ускорение времени разработки, однако для пользователя, навыки которого соответствуют заявленному минимуму для использования системы, такой класс поможет сэкономить огромное количество времени, ведь он сведёт к минимуму необходимость самостоятельной загрузки данных и приведения их к виду, пригодному для дальнейшей их обработки, а также поможет найти подходящие данные во внутренней базе библиотеки и сети Интернет, избавив пользователя от необходимости изучения методов работы с выборками данных.

Важной частью модуля DataExtraction является нахождение и загрузка необходимых пользователю датасетов из сети Интернет (load_data). Одним из важнейших нефункциональных требований являются скорость и эффективность.

Обработка больших объёмов текста и обучение нейронных сетей требуют больших вычислительных мощностей для достижения высокой эффективности. Поэтому система, в первую очередь, разрабатывается для организаций, имеющих доступ к передовым графическим процессорам компании NVIDIA, так как на данный момент она предоставляет лучшие доступные средства для работы с нейронными сетями, а также является самой популярной в России, Европе и США компанией по изготовлению видеокарт. В связи с этим все модели, существующие в модулях GenralModels, TSmodel, MTModels, SAModels и SEModels, обязаны соответствовать стандарту технологии CUDA, позволяющей максимально эффективно использовать вычислительные мощности графических процессоров NVIDIA последнего поколения при работе с нейронными сетями [10].

Для пользователей, не располагающих большими вычислительными мощностями, также предусмотрена опция в виде менее эффективных, но намного более быстрых моделей, которые способны работать на бюджетных видеокартах.

Заключение

Таким образом, на данный момент разработана архитектура программной библиотеки распознавания смысла текста, используемой для анализа естественно-языковых ответов в программных системах тестирования, а также частично реализованы модули машинного перевода, анализа тональности и суммаризации текста, результаты работы которых были представлены ранее.

Литература

1. Jurafsky D., Martin J.H. Speech and Language Processing. – Pearson Prentice Hall, 2008. – 623 p.

2. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. – Pearson Prentice Hall, 2008. – 229 p.

3. Eisenstein J. Introduction to Natural Language Processing (Adaptive Computation and Machine Learning series). – MIT Press Ltd., 2019.– 536 p.

4. Deep Learning for Natural Language Processing: Develop Deep Learning Models for Natural Language in Python. – Brownlee J., 2017. – 414 p.

5. Bird S., Klein E., Loper E. Natural Language Processing with Python. – O'Reilly Media, 2009. – 491 p.

6. Ashish V., Noam S., Niki P., Jakob N., Llion J., Aidan N.G., Polosukhin, I. Attention Is All You Need // 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 2017. – P. 1–11.

7. Crowdflower's Data for Everyone library [Электронный ресурс]. – Режим доступа: https://appen.com/prelabeled-datasets/, свободный (дата обращения: 16.09.2022).

8. Altman A., Tennenholtz M. Ranking Systems: The PageRank Axioms // Proceedings of the 6th ACM conference on Electronic commerce (EC-05). – Vancouver, BC. Retrieved 29 September 2014. – P.1-8.

9. Microsoft Launches Intelligent Cloud Hub To Upskill Students In AI & Cloud Technologies [Электронный ресурс]. – Режим доступа: https://analyticsindiamag.com/microsoft-launches-intelligent-cloud-hub-to-upskill-students-in-ai-cloud-technologies/, свободный (дата обращения: 16.09.2022).

10. Cuda [Электронный ресурс]. – Режим доступа: https://developer.nvidia.com/cuda-toolkit/, свободный (дата обращения: 16.09.2022).

Жуков Даниил Андреевич
Студент каф. АСУ ТУСУРа
Эл. почта: zhukovda01@gmail.com

Катаев Михаил Юрьевич
Профессор каф. АСУ ТУСУРа
Эл. почта: kmy@asu.tusur.ru

Секция 21

РОБОТОТЕХНИКА И МЕХАТРОНИКА

Председатели секции – ***Шурыгин Юрий Алексеевич****, директор департамента управления и стратегического развития ТУСУРа, зав. каф. КСУП, д.т.н., профессор;*
Коцубинский Владислав Петрович*, доцент каф. КСУП, к.т.н.*

УДК 658.012.011.56

И.Н. Булатникова, Н.Н. Гершунина, Е.С. Яковлева

Автоматизация проектирования алгоритмов для микропроцессорных систем

На основе разностно-итерационных алгоритмов типа алгоритмов (шаг за шагом) Вольдера и Меджита предложена методология построения целочисленных алгоритмов вычисления практически любых функциональных зависимостей, заданных как аналитически, так и таблично. Для удобства проектирования таких алгоритмов предложена, математически обоснована и программно реализована САПР алгоритмов для микропроцессорных систем.

Ключевые слова: разностно-итерационные алгоритмы, микропроцессорные системы, проектирование.

С появлением микропроцессоров (МП) обострилось противоречие между микропроцессорным характером обработки информации и антропогенностью ее алгоритмов. В отличие от традиционных, в основном эвристически созданных алгоритмов для ЭВМ (по аналитическим формулам), появились алгоритмы, ориентированные на аппаратную реализацию и отличающиеся простотой и повышенным быстродействием.

К их числу относятся известные алгоритмы Волдера и Меджита («цифра за цифрой») для вычисления тригонометрических, логарифмических и других специальных функций, положившие начало разностно-итерационным алгоритмам (РИА) [1, 2].

Являясь так называемыми неаналитическими алгоритмами, они не содержат операций умножения и деления и поэтому обеспечивают перевод МП на наиболее дешевую и быстродействующую RISC-архитектуру [3, 4].

Предложенные в [5] РИА и методика проектирования целочисленных алгоритмов цифрового функционального преобразования [6] успешно решают проблему реализации в микропроцессорных системах локальной автоматики практически любой функции одной переменной $F(t)$.

Однако эта методика связана с выполнением большого объема вычислений, а самое главное – с перебором допустимых решений некоторой избыточной нелинейной системы алгебраических уравнений и с многокритериальным выбором оптимального из них.

Эти обстоятельства, а также стремление провести автоформализацию знаний инженера-программиста сделали актуальной задачу радикальной автоматизации процесса разработки РИА в виде САПРа целочисленных алгоритмов («БУЛАТ-02»).

В основу проектируемых алгоритмов функционального преобразования положен базовый РИА, способный после некоторых предварительных линейных преобразований аргумента t в начальные значения итерируемых величин x, y, u, w

$$x = k_x \cdot t + m_x, \quad u = k_u \cdot t + m_u, \qquad y = k_y \cdot t + m_y, \quad w = k_w \cdot t + m_u \tag{1}$$

вычислять функцию вида

$$f(t) = \frac{x(t)\cdot u(t) + y(t)\cdot w(t)}{u(t) + w(t)}.$$

Он таков:

$$\xi_{j-1} = \operatorname{sign}(X_{j-1} - Y_{j-1}) \cdot \operatorname{sign}(u + w);\ X_0 = x,$$

$$X_j = X_{j-1} - \xi_{j-1} \cdot w \cdot 2^{-j+1},\ X_n \to \frac{x\cdot u + y\cdot w}{u + w};\quad (2)$$

$$Y_0 = y,\quad Y_j = Y_{j-1} + \xi_{j-1} \cdot u \cdot 2^{-j+1},\quad Y_n \to X_n,$$

где j = 1, 2, ..., n–2 – номер итерации; n – разрядность аргументов (включая знак).

Чтобы исключить умножение в (1) коэффициенты $k_i, (i \in \{x, y, u, w\})$ выбраны равными 0 или целой неотрицательной степени числа 0,5 и взятыми со знаком «+» или «–». Тогда оно выполняется арифметическим сдвигом аргумента t на $\left|\log_2 |k_i|\right|$ разрядов вправо (в сторону младших разрядов) с присвоением результату знака k_i.

Нетрудно видеть, что практически для любого набора k_i существует своя рациональная функция

$$f(t) = \frac{A\cdot t^2 + B\cdot t + C}{D\cdot t + E},$$

где

$$A = k_x \cdot k_u + k_y \cdot k_w, \quad B = k_x \cdot m_u + k_u \cdot m_x + k_y \cdot m_w + k_w \cdot m_y,$$
$$C = m_x \cdot m_u + m_y \cdot m_w, \quad D = k_u + k_w, \quad E = m_u + m_w.$$

Естественна и обратная задача: для наперед заданной функции $F(t)$ и диапазона $t \in [\alpha, \beta]$ подобрать такие коэффициенты k_i, $m_i (i \in \{x, y, u, w, f\})$, чтобы $F(t) \approx m_f + k_f \cdot f(t)$ с некоторой заранее заданной погрешностью (порядка 0,1–0,01%), коэффициент k_f по модулю могут быть равным 1, 2, 4, 8, ...

Для этого находится аппроксимация Паде

$$F(t) \approx \frac{a_2 \cdot t^2 + a_1 \cdot t + a_0}{b_1 \cdot t + b_0},\quad t \in [\alpha, \beta]. \tag{3}$$

По методу неопределенных коэффициентов формируется система из пяти алгебраических уравнений (три из них – нелинейные) с неизвестными k_i, $m_i (i \in \{x, y, u, w, f\})$.

$$\begin{cases} k_x \cdot k_u + k_y \cdot k_w = \gamma \cdot a_2 / k_f, \\ k_x \cdot m_u + k_u \cdot m_x + k_y \cdot m_w + k_w \cdot m_y = \gamma \cdot (a_1 - b_1 \cdot m_f) / k_f, \\ m_x \cdot m_u + m_y \cdot m_w = \gamma \cdot (a_0 - b_0 \cdot m_f) / k_f, \\ k_u + k_w = \gamma \cdot b_1, \\ m_u + m_w = \gamma \cdot b_0, \end{cases} \quad (4)$$

где γ – некоторый условный коэффициент, сокращающий дробь (3).

Система (4) избыточна. Однако есть ряд ограничений, которые сокращают множество решений вплоть до пустого:

1) по представимости констант в разрядной сетке МП

$$|m_i| < 1; \quad (5)$$

2) по представимости начальных значений итерируемых величин *X, Y, U, W* в разрядной сетке МП

$$|k_i \cdot t + m_i| < 1, \quad (i \in \{x, y, u, w\}); \quad (6)$$

3) по сходимости базового РИА

$$|(k_x - k_y) \cdot t + (m_x - m_y)| < 2 \cdot |(k_u + k_w) \cdot t + (m_u + m_w)|. \quad (7)$$

Ограничения (6) и (7) действуют для любого *t* из диапазона [α, β].

Решение обратной задачи должно быть еще и оптимальным по точности полученного алгоритма, ибо чем больше $|k_f|$, тем меньше число эффективно работающих двоичных разрядов вычислителя (с увеличением $|k_f|$ в $|\log_2 |k_f||$ старших разрядах сумматора будут «0»).

Созданная нами САПР как раз и служит в основном для решения выше приведенной обратной задачи (решение системы (4) с учетом ограничений (5)–(7) при одновременной минимизации величины $|k_f|$).

Ее решение производится разработчиком в диалоговом графоаналитическом режиме с персональным компьютером, оснащенным разработанным нами программным обеспечением (комплексом).

Программный комплекс САПРа создан в весьма эффективной инструментальной среде Visual Basic. При разработке приложения использованы принципы нисходящего проектирования. Для реализации процедур общего диалога при файловых операциях, вывода гистограмм, синтаксического анализа формулы функции и динамического отображения степени завершения вычислительных процедур использовались динамические процедуры (DLL) и расширяемые элементы управления (модули OCX и VBX), входящие в состав среды Visual Basic 4.0 Pro и фирмы Visio Soft.

Разработанное нами приложение работает в среде Windows и соответствует стандарту построения графического интерфейса GUI. Готовое приложение представлено в виде дистрибутива и не требует для своего использования присутствия среды разработки Visual Basic.

Разработка алгоритма конкретной функции производится в ходе одного сеанса общения оператора и САПРа. Причем сама функция *F*(*t*) может быть задана не только аналитически, но и таблично.

САПР не требует от разработчика алгоритмов базового математического образования. Сеанс ведется по сценарию САПРа и сопровождается вопросами с альтернативными ответами «ДА» или «НЕТ» проектировщика. При необходимости даются рекомендации по ходу процесса проектирования при его неудачном течении (по мнению разработчика – низкая точность, нет сходимости, нарушение машинного масштаба).

Программный комплекс представлен в виде дистрибутива на двух дискетах. Для установки необходим компьютер (как минимум IBM 486 DX4 и емкостью ОЗУ 8 Мб), а также монитор с разрешением 800×600 пикселей и объемом видеопамяти 512 Кб. Скоростные характеристики САПРа таковы, что его реакции практически мгновенны. Они следуют через десятые доли секунды. Исключение составляют этапы тестирования и оценки погрешностей полученных алгоритмов (1,5 мин).

Алгоритмы выдаются на монитор в виде блок-схем, детализированных до уровня микропроцессорных команд (без умножения и деления).

САПР целочисленных алгоритмов может найти широкое применение в НИИ, КБ и проектных организациях при алгоритмизации и программировании микропроцессорных систем локальной автоматики.

Литература

1. Оранский А.М. Аппаратные методы в цифровой вычислительной технике. – Минск: Изд-во БГУ, 1977. – 208 с.

2. Байков В.Д., Смолов В.Б. Специализированные процессоры: итерационные алгоритмы и структуры. – М.: Радио и связь, 1985. – 288 с.

3. Анишин Н.С. Основные формализмы разностно-итерационных алгоритмов / Н. С. Анишин, И.Н. Булатникова // Известия ТРТУ. – 2006. – № 5(60). – С. 120–122.

4. Bulatnikova I.N. Modification of difference-iterative algorithms / I.N. Bulatnikova, N.N. Gershunina // International Journal of Control Theory and Applications. – 2016. – Vol. 9, No. 30. – PP. 297–304.

5. Bulatnikova I.N. Classification of difference-iterative algorithms / I.N. Bulatnikova, N.N. Gershunina // Nexo Revista Científica. – 2021. – Vol. 34, No. 6. – PP. 1697–1706. DOI 10.5377/nexo.v34i06.13132.

6. Частиков А.П. Математическое обеспечение автоматизированного рабочего места разработчика алгоритмов для микропроцессорных систем управления / А.П. Частиков, И.Н. Булатникова // Изв. высш. учеб. завед. Сев.-Кавказ. регион. – Сер.: Естественные науки. – 1999. – № 2. – С. 86–88.

Булатникова Инга Николаевна
Канд. техн. наук, доцент каф. прикладной математики Кубанского гос. технологического ун-та (КУбГТУ)
Эл. почта: kras.anis@yandex.ru

Гершунина Наталья Николаевна
Канд. техн. наук, доцент каф. начертательной геометрии и компьютерной графики КубГТУ
Эл. почта: Natalia12_gersh10@rambler.ru

Яковлева Елизавета Сергеевна
Студентка Института механики, робототехники, инженерии транспортных и технических систем КубГТУ
Эл. почта: liza.yakovleva.010.ru@mail.ru

УДК 621.396.61

В.А. Казанин, И.Н. Минин, С.А. Ситников, А.В. Бадьин

Автономная сеть устройств удаленного мониторинга почвенно-климатических параметров в сельском хозяйстве

Описывается разработанная сеть сельскохозяйственных зондов, оснащенных датчиками температуры почвы, температуры воздуха, атмосферного давления и влажности воздуха. Представлены концепция и структура программно-аппаратного комплекса для мониторинга почвенно-климатических параметров. Описан принцип работы системы и приведены результаты тестирования на примере динамики изменения температуры почвы на опытном сельскохозяйственном полигоне. Данная сеть позволяет получать большие объемы данных для дальнейшей координации процесса обработки сельскохозяйственных угодий.

Ключевые слова: базовая станция, LoraWAN, датчики, сельское хозяйство, цифровые технологии, климат, программно-аппаратный комплекс.

Цифровые технологии проникают во все сферы жизни людей, сельское хозяйство это также не обошло стороной. Эти новшества позволяют увеличить и автоматизировать выращивание агрокультур.

Климат представляет большое значение для эффективной сельскохозяйственной деятельности, которая играет важную роль в обеспечении продовольствием населения Земли. Одним из важнейших параметров является выбор оптимальных сроков посева, так как для разных семян есть своя температура прорастания. Следовательно, знание и учет погодных и почвенных параметров в разные периоды роста являются необходимыми звеньями в деле получения высоких урожаев. В добавок к этому увеличивается рост популяции населения, что приводит к необходимости в увеличении продовольствия. Для этого рационально использовать автоматизированную систему мониторинга почвенно-климати-ческих параметров в реальном времени, помогающую делать кратковременные прогнозы и анализ влияния окружающей среды на урожайность агрокультур [1]. Успех в цифровизации технологий выращивания сельскохозяйственных культур поможет решить данные проблемы.

Использование автономного сбора микроклиматических параметров позволит отойти от традиционных методов сбора данных, требующих большого количества времени, рабочей силы и, следовательно, финансовых затрат. Эти условия приведут к повышению эффективности использования ресурсов, уменьшат денежные издержки и позволят увеличить качество урожая.

В последнее время участилось количество разработок агроклиматических комплексов. В большинстве своем они требуют постоянного доступа к сети Интернет, что может быть затруднительно для некоторых сельских хозяйств. Решением данной проблемы может стать программно-аппаратный комплекс, подключенный к сети Интернет через транзитный энергоэффективный радиоканал с модуляцией Lora.

Представленный в данной работе доступный программно-аппаратный комплекс для дистанционного сбора почвенно-климатических параметров позволяет мгновенно получить данные о состоянии почвы и окружающей среды. Сеть из таких устройств позволит сократить человеческие ресурсы, необходимые для постоянного мониторинга параметров окружающей среды.

Описание сети

Для построения сети почвенно-климатических параметров используется шаблон сети LPWAN, где присутствуют конечные устройства, станции LPWAN, сервер и приложение [2]. Система состоит из зондов, базовых станций, сервера и веб-интерфейса. Структура сети приведена на рис. 1.

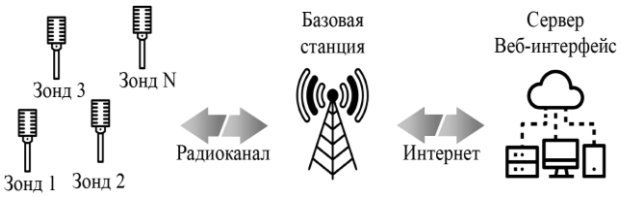


Рис. 1. Структура смарт-сети

Автономные агрозонды могут устанавливаться в различных труднодоступных районах, таких как сельскохозяйственные поля, болота, тайга и т.д. Эти автономные зонды обеспечивают измерение, а также передачу к базовой станции различных почвенно-климатических параметров. Все агрозонды можно настроить в дистанционном режиме с помощью локального веб-интерфейса, который организован в базовой станции.

Внутри самой сети используется принцип временного и частотного разделения каналов, это создано для того, чтобы зонды не мешали друг другу во время «общения» с базовой станцией. Дальность связи между базовой станцией и зондами может достигать 3–7 км в зависимости от выбранной антенны и от особенностей рельефа местности. Данные из базы данных передаются в сетевой веб-интерфейс для отображения конечному пользователю. В сетевом веб-интерфейсе реализован следующий набор функций: система авторизации, карта с отображением зондов и базовых станций на ней, возможность скачать данные, отображение данных зондов в текстовом и графическом видах.

Структура и состав агрозондов

Агрозонды спроектированы специально так, чтобы их можно было с легкостью устанавливать в грунт. По внешнему виду они представляют собой цилиндр с крышкой в виде солнечной батареи. Во время проектирования учитывались антивандальная защита, защита от животных и от неблагоприятных погодных условий. Зонд может комплектоваться датчиками атмосферного давления, влажности воздуха, температуры воды, температуры почвы. Для резервирования собранных данных в зонде установлена карта памяти формата «MicroSD». Структура агрозонда приведена на рис. 2.

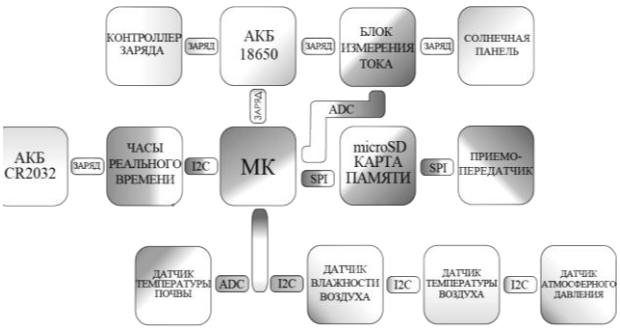


Рис. 2. Структура агрозонда

Описание базовой станции

Базовая станция разработана на основе микроконтроллера ESP-8266. Микроконтроллер построен на базе процессора Tensilica L106 32-bit. Внешняя flash-память микроконтроллера составляет 16 Мб. Этой памяти достаточно, чтобы хранить программу пользователя и небольшие веб-страницы. Кварцевый резонатор для тактирования процессора бывает 80 или 160 МГц.

В базовой станции установлен AMOLED-дисплей для отслеживания состояния подключенных зондов.

Базовая станция осуществляет не только сбор данных с последующей переадресацией на сервер от агрозондов, но также выполняет роль зонда, т.е. на её борту имеются те же датчики, чтобы собирать агроклиматические параметры.

Базовая станция подключена к аккумуляторной батарее, обеспечивающей круглосуточно непрерывную работу. В дневное время подзарядка аккумуляторной батареи осуществляется от поликристаллической солнечной панели.

Для увеличения дальности связи с зондами базовая станция оснащается внешней направленной антенной.

Также она оснащена антивандальной системой для безопасности всей установки, при попытке вскрытия корпуса базовой станции осуществляется звонок в сети GSM на телефон пользователя, ответсвенного за обслуживание системы. Блок-схема базовой станции приведена на рис. 3.

Результаты апробации

Тестирование сети устройств удаленного мониторинга агропараметров проводилось на сельскохозяйственных полях в Томской области путем удаленного мониторинга температуры атмосферы на уровне 30 см от поверхности земли, а также температуры почвы на глубине 10 см. Расстояние между двумя полями (точками мониторинга) составляло 2 км.

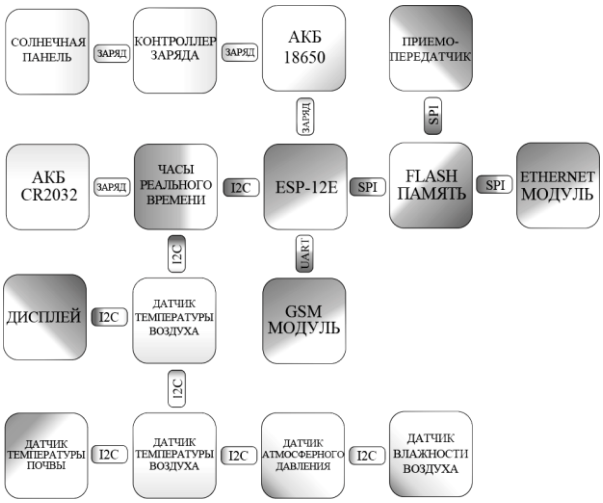


Рис. 3. Блок-схема базовой станции

На рис. 4 приведены данные временной зависимости (сентябрь 2022 г.) температуры почвы на поле № 1 и № 2, соответственно.

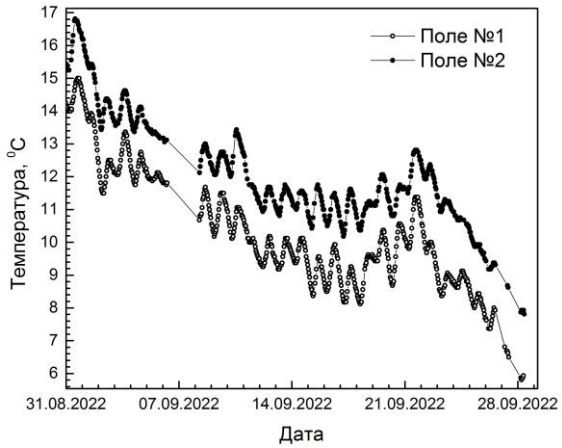


Рис. 4. Динамика температуры почвы на полях № 1 и № 2

Сравнение данных показывает, что температура грунта на поле № 2 выше на 2–3 °C относительно поля № 1. Общая тенденция изменения температуры почвы со временем совпадает на тестовых полях.

Заключение

В приведенной статье описывается разработанная сеть сельскохозяйственных зондов, оснащенная рядом датчиков: температуры почвы, температуры воды, температуры воздуха, влажности воздуха, атмосферного давления. Все это позволяет получать большие объемы данных для мониторинга процесса обработки сельскохозяйственных угодий. Представлены концепция и структура программно-аппаратного комплекса для мониторинга агропараметров. Описан принцип работы системы, приведены результаты тестирования на примере динамики изменения температуры почвы тестовых полей.

Внедрение данной сети на аграрных предприятиях может позволить оперативно принимать решения по засеву либо уборке сельскохозяйственных площадей с учетом локации конкретного поля, тем самым эффективнее расходовать горюче-смазочные материалы.

Работа выполнена в рамках программы развития Национального исследовательского Томского государственного университета (Приоритет-2030).

Литература

1. Южаков М.С. и др. Применение технологии LoRa в IoT системе сбора климатических данных для агробизнеса России // СВЧ-техника и телекоммуникационные технологии / Севастопольский гос. ун-т. – № 3. – С. 101–102.

2. Смольянинов И. Новая технология связи для беспроводных АСКУЭ // Энергетика и промышленность России. – 2016. – № 01-02. – С. 28.

Казанин Владислав Александрович
Студент каф. радиоэлектроники (КРЭ)
Национального исследовательского Томского государственного ун-та (НИ ТГУ)
Эл. почта: kazanin_2013@mail.ru

Минин Илья Николаевич
Студент КРЭ НИ ТГУ
Эл. почта: ilya.kirit@gmail.com

Ситников Серафим Андреевич
Студент КРЭ НИ ТГУ
Эл. почта: simakina200002@mail.ru

Бадьин Александр Владимирович
Канд. физ.-мат. наук, доцент КРЭ НИ ТГУ
Эл. почта: thzlab@mail.ru

УДК 629.735.4

Е.В. Ланин, В.А. Казанин, А.В. Бадьин, К.В. Дорожкин

Концепция использования беспилотного автоматизированного дрона для организации беспроводной сети по протоколу LoraWAN в сельском хозяйстве

Рассмотрена концепция использования БПЛА с базовой станцией как наиболее эффективное средство сбора и обработки информации, поступающей с измерительных устройств (зондов) на значительных площадях сельскохозяйственных угодий. Приведен полный цикл обмена информацией между зондом и конечным сервером, а также описана структура сети с использованием как стационарного типа в специально оборудованном помещении, так и подвижного варианта базовой станции на базе квадрокоптера.

Ключевые слова: квадрокоптер, БПЛА, почвенно-климатические параметры, интернет вещей, LoraWAN.

Количественная оценка почвенно-климатических параметров является ключевым моментом для определения тенденции изменения окружающей среды в сельскохозяйственных угодьях и дальнейшего прогнозирования сроков посева семян [1, 2]. Связано это с тем, что процесс прорастания семян начинается исключительно при определенных температурах. Следовательно, нужно выбрать благоприятные временные сроки для посадки сельскохозяйственных культур.

В идеальном случае количественной оценке присущи такие характеристики, как своевременность получения информации, достоверность результатов исследований и рациональное использование ресурсов. Для получения количественной оценки в сельскохозяйственных угодьях стоит понимать, что обрабатываемая территория достигает несколько сотен квадратных километров. Следовательно, целесообразно использовать сеть устройств автоматизированного сбора и обработки климатических данных со значительным радиусом покрытия.

В настоящее время наблюдается возросшее стремление на разработку агрометеорологических систем, большинству из которых требуется выход в интернет, что ограничивает использование. Данный недостаток, связанный с подключением к сети Интернет, устраняется добавлением радиоканала с протоколом LoraWAN. Технология LoraWAN относится к сети Low Power Wide Area Network (LPWAN) – глобальной сети малой мощности, а сама Lora является сокращением «Long Range», что означает большую дальность. LoraWAN обеспечивает передачу данных на больших территориях с низким энергопотреблением при прямой видимости [3].

В данной работе была поставлена задача – разработать концепцию программно-аппаратной системы для удаленного мониторинга почвенно-климатических параметров с получением данных об окружающей среде как в стационарном варианте, так и в подвижном, используя протокол LoraWAN, взаимодействующего с «Интернет вещей (IoT)».

Организация и состав беспроводной сети

Беспроводная сеть мониторинга почвенно-климатических параметров состоит из стационарной и подвижной (на базе квадрокоптера) базовых станций (БС), измерительных зондов, сервера и двух веб-интерфейсов – локального и сетевого. Пример организации работы сети представлен на рис. 1.

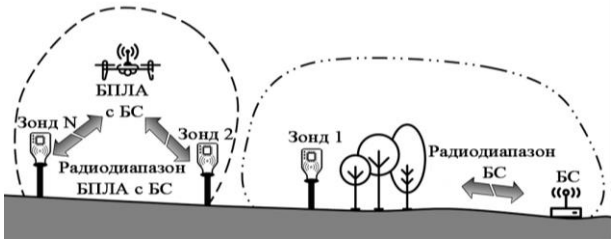


Рис. 1. Организация беспроводной сети

Измерительные зонды устанавливаются в пахотных землях и в автономном режиме измеряют почвенно-климатические параметры с последующей передачей данных на базовую станцию через радиоканал по стандарту LoraWAN.

Стационарная БС монтируется в специально оборудованном помещении, находящемся рядом с сельскохозяйственными угодьями на максимальном расстоянии от зондов – 3 км. Связано это с тем, что оптимальной дистанцией для устойчивого получения информации о состоянии исследуемых полей, с условием прямой видимости между зондом и базовой станцией, является 3 км. На дальность связи влияет рельеф местности, используемая антенна и территории с лесополосами и рощицами.

В оборудованном помещении необходимо постоянное электропитание и доступ в сеть Интернет по проводной или беспроводной линии либо сеть GSM. Базовая станция выполняет следующие задачи: обеспечение приема-передачи данных с агрозондов по радиоканалу и трансляцию их на сервер в интернете через протокол защиты транспортного уровня TLS.

Подвижный вариант БС на базе квадрокоптера используется в том случае, когда нет условий для размещения стационарного вида. С помощью одного квадрокоптера можно собирать данные о почвенно-климатических параметрах со значительной площади пахотных земель – 100–120 км^2, при средних значениях времени полета дрона и крейсерной скорости – 20 мин и 10 м/с. В этом случае передача информации происходит по следующему алгоритму:

измерительные зонды выходят в эфир в назначенный промежуток времени для передачи информации на подлетевший дрон, входящий в эффективный радиус действия обмена данными (рис. 2).

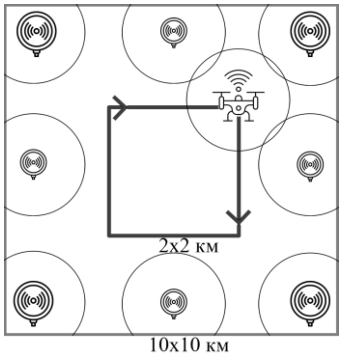


Рис. 2. Организация беспроводной сети
на больших площадях с использованием БПЛА

Подвижный вариант БС отличается от стационарного тем, что данные не передаются на сервер в реальном времени, а по возвращении на наземную станцию – самим оператором либо при подлете в радиус действия вышек сотовой связи (рис. 3).

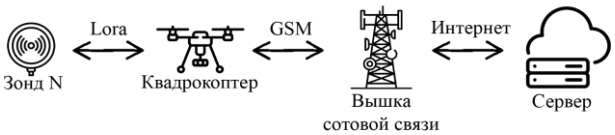


Рис. 3. Принцип передачи информации с зонда на сервер

Логика работы измерительных зондов

Принцип работы агрозонда следующий: после включения устройства происходит регулировка мощности передатчика, считывание показаний с датчиков и дальнейшая отправка на БС. Если передача данных не удалась, то измерительный зонд сохраняет результаты измерений в энергонезависимой памяти и уходит в спящий режим до следующей запланированной активации. Также при успешной передаче информации на БС и отсутствии новых настроек для измерительного зонда последний переходит в максимальное энергосбережение (рис. 4).

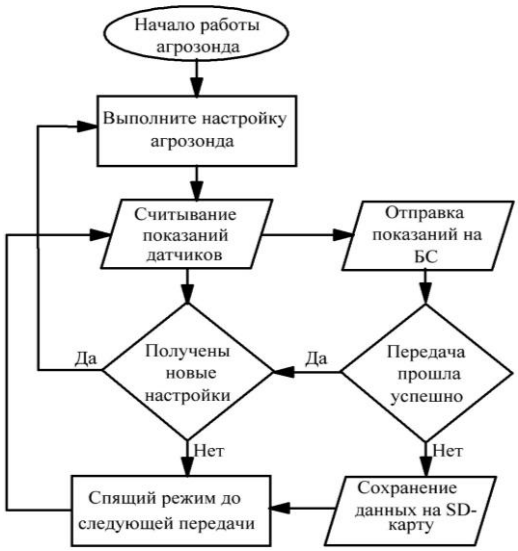


Рис. 4. Логика работы агрозонда

Логика работы базовой станции

При активации БС выполняется первоначальная настройка, включающая в себя первичное подключение и получение конфигурационных данных от сервера. После выставления первоначальных настроек БС переходит в рабочий цикл – проверка установления TLS-подключения к серверу, если соединение успешное, то БС принимает пакеты данных от зондов и отправляет их на сервер. В противном случае БС пытается установить соединение с сервером (рис. 5).

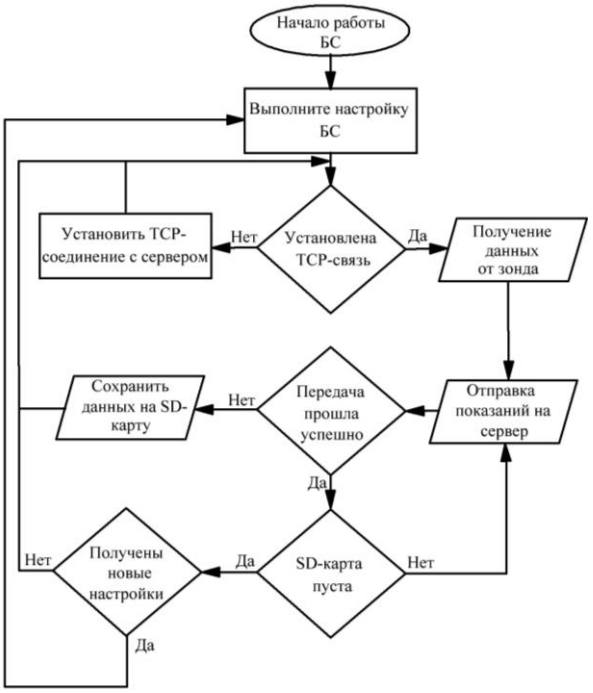


Рис. 5. Логика работы базовой станции

После передачи данных на сервер базовая станция ожидает подтверждения о приеме информации от сервера. При отрицательном ответе все данные записываются на карту памяти, а при положительном ответе базовая станция проверяет, пустая ли карта памяти. Если на SD-карте есть данные, то они передаются на сервер, а если пакета информации на карте памяти нет, то проверяется наличие конфигураций, полученных от сервера и далее либо заново выполняется настройка, либо цикл повторяется без изменения настроек.

Заключение

В приведенной статье рассмотрена концепция использования БПЛА с базовой станцией как эффективное средство сбора и обработки информации, приходящей с измерительных зондов на значительных площадях сельскохозяйственных угодий – 100–120 км^2. Проиллюстрирован полный цикл передачи данных с зонда на сервер, а также изображены и описаны принципы работы измерительных устройств и базовой станции. Представленная концепция сбора данных будет полезна для мониторинга локальных параметров почвы и атмосферы на больших площадях в условиях, когда организация устойчивого канала связи с сетью Интернет отсут-

ствует, при этом сельскохозяйственные угодья расположены в местности с изрезанным ландшафтом и преобладанием лесных массивов.

Работа выполнена в рамках программы развития Национального исследовательского Томского государственного университета (Приоритет-2030).

Литература

1. Rezende J.C.V. Gathering Big Data in Wireless Sensor Networks by Drone / J.C.V. Rezende, R.I. Silva, M.J.F. Souza // Sensors. – 2020. – Vol. 20, № 23. – P. 6954.

2. System of automated weather stations to collect and transmit environmental parameters / M.S. Yuzhakov, A.A. Pavlova, D.A. Pidotova et al. // Journal of Physics: Conference Series: International Conference «Actual Trends in Radiophysics», Tomsk, 01–04 okt. 2019. – Tomsk: Institute of Physics Publishing, 2020. – P. 012023.

3. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios / M. Centenaro, L. Vangelista, A. Zanella and M. Zorzi // IEEE Wireless Communications. – 2015. – Vol. 23 (5). – P. 60–67.

Ланин Евгений Владимирович
Студент каф. радиоэлектроники (КРЭ)
Национального исследовательского
Томского государственного ун-та (НИ ТГУ)
Эл. почта: zenkalanin@gmail.com

Казанин Владислав Александрович
Студент КРЭ НИ ТГУ
Эл. почта: kazanin_2013@mail.ru

Бадьин Александр Владимирович
Канд. физ.-мат. наук, доцент КРЭ НИ ТГУ
Эл. почта: thzlab@mail.ru

Дорожкин Кирилл Валерьевич
Ассистент КРЭ НИ ТГУ
Эл. почта: yasbtk@yandex.ru

Секция 22

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ВЕБ-СИСТЕМ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

*Председатель секции – **Зариковская Наталья Вячеславовна**, доцент каф. АОИ, к.ф.-м.н.*

УДК 004.4

С.А. Селезнев, Р.Е. Богачев, Н.В. Зариковская

Реализация клиентской части административной панели для системы онлайн-бронирования жилищных объектов

Описана реализация административной панели, отвечающей за управление и модерацию контентом для системы онлайн-бронирования жилищных объектов и стек технологий, используемых в данном проекте.
Ключевые слова: веб-разработка, веб-системы, бронирование, онлайн-бронирование жилищных объектов.

В настоящее время большое количество людей передвигаются по стране для работы или отдыха. Развитие внутреннего туризма в России является важной целью правительства.

По данным Федеральной службы государственной статистики, номерной фонд и вместимость в 2017 г. достигли уровня 254 млн ночевок, а количество граждан, путешествующих внутри страны, составляет около 54 млн человек.

В условиях ухода с российского рынка большой части онлайн-сервисов, направленных на бронирование жилищных объектов, реализация аналог,а способного заменить ушедших с рынка конкурентов, является особенно актуальной задачей.

Целью работы является реализация клиентской части системы онлайн-бронирования жилищных объектов.

В данной статье будет рассмотрена реализация клиентской части административной панели для системы онлайн-бронирования жилищных объектов.

Функциональные требования к системе

Система онлайн-бронирования жилищных объектов – это сложная информационная система, реализованная в виде веб-системы, основной задачей которой является поиск доступных предложений на рынке жилищных объектов по целому списку параметров.

Исходя из этого, система онлайн-бронирования жилищных объектов должна обладать следующим списком функциональных возможностей:

- регистрация и авторизация пользователя в системе;
- поиск жилищных объектов по целому списку параметров;
- поиск жилищных объектов на карте;
- регистрация жилищных объектов;
- возможность оставлять отзывы;
- возможность модерации жилищных объектов;
- возможность модерации отзывов;
- возможность модерации пользователей системы.

Схема системы онлайн-бронирования жилищных объектов представлена на рис. 1.

Так как данная работа посвящена реализации клиентской части административной панели, рассмотрим более подробно функциональные требования, предъявляемые к данной подсистеме. Административная панель должна обладать следующим списком функциональных возможностей:

- предоставлять возможность авторизации;
- должна присутствовать возможность управления пользователями системы;
- в административную панель должна быть заложена возможность модерации жилищных объектов;
- должна присутствовать возможность модерации отзывов;
- для списка владельцев жилищных объектов должна быть реализована возможность фильтрация;
- должно быть реализовано отображение подробной информации о владельце жилищного объекта;
- для списка жилищных объектов должна быть реализована возможность фильтрации;
- должно быть реализовано отображение подробной информации о жилищном объекте.

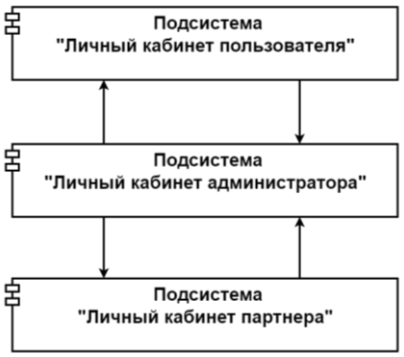


Рис. 1. Схема системы онлайн-бронирования жилищных объектов

Проектирование административной панели

Для определения основных вариантов работы с административной панелью была реализована диаграмма вариантов использования [1], которая наглядно демонстрирует основные варианты взаимодействия администратора с административной панелью. Данная диаграмма представлена на рис. 2.

Именно в соответствии с данной диаграммой велась дальнейшая реализация административной панели.

Стек технологий

Для реализации интерактивности клиентской части веб-приложения были выбраны строго типизированный, высокоуровневый язык программирования TypeScript [2] и библиотека пользовательских интерфейсов React [3].

TypeScript является обратно совместимым с JavaScript и компилируется в последний. Фактически после компиляции программу на TypeScript

можно выполнять в любом современном браузере. TypeScript реализует поддержку подключения модулей, что существенно может повысить скорость разработки, облегчить читаемость, рефакторинг и повторное использование кода, помочь осуществлять поиск ошибок на этапе разработки и компиляции.

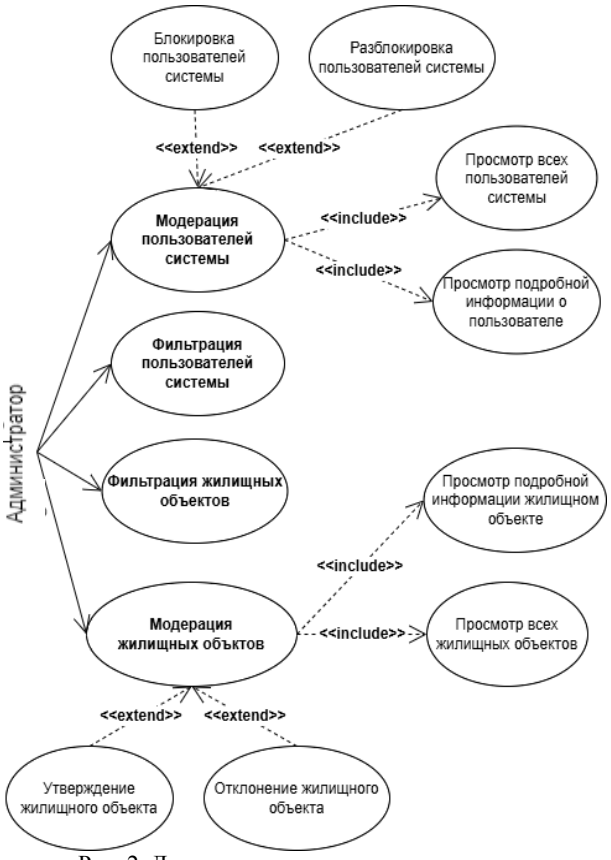


Рис. 2. Диаграмма вариантов использования административной панели

React – JavaScript библиотека с открытым исходным кодом для создания пользовательских интерфейсов.

Для стилизации компонентов системы была выбрана библиотека для управления и написания CSS, Styled components [4].

Библиотека Styled components использует шаблонные строки для стилизации компонентов, данный способ стилизации компонентов, позволяет реализовать простой способ передачи аргумента функции.

Также в STyled component присутствует возможность стилизации компонентов на основе ранее описанных стилей, конструктор библиотеки позволяет реализовать наследование стилей, что, в свою очередь, способствует уменьшению количества кода и увеличивает скорость разработки веб-приложения.

Для реализации сложных компонентов системы была выбрана библиотека готовых пользовательских интерфейсов Ant Design [5].

Данная библиотека имеет большое количество готовых компонентов пользовательского интерфейса и очень подробную сопроводительную документацию. Все таблицы административной панели реализованы с помощью данной библиотеки.

Таблицы в Ant Design имеют встроенную пагинацию, готовую фильтрацию по выпадающему селектору с опциями, которую при необходимости можно модифицировать, поддержку редактируемых ячеек таблицы. Обширный функционал таблиц и большое количество возможностей для их кастомизации значительно ускорили разработку административной панели.

Для взаимодействия с серверной частью веб-приложения были выбраны библиотека Axios и библиотека для управления запросами React Query [6].

React Query предоставляет готовые инструменты, связанные с сетевыми запросами, такими как повторная выборка, предварительная выборка и кэширование. Она разделяет состояние запроса по всему дереву React, к которому существует возможность получить доступ с помощью ключа, и поставляется с готовым механизмом для поддержания актуальности данных путем автоматической повторной выборки данных, когда они необходимы и определяются как устаревшие.

Реализация административной панели

Рассмотрим реализацию административной панели на примере шаблонного компонента для генерации модальных окон и компонента карточки объекта.

Так как все модальные окна в административной панели имеют схожую структуру, на этапе проектирования было принято решение реализовать компонент-обертку, который будет отвечать за генерацию модального окна и в качестве аргумента будет принимать дочерний компонент, который, в свою очередь, будет отвечать только за отрисовку содержимого модального окна. Пример модального окна представлен на рис. 3.

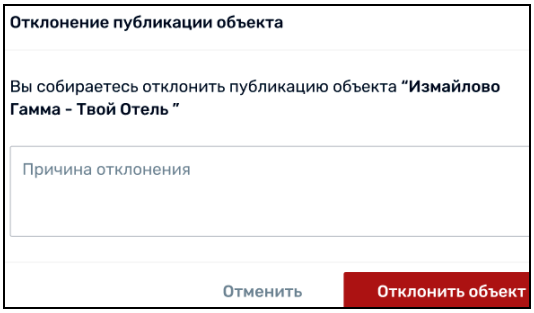


Рис. 3. Реализованое модальное окно

Также на этапе проектирования для всех видов карточек объектов были определены общие структурные блоки, компонент карточки объекта был реализован таким образом, что его содержание и список предоставляемых возможностей зависят от атрибута, отвечающего за текущий статус жилищного объекта. Пример карточки жилищного объекта представлен на рис. 4.

Для адаптивного поведения страниц с большим количеством карточек жилищных объектов был реализован компонент обертки, который отвечает за позиционирование списка карточек жилищных объ-

ектов. Данный компонент обертки позволяет корректно отображать карточки жилищных объектов на разрешении от 320×460 и выше.

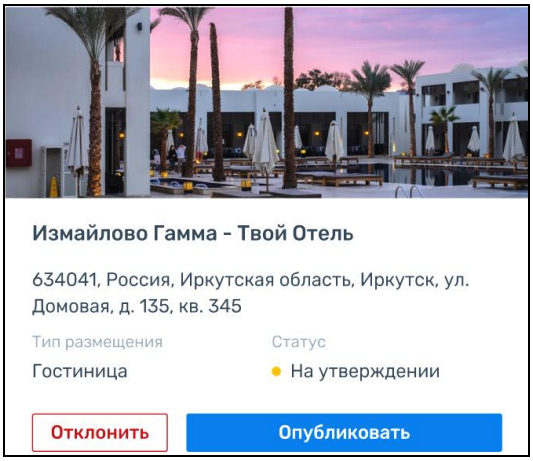


Рис. 4. Реализованная карточка жилищного объекта

Вывод

Приведено описание основных компонентов административной панели для системы онлайн-бронирования жилищных объектов.

Реализованная административная панель полностью соответствует функциональным требованиям.

Литература

1. Диаграмма вариантов использования [Электронный ресурс]. – Режим доступа: https://flexberry.github.io/ru/fd_use-case-diagram.html, свободный (дата обращения: 13.08.2022).

2. TypeScript [Электронный ресурс]. – Режим доступа: http://web.spt42.ru/index.php/chto-takoe-typescript, свободный (дата обращения: 13.08.2022).

3. React [Электронный ресурс]. – Режим доступа: https://ru.reactjs.org/, свободный (дата обращения: 30.05.2022).

4. Styled components [Электронный ресурс]. – Режим доступа: https://tproger.ru/articles/styled-components-idealnaja-stilizacija-react-prilozhenija/, свободный (дата обращения: 13.08.2022).

5. Ant Design [Электронный ресурс]. – Режим доступа: https://ant.design/, свободный (дата обращения: 13.08.2022).

6. React Query [Электронный ресурс]. – Режим доступа: https://react-query.tanstack.com/, свободный (дата обращения: 13.08.2022).

Селезнев Сергей Андреевич
Студент каф. АОИ ТУСУРа
Эл. почта: seleznyov.s.428-1@e.tusur.ru

Богачев Роман Евгеньевич
Студент каф. АОИ ТУСУРа
Эл. почта: bogachev.r.438-1@e.tusur.ru

Зариковская Наталья Вячеславовна
Канд. физ.-мат. наук, доцент каф. АОИ ТУСУРа
Эл. почта: nata.chepko@gmail.com

УДК 004.054

Н.С. Вершинина, З.А. Чепурова, Н.В. Зариковская

Процесс обеспечения качества мобильного приложения

Описаны процесс обеспечения качества мобильного приложения, его этапы, их содержание и ожидаемые результаты.

Ключевые слова: мобильное приложение, тестирование, обеспечение качества, тестовая документация, приемо-сдаточные испытания.

На текущем этапе развития IT-индустрии мобильные приложения (МП) являются одним из самых распространенных видов программного обеспечения. Это обусловлено повсеместным использованием смартфонов ввиду их портативности и эффективности в решении ежедневных пользовательских задач.

Данная статья направлена на описание этапов процесса обеспечения качества мобильного приложения, а также проблем, возникающих на различных этапах процесса и способах их решения.

Рассмотрим более детально работу по обеспечению качества мобильного приложения. Как известно, при реализации мобильного приложения чаще всего используется клиент-серверная архитектура, где непосредственно само мобильное приложение выступает в качестве клиента.

Процесс разработки мобильного приложения ведется итерационно, и для каждой итерации в процессе обеспечения качества выделяются следующие этапы:

- тестирование проектной документации;
- формирование тестовой документации;
- подготовка к тестированию;
- тестирование;
- приемо-сдаточные испытания (ПСИ) и выход приложения на пользователей;
- анализ проблем, возникших за время итерации разработки.

Рассмотрим каждый из перечисленных этапов.

Тестирование проектной документации

Первым этапом процесса обеспечения качества является тестирование проектной документации. В первую очередь, это проверка задокументированных системных и бизнес-требований продукта на полноту, корректность, однозначность, непротиворечивость, осуществимость, проверяемость, необходимость и других характеристик [1].

К тестированию проектной документации также относится анализ дизайн-макетов, по которым мобильные разработчики реализуют клиентскую часть приложения, так как исправление ошибок несоответствия дизайна с задокументированными требованиями на этапе тестирования документации эффективнее и дешевле, чем при непосредственном тестировании уже реализованного функционала.

Основными проблемами данного этапа являются нехватка времени и ресурсов для проведения тестирования документации. Кроме того, на данном этапе очень важны знания по архитектуре и бизнес-логике продукта, отсутствие которых могут потребовать больше времени для тестирования документации.

Для решения этих проблем рекомендуется использовать следующие меры:

- на Confluence (вики-системе для внутреннего использования организациями с целью создания единой базы знаний) создается отдельный раздел, содержащий вводную информацию для быстрого входа в тестирование на проекте и набор ссылок на основную информацию о проекте;
- этап тестирования документации совмещается с этапом формирования тестовой документации;
- специалисты по тестированию принимают активное участие в обсуждении планируемых к реализации функций, одновременно выявляя возможные проблемы в логике предполагаемой реализации.

Формирование тестовой документации

Следующим этапом процесса является формирование тестовой документации. Для тестирования используются:

- тестовые сценарии (тест-кейсы);
- тестовые планы (тест-планы).

В дальнейшем эти артефакты тестирования используются при создании документов «Программа и методика испытаний (ПМИ)», «Стратегия тестирования», «Протокол тестирования», необходимых для допуска продукта к ПСИ и вывода приложения на пользователей.

Основной проблемой данного этапа является отсутствие единого шаблона для создания тест-кейсов. Для решения этой проблемы рекомендуется в командах по тестированию согласование и утверждение единого формата тестовой документации. Однако из-за объединения этапов создания тестовой документации и тестирования проектной документации процесс создания занимает больше времени.

На текущий момент для решения вышеописанных проблем используются следующие методики:

- стандартизация формата тест-кейсов, создание правил оформления;
- создание модульных тест-кейсов и тест-кейсов на API (Application Programming Interface) и их дальнейшее переиспользование в интеграционных и системных сценариях;
- налаживание коммуникаций с аналитиками и дизайнерами проектов для быстрого решения вопросов, связанных с некорректным или недостаточным описанием функционала в документации;

• при возникновении несостыковок в описании и дизайне рекомендуется выносить часть вопросов на обсуждение всей команды проекта для ускорения их решения.

Следует отметить, что с появлением новых функций происходит создание новых тест-кейсов и тест-планов, что, в свою очередь, требует обновления ранее созданной документации. Поддержание в актуальном состоянии артефактов является очень важной проблемой при обеспечении качества любого проекта.

Подготовка к тестированию

Следующий этап – подготовка к тестированию. На текущем этапе проверяется доступность тестовых сред (контуров), создаются тестовые данные, налаживается коммуникация со специалистами из смежных интеграционных систем. Длительность этапа может варьироваться от 0 дней (т.е. его полное отсутствие) до нескольких недель, если, например, на тестовых контурах проводятся технические работы.

Непредсказуемость данного этапа и есть его основная проблема. Специалисты могут столкнуться с проблемами при заведении тестовых данных из-за большого количества интеграционных взаимодействий (для одного пользователя требуется наличие данных о нем более чем в пяти разных системах) или недоступности тестовых контуров (если недоступна одна смежная система, не удастся создать ни одного полноценно работающего пользователя).

Однозначного решения этой проблемы пока не было найдено, однако длительность этапа все еще можно сократить, взаимодействуя со специалистами из смежных систем. Для многих интеграций существуют еженедельные звонки, на которых команды синхронизируются по срокам и подсвечивают проблемные моменты. Также существуют отдельные каналы, которые сообщают о недоступности тестовых контуров и примерном времени решения проблемы. Если команда тестирования не может наладить коммуникацию самостоятельно, она может обратиться для этого к менеджерам проекта.

Тестирование

Тестирование реализации – это следующий этап процесса. Во время тестирования команда опирается на проектную документацию и сформированные тест-кейсы и тест-планы, использует ранее заведенные тестовые данные.

Для тестирования backend (серверной) части приложения используются:

• инструменты тестирования API – SoapUI, Postman, Swagger [2];

• Kafka Tools для тестирования интеграционного взаимодействия через брокер сообщений Kafka;

• инструменты журналирования – Kibana, журнал контейнера OpenShift.

Для тестирования frontend (клиентской) части используются:

• интегрированные среды разработки (IDE) – Android Studio, Xcode;

• инструменты мониторинга трафика HTTP/HTTPS (снифферы) – Charles Proxy, Mitmproxy;

• для записи с экранов используются встроенные утилиты или скачанное из Google Play ПО.

Мобильное приложение может быть реализовано несколькими способами:

• нативным способом – для каждой из платформ iOS и Android разрабатываются отдельные приложения разными специалистами (командами);

• кроссплатформенно – для обеих платформ разрабатывается чаще всего одной командой.

При реализации мобильных приложений специалистам по тестированию нужно понимать, почему на разных платформах одни и те же элементы могут реагировать на действия пользователя поразному.

Первая из основных проблем, с которой могут столкнуться специалисты по тестированию на данном этапе, – недостаток технических знаний. Вторая проблема – недоступность тестовых контуров.

Были найдены следующие способы решения проблем:

• эффективно налаженная коммуникация с командами, разрабатывающими смежные системы, что существенно поможет не только своевременно выявить интеграционные дефекты и понять, со стороны какой из систем они появляются, но и быстро решить проблему в случае недоступности систем;

• на Confluence с описанием проекта хранится информация по инструментам, используемым при тестировании, а также инструкции по их применению и настройке подключений;

• во время формирования тест-кейсов явно указываются использованные макеты и документация, что сокращает время на поиск необходимых для тестирования требований;

• введение автоматизированного тестирования на проект как на отдельные модули, так и на весь проект в целом;

• появление mock-серверов (имитационных серверов), с помощью которых можно тестировать мобильное приложение без привязки к неработающему тестовому контуру.

По факту завершения тестирования новых функций также необходимо провести регрессионное тестирование – вид тестирования, направленный на проверку ранее протестированного функционала после внесения изменений, например, после доработок функций или исправления дефектов.

ПСИ

Следующий этап – этап приемо-сдаточных испытаний (ПСИ). Как упоминалось ранее, для того чтобы продукт вышел на ПСИ, необходимы готовые артефакты ПМИ, «Стратегия тестирования», «Протокол тестирования». Обычно данными документами занимается технический писатель или специалист по тестированию, которому переданы данные обязанности.

Как правило, на этапе ПСИ может возникнуть лишь одна проблема – недоступность тестовых контуров.

После выхода на реальных пользователей и проведения тестирования базового функционала на

реальном (production) контуре (smoke-тестирование) команда практически не взаимодействует с prod-контуром, хотя иногда её могут подключать для локализации особо сложных дефектов.

Главная задача специалистов по тестированию в такой переходный момент – начать анализировать документацию, сформированную аналитиками, к следующему релизу, и выявить проблемы, с которыми они столкнулись в ходе работы.

В результате проделанной работы подготовлена вся необходимая тестовая документация, проведено последовательное тестирование как документации, так и реализованного функционала, устранены основные дефекты, проведено приемо-сдаточное испытание, проект готов к передаче основному потребителю.

Литература

1. Тестирование программного обеспечения. Базовый курс [Электронный ресурс]. – Режим доступа: http://svyatoslav.biz/software_testing_book/ (дата обращения: 09.09.2022).

2. Инструменты тестирования [Электронный ресурс]. – Режим доступа: https://software-testing.ru/library/testing/testing-tools (дата обращения: 12.09.2022).

Вершинина Наталья Сергеевна
Вед. специалист по тестированию ООО «АльдераСофт»

Чепурова Зоя Андреевна
Рук. отд. обеспечения качества ООО «АльдераСофт»

Зариковская Наталья Вячеславовна
Канд. физ.-мат. наук, доцент каф. АОИ ТУСУРа
Эл. почта: nata.chepko@gmail.com

УДК 004.4

Р.Е. Богачев, С.А. Селезнев, Н.В. Зариковская

Разработка модуля администратора серверной части веб-системы для терминалов

Описана схема разработки и реализации части системы, отвечающей за выполнение административного функционала. Представлены архитектура и стек технологий на данном проекте.
Ключевые слова: автоматизация, информационная система, разработка, веб-система.

В настоящее время наиболее актуальной является задача автоматизации любого бизнес-процесса, что позволяет существенно сокращать трудовые, временные и материальные затраты. Одним из таких процессов, которые требуют автоматизации, является процесс продажи SIM-карт.

В современном мире наличие смартфона с доступом к Всемирной паутине, а также возможностью осуществления аудио и текстовой коммуникации, уже становится чуть ли не обязательным условием существования человека в современном обществе. Однако сам по себе смартфон не обладает вышеперечисленным функционалом. Для этого необходимо наличие SIM-карты, которая оформляется у соответствующего оператора связи. Однако на текущий момент данный процесс является достаточно неэффективным.

Соответственно, с целью автоматизации бизнес-процесса продажи SIM-карт необходимо создать веб-систему, состоящую из следующих частей:

- оператор;
- покупатель;
- администратор;
- служба безопасности;
- техник.

В рамках данной работы будет рассмотрена реализация модуля Администратор системы.

Система создается как единая, масштабируемая информационная система и предназначена для решения следующих задач:

- продажи SIM-карт клиентам операторов сотовой связи с помощью средств аудио- и видеосвязи через терминал;
- формирование договора на приобретение SIM-карты на основе сканов документов.

Для корректного взаимодействия различных участников система предполагает разделение пользователей по ролям. Каждая роль представляет собой отдельный модуль в системе:

- Администратор. Обеспечивает выполнение административных и технических функций по обслуживанию работы терминала и порученной ему части системы.
- Оператор. Выполняет взаимодействие с покупателем на протяжении всего жизненного цикла продажи SIM-карты.
- Техник. Занимается обслуживанием терминалов в ручном режиме.
- Покупатель. Выполняет взаимодействие с оператором посредством терминала на протяжении жизненного цикла продажи SIM-карты.
- Служба безопасности. Обеспечивает выполнение административных функций по работе пользователей в системе.

Рассмотрим жизненный цикл продажи SIM-карт через реализуемую систему. Диаграмма жизненного цикла представлена на рис. 1.

Рис. 1. Диаграмма жизненного цикла продажи SIM-карт

Модуль администратора реализован в соответствии со следующими требованиями:

- для доступа к функционалу администратору необходимо иметь возможность авторизации в системе, а также возможность просмотра и редактирования своей информации;
- администратор должен обеспечивать выполнение административных функций по работе пользователей в системе (просмотр, редактирование, создание и удаление);
- при наличии соответствующих прав администратор может получить возможность по управлению администраторами;
- администратор должен обеспечивать выполнение технических функций по обслуживанию работы терминала и заявок на обслуживание (просмотр, создание, редактирование (в ограниченных масштабах)).

В основе реализуемой системы использована клиент-серверная архитектура [1], как показано на рис. 2. Архитектура клиент-сервер – это архитектура компьютерной сети, в которой множество клиентов (удаленных процессоров) запрашивают и получают услуги от централизованного сервера (главного компьютера). Клиентские компьютеры предоставляют интерфейс, позволяющий пользователю компьютера запрашивать услуги сервера и отображать результаты, которые возвращает сервер. Серверы ожидают поступления запросов от клиентов и затем отвечают на них. Эта вычислительная модель особенно эффективна, когда у клиентов и сервера есть отдельные задачи, которые они регулярно выполняют.

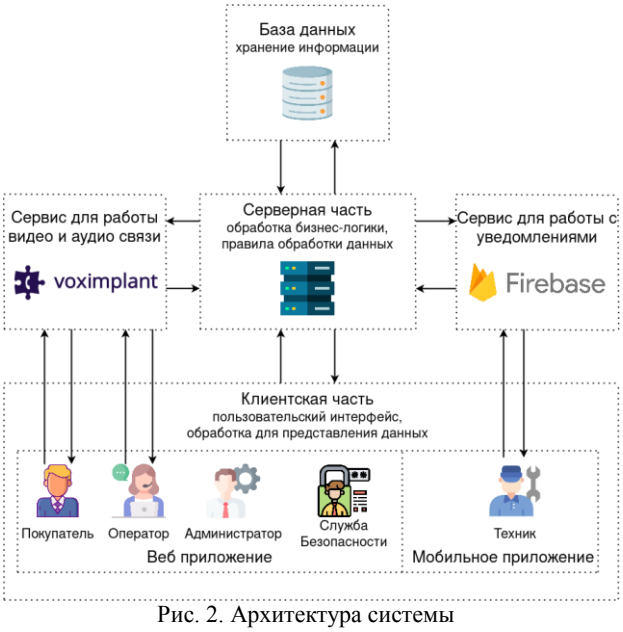


Рис. 2. Архитектура системы

Серверная часть предоставляет доступ к функционалу с помощью Web API, построенного на основе стиля REST. API означает интерфейс прикладного программирования. Web API – это интерфейс прикладного программирования для Web (Всемирной паутины), т.е. Web API – это программный интерфейс, состоящий из одной или нескольких точек доступа, привязанной к определенной системе сообщений вида «запрос–ответ». В данном случае сообщения передаются в формате JSON (JavaScript Object Notation). JSON – это открытый стандартный формат файлов и формат обмена данными, который использует человекочитаемый текст для хранения и передачи объектов данных, состоящих из пар ключ-значение и массивов или других сериализуемых значений.

REST (англ. Representational State Transfer) – это архитектурный стиль для обеспечения стандартов между компьютерными системами в интернете, облегчающий взаимодействие систем друг с другом. REST определяет набор ограничений, учитываемых при проектировании веб-систем. Данные ограничения нацелены на увеличение производительности, масштабируемости и модифицируемости. В качестве протокола доступа к ресурсам используется HTTPS.

Помимо этого, для поддержки коммуникации в режиме реального времени используется протокол WebSocket [2]. WebSocket – это передовая технология, позволяющая устанавливать двусторонний интерактивный сеанс связи между клиентом и сервером. Это делает их взаимодействие более тесным, способствуя распространению интерактивного содержимого и созданию приложений реального времени.

Для разработки серверной части использовались язык программирования Python и его фреймворк FastAPI. Python – это высокоуровневый язык программирования общего назначения с динамической типизацией и автоматическим управлением памятью, ориентированный на повышение производительности разработчика, читаемости кода и его качества, а также на обеспечение переносимости написанных на нём программ [3]. Эталонной реализацией Python является интерпретатор CPython, поддерживающий большинство активно используемых платформ и являющийся стандартом де-факто языка. Он распространяется под свободной лицензией Python Software Foundation License, позволяющей использовать его без ограничений в любых приложениях, включая проприетарные. На текущий момент активно развивается версия языка Python 3.

В контексте веб-разработки и веб-фреймворков Python, можно выделить FastAPI – современный ASGI веб-фреймворк [4]. FastAPI был выпущен в 2018 г. и создан Себастьяном Рамиресом. Рамирес был недоволен существующими фреймворками, такими как Flask и DRF (Django Rest Framework), поэтому он создал свой собственный фреймворк, используя для этого такие библиотеки, как Starlette и Pydantic. Сейчас многие крупные технологические компании, такие как Uber, Netflix и Microsoft, используют FastAPI для создания своих приложений. К основным преимуществам относятся: поддержка как синхронного, так и асинхронного режима работы, легкость в понимании и использовании, встроенная и автоматическая генерация документации

API Swagger и Redoc, а также полное покрытие кодовой базы аннотациями типов.

Для статической проверки типов используется mypy – один из лучших инструментов в этой области, что в итоге позволяет свести ошибки типов на минимум.

Помимо этого, при разработке также использовалась распределенная очередь заданий Celery [5]. Это удобный инструмент, который помогает запускать отложенный или выделенный код в отдельном процессе или даже на отдельном компьютере или сервере.

Для хранения данных была выбрана реляционная база данных, а в качестве СУБД – PostgreSQL. Для интеграции с серверной частью использовалась библиотека SQLALChemy, которая представляет как функционал ORM (технология, которая связывает базы данных с концепциями объектно-ориентированного программирования (далее – ООП)), так и конструктор SQL-запросов [6].

Возможность видеообщения между модулями оператора и покупателя в режиме реального времени достигается благодаря сервису Voximplant – платформы, позволяющей разработчикам интегрировать аудио- и видеозвонки в свою приложения [7].

FastAPI предоставляет несколько инструментов, которые помогут справиться с проблемой безопасности и авторизации. Данная система использует спецификацию OAuth2. OAuth2 – это протокол индустриального стандарта для авторизации. Данная спецификация и ее расширения разрабатываются в рамках рабочей группы IETF Oauth [8]. В качестве средства авторизации используются JWT-токены. JWT-токены (JSON Web Tokens) – это открытый стандарт для создания данных с необязательной подписью и/или необязательным шифрованием, полезная нагрузка (от англ. payload) которого включает JSON, содержащий некоторое количество утверждений (от англ. claims). Токены создаются сервером, подписываются секретным ключом и передаются клиенту, который в дальнейшем использует их для подтверждения своей личности. В общем случае используется два вида токенов: access (для доступа к другим точкам доступа) и refresh (для обновления пары токенов).

Как уже сказано выше, FastAPI способен автоматически генерировать документацию Swagger. Swagger – это набор правил, спецификаций и инструментов с открытым исходным кодом для разработки и описания RESTful API (так называют API, соответствующий ограничениям REST). Фреймворк Swagger позволяет разработчикам создавать интерактивную, машиночитаемую и человекочитаемую документацию API.

Генерация полной и точной документации достигается благодаря использованию библиотеки Pydantic, которая обеспечивает проведение валидации данных и управление настройками с помощью аннотаций типов, а также способна генерировать схемы, совместимые с OpenAPI (спецификация, на которой основан Swagger), JSON Schema Core и JSON Schema Validation, а также производить конвертацию входных данных к нужному типу [9]. Под схемой понимается описание структуры данных (название переменных, их тип, ограничения и т.д.), которые могут как возвращаться с сервера, так и приходить от пользователя. Сами модели – это просто классы, которые наследуются от базовой модели pydantic.BaseModel и определяют необходимые атрибуты. Все модели валидируют входные данные, в результате получаем целостность данных с обеих сторон: со стороны клиента, когда он делает запрос на сервер, и со стороны сервера, когда он отправляет ответ клиенту. Это позволяет обеим сторонам быть уверенными в том, что приходящие данные корректны и в нужном формате.

Серверную часть системы логически можно разделить на 5 уровней, которые представлены на рис. 3.

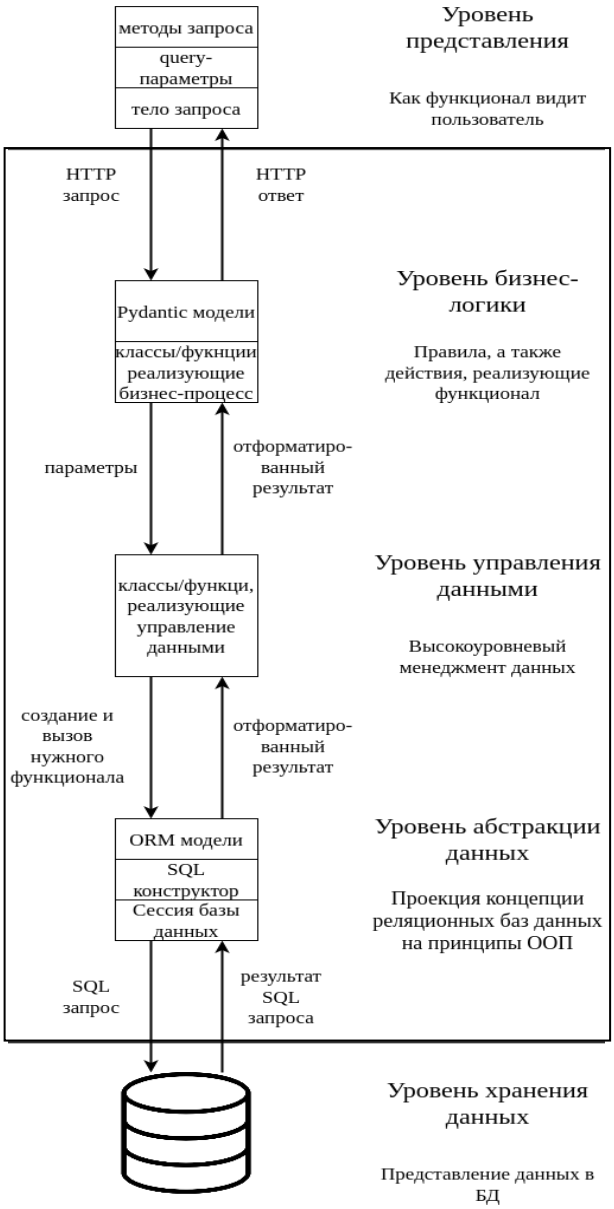


Рис. 3. Уровни серверной части системы

Из этого можно сделать вывод, что система является многоуровневой. Многоуровневый дизайн фокусируется на возможностях тонкой настройки системы, позволяя различным технологиям хорошо работать вместе. Многоуровневая архитектура ПО имеет ряд преимуществ, среди которых одним из самых главных является то, что разделение позволяет управлять и поддерживать каждый уровень соответствующим образом. Это помогает значительно упростить управление программной инфраструктурой. В основном все реализуемые модули взаимодействуют с конкретными сущностнями (например, терминал или тариф) и, в основном, могут проводить операции CRUD (акроним, обозначающий четыре базовые функции, используемые при работе с базами данных: создание (от англ. create), чтение (от англ. read), модификация (от англ. update), удаление (от англ. delete)) над ними. Это всё достигается благодаря библиотеке SQLAlchemy и реализуется на уровнях управления и абстракции данных. По сути, эти операции не являются сложными. Для всех сущностей алгоритм действий будет примерно одинаковый: на уровне представления пользователю даётся набор точек доступа, а также допустимые параметры. На уровне бизнес-логики происходит валидация (с помощью pydantic) и проверка переданных пользователем параметров. Далее идёт обращение на уровень управления данными, который уже совершает запрашиваемую операцию (создание, чтение, модификация или удаление) и возвращает результат, который конвертируется к pydantic модели и отдаётся пользователю.

Выбранный стек технологий позволит реализовать систему достаточно гибкой, масштабируемой и надежной.

В результате проделанной работы был реализован модуль Администратор, помимо этого были реализованы:

- протокол авторизации в соответствии со спецификацией OAuth2;
- необходимые инструменты и компоненты, позволяющие Администратору выполнять CRUD-операции над пользователями и терминалами;
- гибкая системы прав, позволяющая наделять Администратора дополнительными возможностями.

Литература

1. Клиент-серверная архитектура [Электронный ресурс]. – URL: https://cio-wiki.org/wiki/Client_Server_Architecture.

2. Web API [Электронный ресурс]. – Режим доступа: https://www.w3schools.com/js/js_api_intro.asp, свободный.

3. Богачёв Р.Е., Зариковская Н.В. Возможности вебфреймворка FastAPI для реализация серверной части вебсистем // Сборник науч. статей по матер. Всерос. науч. конф. «Достижения науки и технологий» (ДНИТ–2021), Красноярск, 10–11 декабря, 2021 г. – С. 289–293.

4. FastAPI [Электронный ресурс]. – URL: https://fastapi.tiangolo.com/.

5. Celery [Электронный ресурс]. – URL: https://docs.celeryq.dev/en/stable/getting-started/introduction.html

6. SQLAlchemy: The Python SQL Toolkit and Object Relational Mapper [Электронный ресурс]. – URL: https://www.sqlalchemy.org/

7. Voximplant [Электронный ресурс]. – URL: https://voximplant.com/

8. OAuth 2.0 [Электронный ресурс]. – URL: https://oauth.net/2/

9. Pydantic [Электронный ресурс]. – URL: https://pydantic-docs.helpmanual.io/

Богачев Роман Евгеньевич
Студент каф. АОИ ТУСУРа
Эл. почта: bogachev.r.422-m1@e.tusur.ru

Селезнев Сергей Андреевич
Студент каф. АОИ ТУСУРа
Эл. почта: seleznyov.s.422-m1@e.tusur.ru

Зариковская Наталья Вячеславовна
Канд. физ.-мат. наук, доцент каф. АОИ ТУСУРа
Эл. почта: nata.chepko@gmail.com

УДК 004.4'236

А.С. Стецко

Графическая среда программирования «ДРАКОН-Си». Редактор ДРАКОН-схем

Рассмотрены аспекты создания редактора дракон-схем в виде веб-приложения. Дано описание операций добавления, удаления и редактирования графических элементов. В заключение были представлены возможные улучшения редактора.

Ключевые слова: язык ДРАКОН, ДРАКОН-схемы, SVG, скрипт.

Визуальное программирование стало новым и более эргономичным способом формализации знаний. ДРАКОН – один из языков визуального программирования, отличающийся от своих аналогов тем, что является формальным языком. Поэтому ДРАКОН-схема алгоритма может быть однозначно интерпретирована в текст программы на любом текстовом языке программирования. Правила построения ДРАКОН-схем гораздо строже, чем у аналогов. На сегодняшний день не существует графической среды построения ДРАКОН-схем, реализованной в виде веб-приложения с возможностью генерации схемы в текст программы на Си [1]. Следовательно, была поставлена цель разработать такое приложение. Для достижения цели решались следующие задачи: 1) исследовать аналогичные системы; 2) составить требования к среде «Дракон-Си»; 3) выбрать инструменты разработки; 4) реализовать редактор ДРАКОН-схем.

Требования к среде

Главная составляющая среды программирования «ДРАКОН-Си» – редактор. В нём пользователь создаёт дракон-схемы. Среда программирования предоставляет пользователю возможность сохранить дракон-схему с целью дальнейшего редактирования и интерпретировать её в текст программы на языке программирования Си. Необходимо определить правила и требования, по которым функционируют среда графического программирования и редактор ДРАКОН-схем, в частности. Требования были составлены на основе исследования аналогичных систем: программы «Фабула» и веб-приложения «DrakonHub» [2, 3].

Графическая среда программирования должна обеспечивать: 1) соблюдение подмножества правил языка ДРАКОН; 2) создание и редактирование ДРАКОН-схем; 3) сохранение схемы с возможностью её дальнейшего редактирования; 4) сохранение полученной схемы в одном из распространённых графических форматов; 5) интерпретацию ДРАКОН-схем в тексты программ на языке Си.

Среди правил языка ДРАКОН выделим самые важные для разработки редактора: 1) схема должна содержать графические элементы, явно выделяющие начало и конец алгоритма; 2) добавление графических элементов возможно только в определённых местах ДРАКОН-схемы; 3) пользователь должен видеть все возможные места для добавления элемента и выбирать только одно из них. В теории языка ДРАКОН такие места называются «валентными точками» и изображаются в виде кругов, лежащих на соединительных линиях схемы, однако не исключены другие вариации.

Используемые инструменты

Графическая среда программирования «Дракон-Си» реализована в виде веб-приложения, для работы которого требуется только клиентская часть [4]. Причины выбора такого варианта реализации [5]: веб-приложение не требует установки на компьютер, в малой степени зависит от аппаратного обеспечения, потенциальная возможность внедрения среды в систему управления обучением (LMS).

«Классическими» инструментами для создания веб-приложения можно назвать HTML, CSS, JavaScript. HTML – язык гипертекстовой разметки – необходим для выделения участков веб-страницы. CSS – инструмент задания стилей и позиций объектам. С помощью стилей интерфейс становится более дружелюбным. JavaScript – язык программирования, обеспечивающий динамику веб-приложения. Весь функционал приложения, начиная от нажатия на кнопку и заканчивая отображением ДРАКОН-схемы, должен быть прописан в js-файле (скрипте).

В качестве инструмента работы с графикой был выбран тег <svg> языка HTML. Этот тег инициализирует контейнер масштабируемой векторной графики, примитивы которой также объявляются тегами. Важно отметить превосходство векторной графики над растровой: масштабируемость, экономия памяти, предпочтительность при рисовании икон, логотипов. Главные сложности в использовании SVG – зависимость положения фигур от координат и отсутствие представления текста как содержимого фигуры. Для решения этих проблем были разработаны специальные функции в скрипте.

Структура проекта

Владимир Паронджаров, создатель языка ДРАКОН, определил полный набор икон и макроикон [6]. Из полного перечня были выбраны только те, которые применимы в большинстве алгоритмов и которые не затруднительно ассоциировать с фрагментом программы на языке программирования Си (рис. 1). Эти элементы помещены в меню пользователя и называются «атомами».

Разрабатываемое веб-приложение должно корректно отображать все атомы. Также оно должно

хранить ДРАКОН-схему в таком виде, чтобы не возникало затруднений при её интерпретации.

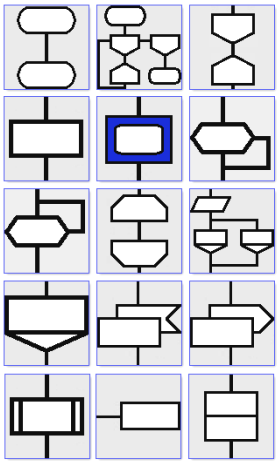


Рис. 1. Меню атомов

Эти причины привели к выбору способа представления ДРАКОН-схемы в приложении в виде матрицы графических контейнеров SVG (рис. 2).

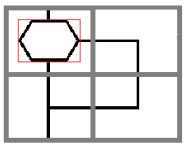


Рис. 2. Атом «Развилка» в виде матрицы блоков

В результате анализа используемых контейнеров SVG был составлен набор используемых блоков, т.е. уникальных по назначению ячеек матрицы. Таким образом, добавление и удаление атомов сводится к операциям добавления строк и столбцов в матрицу и удаления их.

Исходя из представления ДРАКОН-схемы в среде «ДРАКОН-Си» в виде матрицы блоков, была составлена общая структура хранения схемы (рис. 3). Структура будет учитываться при написании функций в скрипте. В рамках языка JavaScript сущности будут называться «объектами», а их параметры – «ключами».

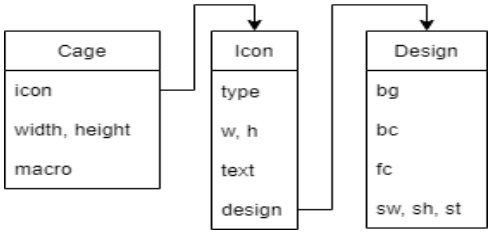


Рис. 3. Структура ДРАКОН-схемы

Объект Cage является контейнером SVG. Он имеет определённые размеры координатной сети для размещения примитивов. Начало координатной сетки расположено в левом верхнем углу [7].

Добавление любого атома обязательно проходит через функции четырёх модулей: приём, хранение, эргономизация, рисование. ДРАКОН-схема редактируется поэтапно. Переход от одного этапа к другому происходит путём вызова функцией, принадлежащей одному модулю, функций, принадлежащих другим модулям. Факт обращения к модулям можно отразить на схеме (рис. 4). Особый интерес представляет модуль рисования и модуль приёма данных. Именно благодаря этим модулям можно создавать, редактировать и отображать ДРАКОН-схему.

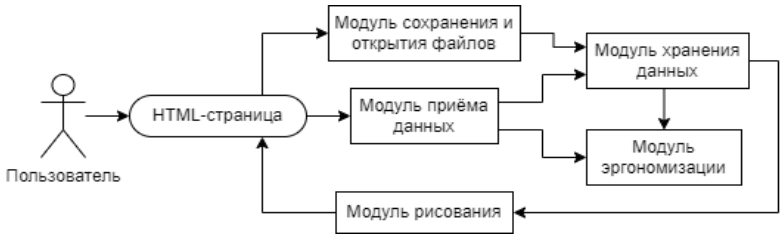


Рис. 4. Обращение пользователя к JavaScript-файлам

Модуль приёма данных

Функции модуля приёма данных позволяют пользователю добавлять атомы, удалять атомы и редактировать иконы. Чтобы добавить атом, пользователь должен выбрать его из меню. В выборе корректного атома пользователю помогают подсказки (рис. 5).

Рис. 5. Подсказка для атома «Примитив»

Затем пользователь выбирает позицию добавления атома, валентную точку. Доступные позиции выделяются зелёным цветом (рис. 6).

Рис. 6. Валентные точки дракон-схемы

После этого пользователь указывает содержание икон и стилевые параметры в специальной па-

нели (рис. 7). Чтобы отредактировать икону, необходимо кликнуть по ней левой кнопкой мыши и ввести новые параметры в другую специальную панель (рис. 8). Так как икона однозначно принадлежит одному атому, удаление иконы означает удаление атома. Совершается удаление иконы в той же панели, что и её редактирование.

Рис. 7. Панель ввода параметров атома

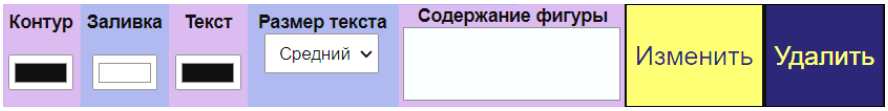


Рис. 8. Панель редактирования иконы

Модуль рисования

Результаты действий пользователя – добавления, удаления и редактирования – должны быть отображены на веб-странице. Функции модуля рисования позволяют это сделать. Они визуализируют матрицу контейнеров SVG. Содержимое контейнера называется примитивами, которые представляются в виде тегов. Из всех примитивов [8] были выбраны только те, которые пригодны для использования в функции рисования блоков DrawCage (таблица). Кроме параметров, перечисленных в таблице, в тегах используются атрибуты стилей: толщина контура (stroke-width), цвет контура (stroke), заливка (fill).

Примитивы SVG, используемые в модуле рисования

Примитив	HTML-тег	Атрибуты
Отрезок	<line>	x1, y1 – координаты начала x2, y2 – координаты конца
Прямоугольник	<rect>	x, y – координаты верхнего левого угла width, height – ширина, высота
Многоугольник	<polygon>	points – координаты вершин
Текст	<text>, <tspan>	x,y – координаты левого нижнего угла

Выводы

В результате работы был создан редактор ДРАКОН-схем. В созданной среде пользователь может добавлять, изменять и удалять графические элементы. ДРАКОН-схема оперативно обновляется после каждой операции и доступна в любой момент для просмотра и сохранения. Однако графическая среда программирования «ДРАКОН-Си» имеет недостатки, подмеченные как разработчиками среды, так и экспертами в данной предметной области [9]. Процесс редактирования ДРАКОН-схем довольно громоздок для пользователя – зрительная, физическая и умственная нагрузка не соответствует простоте операции. Особого внимания заслуживают несколько предложений по исправлению этого недостатка. Во-первых, необходимо пересмотреть реализацию валентных точек, указывать потенциальную позицию добавления атома более явно. Во-вторых, необходимо решить вопрос удаления непустых циклов, развилок и переключателей. В-третьих, необходимо позволить пользователю создавать абстрактные дракон-схемы, т.е. исключить панель ввода параметров атома, а наличие текста проверять только в необходимых случаях (генерация текста программы). В-четвёртых, необходимо скрывать фрагменты программы в иконе, чтобы ДРАКОН-схема содержала только общепонятные формулировки. Главной идеей усовершенствованной графической среды программирования ДРАКОН-Си можно назвать принцип создания ДРАКОН-схемы за минимальное число кликов.

Литература

1. Визуальный язык ДРАКОН [Электронный ресурс]: дружелюбный русский алгоритмический язык, который обеспечивает наглядность. – URL: https://drakon.su, свободный (дата обращения: 14.09.2022).

2. Программа «Фабула» [Электронный ресурс]. – URL: https://drakon.su/programma_fabula_._redaktor_drakon-sxem, свободный (дата обращения: 14.09.2022).

3. DrakonHub [Электронный ресурс]. – URL: https://drakonhub.com, свободный (дата обращения: 14.09.2022).

4. Как работают веб-приложения [Электронный ресурс]. – URL: https://habr.com/ru/post/450282/, свободный (дата обращения: 14.09.2022).

5. Десктопное или веб-приложение: плюсы и минусы [Электронный ресурс]. – URL: https://vc.ru/services/297762-desktopnoe-ili-veb-prilozhenie-plyusy-i-minusy, свободный (дата обращения: 14.09.2022).

6. Пароннджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. – М.: ДМК Пресс, 2012. – 520 с.

7. SVG. Координаты [Электронный ресурс]. – URL: http://rezerford-svg.blogspot.com/2011/06/2.html, свободный (дата обращения: 14.09.2022).

8. Учебник HTML5 графики – SVG [Электронный ресурс]. – URL: https://msiter.ru/tutorials/svg, свободный (дата обращения: 14.09.2022).

9. Графическая среда программирования «ДРАКОН-Си» [Электронный ресурс]. – URL: https://forum.drakon.su/viewtopic.php?f=143&t=7151, свободный (дата обращения: 14.09.2022).

Стецко Алексей Сергеевич
Студент каф. АСУ ТУСУРа
Эл. почта: stetsko-alexeist2015@ya.ru

УДК 004.772

А.Б. Виноградов, Е.В. Горелкина, К.К. Ермаков

Система мониторинга соблюдения техники безопасности и состояния здоровья сотрудников

Проведен анализ существующих систем мониторинга техники безопасности и состояния здоровья сотрудников, на основе которого сформулированы задачи по разработке собственной системы удаленного мониторинга. Разработана схема комплекса инструментального программного обеспечения и описаны функциональное назначение и взаимосвязь всех его элементов.

Ключевые слова: система мониторинга состояния здоровья, система удаленного мониторинга, разработка программного обеспечения, индивидуальные гаджеты.

Создание безопасных условий труда на рабочих местах является важной задачей охраны труда любого предприятия, так как от качества ее выполнения зависит не только деятельность самого предприятия, но и жизнь и здоровье его сотрудников. Именно поэтому данная задача является актуальной и требует постоянной модернизации средств и способов ее выполнения.

Для быстрой и удобной реализации мероприятий по контролю соблюдения техники безопасности и состояния здоровья сотрудников в современном мире все больше используются различные беспроводные системы безопасности, например, цифровые решения с использованием индивидуальных гаджетов. Все больше производственных предприятий ставят разработчикам задачи по удаленному мониторингу сотрудников. Подобную задачу поставил ежегодный конкурс инновационных проектов «Лаборатория энергетики».

Задачей проекта является предложение решения удаленного мониторинга соблюдения техники безопасности и состояния здоровья сотрудников. Для предприятий, производственные участки которых удалены от специалистов по охране труда, существует потребность в оснащении производственных рабочих индивидуальными гаджетами (далее – браслетами), которые позволят осуществлять мониторинг состояния здоровья, а также предупреждать производственный травматизм за счет контроля перемещений сотрудника. В рамках проекта предлагается разработать прототип, основанный на измерении показателей состояния здоровья рабочих с возможностью геопозиционирования и разграничения рабочих и опасных зон [1].

Анализ существующих систем

Проведен анализ существующих систем мониторинга техники безопасности и состояния здоровья сотрудников, использующих индивидуальные гаджеты как основной инструмент для своего проекта.

В таблице представлены данные системы удаленного мониторинга в соотношении с задачами, поставленными «Лабораторией энергетики». В случае наличия у системы необходимого функционала, на пересечении наименования системы мониторинга и основных функциональных требований ставится соответствующая пометка. Для анализа была использована информация о системах мониторинга, которая находится в общем доступе и представляет базовый вид проекта организации-поставщика [2–5].

По результатам анализа, представленным в таблице, можно сделать вывод, что на данный момент система мониторинга, отвечающая всем требованиям «Лаборатории энергетики», отсутствует. Следовательно, было принято решение по разработке собственного комплекса инструментального программного обеспечения (ПО), выполняющего все вышеприведенные задачи.

Схема работы системы

Комплекс инструментального ПО разработан для реализации системы мониторинга техники безопасности и состояния здоровья сотрудников. Он включает в себя 3 основных компонента (рис. 1):

- комплекс серверного ПО;
- приложение для персонального компьютера (ПК) оператора;
- приложение для смартфона сотрудника.

Данные о состоянии здоровья сотрудника, вызове кнопки SOS и др. получает браслет и, посредством беспроводной Bluetooth связи они отправляются на смартфон пользователя, который осуществляет сбор информации о местоположении сотрудника с помощью встроенного модуля GPS.

Основным элементом схемы взаимодействия является сервер, выступающий связующим элементом между смартфоном сотрудника и ПК оператора. Связь с сервером осуществляется по протоколу TCP. Данные поступают от мобильного телефона пользователя на сервер, который их обрабатывает и хранит. После анализа данных сервер может отправлять отдельные показатели оператору на ПК в том случае, если они не соответствуют заданным нормам, а также предупреждать сотрудника уведомлением на мобильный телефон или браслет в случае совершения опасных действий, например о перемещении в опасную зону и т.д. Кроме того, сам оператор может подключаться к серверу через приложение на ПК и контролировать полученные данные. Сервер хранит информацию в течение месяца, что позволяет оператору составлять ежемесячные отчеты и отслеживать динамику показателей по каждому сотруднику.

Сравнение существующих на российском рынке систем мониторинга соблюдения техники безопасности и состояния здоровья сотрудников с помощью индивидуальных гаджетов

№ п/п	Основные функциональные требования к системе	Наименование системы мониторинга			
		1	2	3	4
		Стрелец-ПРО	Sitronics Smart Watch	Smart Team	SiWatch
1	Отслеживание температуры тела, пульса, резких перемещений в режиме онлайн	+	+	+	+
2	Сохранение данных о состоянии сотрудников в течение суток			+	+
3	Хранение информации о входе/выходе из рабочей зоны в течение месяца			+	+
4	Возможность задачи оператором нормативных показателей для сотрудника				
5	Возможность подачи сигнала SOS с устройства оператору	+	+	+	+
6	Ношение устройства со штатными средствами индивидуальной защиты, отсутствие рисков нарушения требований охраны труда	+	+	+	+
7	Комплектация устройством в течение не более 2 мин	+	+	+	+
8	Время автономной работы не менее 12 ч			+	+
9	Выдача заданий и направление уведомлений о начале/завершении работ, перемещении между зонами объекта	+	+	+	+
10	Фиксация на устройствах момента завершения работ, итоговых временных затрат на их выполнение		+	+	+

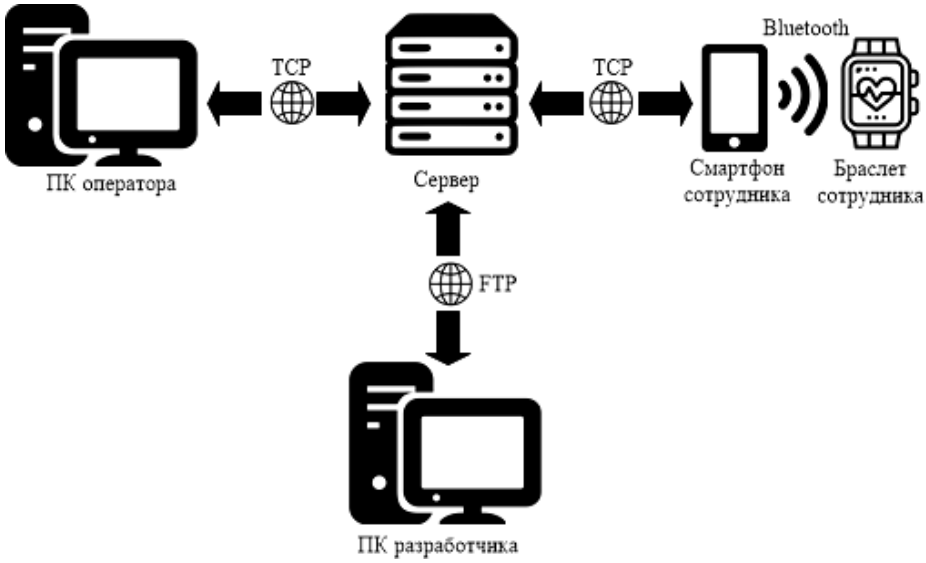


Рис. 1. Схема работы системы мониторинга соблюдения техники безопасности и состояния здоровья сотрудников

Разработка сервера

Разработка сервера является первостепенной задачей создания проекта, так как он выступает основным элементом системы. Ввиду сложившейся в настоящее время ситуации было решено написать собственный сервер, так как большинство ПО и сервисов закрыты. Вследствие этого данный проект сможет работать и развиваться без значительных простоев и потерь времени.

Функционал сервера базируется на основных требованиях, выдвинутых системе, и решает такие задачи, как ограниченный доступ к данным, защищенным базой логинов и паролей; точная и безошибочная передача данных между элементами системы; обработка и хранение всех поступивших данных; рассылка указанной информации. Для обработки полученной информации требуются базы данных и система управления ими, что позволит создать на сервере систему выполнения задач от оператора. Эта система позволит ставить серверу задачи, выполнение которых возможно осуществить без участия оператора.

Для первоначального запуска проекта было принято решение выбрать в качестве сервера локальный ПК со статическим IP-адресом. FTP-сервер, созданный на том же ПК, позволит разработчикам осуществлять удаленный доступ и устанавливать обновления версий ПО сервера [6]. Кроме того, для удобства и универсальности доступа к данным со стороны оператора требуется создать веб-сервер на том же ПК. После проведения тестирования и запуска пилотного проекта планируется переход на облачные сервисы, так как они обладают большей мощностью и смогут обеспечивать доступ большему количеству пользователей.

Разработка приложения для смартфона

Решение об использовании смартфона принято из-за соображений экономии времени и ресурсов, так как автономное устройство в виде браслета с поддержкой необходимых модулей имеет большие

габариты. Поэтому приложение для смартфона является неотъемлемой частью проекта, так как связывает браслет и сервер, обеспечивая передачу данных о состоянии сотрудников, уведомлениях и сигналах посредством Bluetooth связи и мобильной сети. Кроме того, сбор информации о местоположении пользователя осуществляется через встроенный в телефон модуль GPS.

Интерфейс приложения представляет собой навигационное меню, простое и удобное в использовании.

Функционал приложения на первоначальном этапе содержит в себе получение уведомлений и оповещений от оператора или сервера, а также отправку обратного сообщения; краткую справку по работе с приложением.

В будущем планируется добавить в приложение указанные работодателем нормативы показателей, а также уведомления о наступлении и окончании перерыва. Кроме того, после тестирования продукта и проведения опроса пользователей, работавших с приложением, предполагается осуществление доработки функционала приложения и изменения интерфейса в соответствии с пожеланиями работника, которые будут одобрены заказчиком.

Разработка приложения для оператора

Взаимодействие оператора с информацией, полученной от сервера, осуществляется с помощью приложения для ПК. Программа представляет собой окно для запроса данных по каждому сотруднику отдельно или по группе сотрудников. Кроме того, существует возможность отправки уведомлений сотруднику оператором.

В будущем планируется модернизация приложения до уровня программы, содержащей в себе данные от каждого сотрудника в режиме онлайн, которые хранятся в течение определенного периода времени с возможностью составления отчетов по заранее заданным формам. В случае возникновения происшествия это поможет в восстановлении объективной картины для дальнейшего предупреждения подобных ситуаций. Кроме того, добавится возможность просмотра карт разграничений рабочих зон с указанием текущего местоположения сотрудника и истории показателей здоровья и перемещений сотрудника за последние сутки.

Заключение

Данная система мониторинга техники безопасности и состояния сотрудников актуальна как для крупных компаний, производственные участки которых находятся на удаленных территориях, так и для небольших фирм, которым требуется контроль доступа сотрудников в то или иное помещение. Поэтому система востребована широкой аудиторией и ее можно интегрировать в любое предприятие, имеющее доступ к сети.

Основная задача системы – это предупреждение производственного травматизма и помощь охране труда в предоставлении данных, но ее функционал также открывает перед работодателем дополнительные функции, например контроль рабочего времени, которые он может использовать по своему усмотрению.

Литература

1. Лаборатория энергетики. Кейс 17 [Электронный ресурс]. – Режим доступа: https://лабораторияэнергетики.рф/#rec420761871, свободный (дата обращения: 12.07.2022).

2. «Умные» браслеты для безопасности и контроля персонала [Электронный ресурс]. – Режим доступа: https://ohranatruda.ru/news/901/254732/, свободный (дата обращения: 12.07.2022).

3. Sitronics. Продукты. Умные часы [Электронный ресурс]. – Режим доступа: https://www.sitronics.com/products#smart-watch, свободный (дата обращения: 12.07.2022).

4. SmartTeam – Мониторинг здоровья [Электронный ресурс]. – Режим доступа: https://ctrl2go.solutions/solutions/smart-team/smart-team-health/, свободный (дата обращения: 12.07.2022).

5. Система мониторинга персонала SiWatch [Электронный ресурс]. – Режим доступа: http://ast-expert.by/news/sistema-monitoringa-personala-/, свободный (дата обращения: 12.07.2022).

6. Ермаков К.К., Виноградов А.Б. Разработка комплекса программного обеспечения для трансляции данных // XVII Всерос. (IX междунар.) науч.-техн. конф. студентов, аспирантов и молодых ученых «Энергия–2022»: матер. конф. – 2022. – Т. 4. – С. 48.

Виноградов Анатолий Брониславович
Д-р техн. наук, профессор, доцент каф. электропривода и автоматизации промышленных установок (ЭПиАПУ) Ивановского государственного энергетического ун-та (ИГЭУ)
Эл. почта: vinogr_ab@mail.ru

Горелкина Екатерина Владимировна
Магистрант каф. ЭПиАПУ ИГЭУ
Эл. почта: teterkinaejaterina97@gmail.com

Ермаков Кирилл Константинович
Аспирант каф. ЭПиАПУ ИГЭУ
Эл. почта: nice.ermakov19@gmail.com

УДК 004.9

Ю.О. Лобода, Н.С. Баулина, А.К. Алымкулова

Сравнительный анализ сайтов инновационной компании

Представлены результаты оценки сайтов ООО «НейроМех». Описаны критериеи и параметры, составлена сравнительная аналитическая таблица.

Ключевые слова: сайт, оценка, юзабилити, контент, аналитика, интеграция.

В настоящее время сайт является важным составляющим рекламы каждой компании. Сайт повышает имидж компании, помогает повысить уровень продаж и правильно подать нужную информацию. Однако не все сайты достигают оптимального результата. В этом случае нужно прибегнуть к тщательному анализу характеристик сайтов и выработке соответствующих рекомендаций.

В проекте оценивались сайты ООО «НейроМех»:

– НейроМех [1].

– NeVRoPlex [2].

– Обучающая система RAS [3].

– Нейробьюти [4].

Приведём краткое описание содержания проектов, размещённых на оцениваемых сайтах.

Нейромех

Сайт описывает деятельность ООО «НейроМех». ООО «Нейромех» разрабатывает программное обеспечение на основе искусственного интеллекта для нейроуправления.

Универсальное программное обеспечение, созданное с использованием нейросетей, позволяет управлять любой мехатронной системой с помощью человеческого мозга.

Разработки нейроконтроля используются в управлении роботами, в VR-индустрии и индустрии компьютерных игр, в медицине и в IoT.

NeVRoPlex

ООО «НейроМех» предлагает сайт, который представляет комплекс NeVRoPlex. Это программно-аппаратный комплекс по повышению производительности труда сотрудников компаний через снижение стресса. Комплекс создан на основе VR, нейротехнологий и технологий искусственного интеллекта. В комплекс включен модуль для повышения заинтересованности сотрудников.

Обучающая система RAS

РАС – автоматизированный комплекс для организации самостоятельной работы ребенка, основанный на машинном обучении, элементах искусственного интеллекта, нейротехнологиях.

Нейробьюти

Мобильное приложение «Нейробьюти» – это приложение, которое позволяет лучше ухаживать за кожей лица и добиться здорового, цветущего вида. Также помогает определить состояние кожи онлайн и предлагает консультацию с косметологом для выработки индивидуального подхода к уходу для разного типа кожи. Анализ состояния кожи и рекомендации проводятся с применением нейротехнологий.

Первое, что оценивает пользователь, – это дизайн, поскольку порядка 80% информации воспринимается визуально. Поэтому, получив сайт со спецэффектами и обилием картинок, многие считают, что задача уже выполнена, но это не так.

Основной характеристикой эффективной работы сайтов является посещаемость.

Во многих источниках эффективность работы сайтов производится по четырем показателям:

– конверсия;

– время пребывания на сайте;

– количество отказов;

– количество просмотренных страниц.

Когда сайт уже создан, то оценить время пребывания пользователя на сайте можно с помощью специальных инструментов [5].

Например, для оценки используют специальный код сайта, который вставляют в установленную программу «Яндекс.Метрика». Это бесплатный интернет-сервис компании Яндекс, предназначенный для оценки посещаемости веб-сайтов и анализа проведения пользователей. Можно также использовать и другие программы.

Однако можно проанализировать и опосредованные критерии оценки, которые важны при сравнительном анализе сайтов, если проекты находятся на разных уровнях разработки, как зачастую бывает с инновационными проектами.

Для сравнительного анализа сайтов ООО «НейроМех» были выбраны следующие опосредованные критерии:

– юзабилити, удобство использования – способность продукта быть понимаемым, изучаемым, используемым и привлекательным для пользователя в заданных условиях;

– внутреннее SEO (внутренняя оптимизация сайта) – это комплекс мер, направленных на улучшение восприятия ресурса поисковыми системами и пользователями. Включает оптимизацию кода и структуру сайта, работу с контентом и коммерческими факторами;

– контент (информационное содержание сайта);

– аналитика;

– интеграция [6].

Описание критериев параметров оценки сайтов

Юзабилити

В данный критерий входят параметры:

– адаптивная верстка;

– отдельная проработка юзабилити для каждого типа устройств. Ведь экран мобильного несравним с монитором компьютера;

– кнопки (они должны быть достаточно крупными, чтобы человеку не приходилось в них «целиться»);

– единые шрифты и стили на всех страницах;

– достаточная контрастность шрифта и фона страницы.

Внутреннее SEO

Данный критерий содержит параметры:

– Title и Description (заголовок страницы, описание), ключевые слова;

– релевантность сайта;

– соответствие структуры URL градации меню и карте сайта.

Контент

Все большее значение поисковые системы придают именно содержанию страниц. Поэтому нужно проверить и тексты, и изображения:

– изображения должны обладать достаточной контрастностью и сочетаться с дизайном сайта;

– текст должен легко читаться и восприниматься обычным человеком;

– все тексты должны быть уникальны;

– не должно быть орфографических и пунктуационных ошибок;

– в текстах должны быть призывы к действию;

– на каждом сайте должны быть заголовки H1–H6. H1 – это главный заголовок в статье, заголовок первого уровня. Каждая страница в интернете имеет свою структуру. Например, заголовки H1–H6 позволяют правильно структурировать материалы, причём как для поисковиков, так и для пользователей. H2–H6 – второстепенные, вложенные подзаголовки.

Аналитика

Данный критерий включает следующие требования:

– указаны виды деятельности и основные категории товаров;

– URL сайта ограничен приемлемой длиной.

Наличие этих настроек поможет в дальнейшем анализировать результативность рекламных кампаний.

Интеграция

Данный критерий должен иметь параметры:

– кнопки популярных социальных сетей;

– проверка заявок специалистами, поступление заявок в базу данных.

Выполнение всех этих пунктов гарантирует удобство работы с потенциальными клиентами [7].

Для проверки уникальности сайта, а также проверки орфографических и пунктуационных ошибок стоит использовать программу Антиплагиат. В ходе проверки программа непосредственно выдает ссылки сайтов. Поскольку выдаются ссылки на сайты ООО «НейроМех», это доказывает, что тексты полностью уникальны.

Чтобы сравнить сайты по проектам ООО «НейроМех», оценивались параметры по балльной системе: 1 – сайт содержит параметр, 0 – не содержит.

Непосредственно по баллам высчитывают наиболее соответствующие опосредованным требованиям сайты, в данном случае это: Обучающая система RAS (15), Нейробьюти (14) (табл. 1).

Таблица 1

Сравнительная аналитическая таблица сайтов

№	Критерии	Параметры	НейроМех	NeVRoPlex	Обучающая система RAS	Нейробьюти
1	Юзабилити	Адаптивная верстка	1	0	1	1
		Отдельная проработка юзабилити	1	0	0	0
		Кнопки	1	0	1	1
		Единые шрифты и стили	1	0	1	1
		Контрастность шрифта и страницы	1	1	1	1
2	Внутреннее SEO	Title и Description	1	1	1	1
		Релевантность сайта	1	1	1	1
		Структура URL	0	0	0	0
3	Контент	Читаемость, восприятие	1	1	1	1
		Уникальность	1	1	1	1
		Отсутствие орфографических и пунктуационных ошибок	0	0	1	0
		Призывы к действию	1	1	1	1
		Заголовки H1-H6	1	1	1	1
4	Аналитика	Программа для анализа посещаемости сайта	0	1	1	1
		Виды деятельности, основные категории товаров	1	1	1	1
		Длина URL	1	1	1	1
5	Интеграция	Кнопки популярных соц. сетей	0	0	0	1
		Мгновенное поступление заявок в базу данных	0	0	1	0

Оценки критериев распределились, как показано в табл. 2.

Таблица 2

Оценки критериев

Критерии	НейроМех	NeVRoPlex	Обучающая система RAS	Нейробьюти
Юзабилити	5	1	4	4
Внутреннее SEO	2	2	2	2
Контент	4	4	5	4
Аналитика	2	3	3	3
Интеграция	0	0	1	1

Список рекомендаций по каждому из проектов

НейроМех

– Добавление специальной программы для анализа посещаемости сайта [1].

NeVRoPlex

– Смена дизайна, подбор сочетаемых цветов.

– Исправление пунктуационных ошибок (добавление пробела после запятой).

– Единообразное форматирование (шрифтов и стилей).

– Отдельная проработка юзабилити для каждого типа устройств.

– Добавление специальной программы для анализа посещаемости сайта.

– Изменение стилей иллюстраций [2].

Обучающая система RAS

– Отдельная проработка юзабилити для каждого типа устройств.

– Добавление специальной программы для анализа посещаемости сайта [3].

Нейробьюти

– Отдельная проработка юзабилити для каждого типа устройств [4].

Вывод

Оценка сайта периодически необходима каждому проекту, ведь технологии поисковых систем неустанно развиваются, за счёт чего могут варьироваться алгоритмы ранжирования и другие показатели, влияющие на трафик и показатели качества работы сайта. Планируется дальнейшее развитие системы оценки сайтов, при необходимости можно добавлять различные весовые коэффициенты в зависимости от важности параметра.

Были изучены содержание и функционал сайтов ООО «НейроМех». С помощью критериев были проведён анализ и оценка по каждому из проектов, и выявлены наиболее соответствующий опосредованным требованиям сайт «Обучающая система RAS».

Литература

1. Сайт «НейроМех» [Электронный ресурс]. – Режим доступа: https://neuronmech.com/ru, свободный (дата обращения: 7.09.2022).

2. Сайт «NeVRoPlex» [Электронный ресурс]. – Режим доступа: https://nevroplex.com/, свободный (дата обращения: 8.09.2022).

3. Сайт «Обучающая система RAS» [Электронный ресурс]. – Режим доступа: https://ras-help.ru/, свободный (дата обращения: 9.09.2022).

4. Сайт «Нейробьюти» [Электронный ресурс]. – Режим доступа: https://neuro-beauty.ru/, свободный (дата обращения: 10.09.2022).

5. Применение математических моделей для оценки эффективности Web-сайтов [Электронный ресурс]. – Режим доступа: https://www.elibrary.ru/item.asp?id=26591024, свободный (дата обращения: 13.10.2022).

6. Список критериев для оценки эффективности сайтов [Электронный ресурс]. – Режим доступа: https://digitalrocket.ru/audit/oes/, свободный (дата обращения: 13.09.2022).

7. Параметры критериев оценки сайтов [Электронный ресурс]. – Режим доступа: https://dzen.ru/media/id/5bc5862fb7a6b100ac9084cf/tak-li-horosh-vash-sait-top52-kriteriev-ocenki-5f22de5042a52b7dd0497440, свободный (дата обращения: 16.09.2022).

Лобода Юлия Олеговна
Доцент каф. управления инновациями (УИ) ТУСУРа
Эл. почта: yulloboda@gmail.com

Баулина Наталья Сергеевна
Инженер каф. УИ ТУСУРа
Эл. почта: Baulina-Tasha217@yandex.ru

Алымкулова Анель Кубатовна
Студентка фак-та инновационных технологий (ФИТ) ТУСУРа
Эл. почта: technoakk.99087@gmail.com

СОДЕРЖАНИЕ

Секция 14

АВТОМАТИЗАЦИЯ И ОПТИМИЗАЦИЯ СИСТЕМ УПРАВЛЕНИЯ И ОБРАБОТКИ ИНФОРМАЦИИ

*Председатели секции – **Кориков Анатолий Михайлович**, профессор каф. АСУ, д.т.н.;*
***Катаев Михаил Юрьевич**, профессор каф. АСУ, д.т.н.*

А.Б. Кураков, А.Б. Гомбоин
Нейросетевой подход в задаче определения центра тяжести человека 7
Р.К. Карпов, М.Ю. Катаев
Методика оценивания цветового качества кирпичной продукции 10
В.Т. Тран, А.М. Кориков
Управление с использованием нейронной сети RBF трехосным карданным подвесом на беспилотных летательных аппаратах 14
А.Б. Гомбоин, А.Б. Кураков
Серверная часть веб-системы оценки походки человека по скелетной модели на языке Dart 19
М.Д. Пахмурин, В.С. Шелихов
Разработка программного обеспечения обработки DICOM-файлов 22
А.В. Соколов
Методика и программа оценки положения головы человека около монитора 24
В.В. Болдырев, М.А. Мельниченко, Н.А. Мурзин
Идентификация роботизированного технологического процесса механической обработки с применением графической нотации IDEF0 28
Д.А. Русаков
Методика и программа определения общего содержания газовой составляющей атмосферы 32
С.А. Кондратьев, А.К. Ершова
Разработка структуры web-сервиса центра космического монитринга Земли в ЦКМЗ ТУСУРа 35

Секция 15

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

*Председатель секции – **Шелупанов Александр Александрович**, президент ТУСУРа, профессор, д.т.н.*

Д.С. Лазорин, Д.И. Правиков
О подходах к обеспечению информационной безопасности цифровых двойников 41
Д.Е. Каширский, Ф.А. Якименко
Применение нейронных сетей для исправления ошибок в криптографических ключах, полученных методами квантовой криптографии 44
А.С. Звиденцева, Е.Ю. Костюченко
Использование методов ансамблирования для обнаружения DDoS-атаки 47
В.В. Ботвинко, Г.Р. Хазанкин
Разработка интерфейсного модуля для обработки сетевого трафика 51
Ю.Э. Корчагин, К.Д. Титов, О.Н. Завалишина
Различение двух узкополосных радиосигналов с неизвестными амплитудами и начальными фазами 54
Я.В. Калинкин, И.А. Тутов
Модуль информационной безопасности в сетях Modbus RTU 57
А.А. Чумаков, Р.Я. Панцыр, Н.О. Басков
Методы и средства защиты устройств средств вычислительной техники от несанкционированного доступа, реализуемого при помощи уязвимостей Intel Management Engine 61
А.Д. Калякин, А.С. Коваленко, В.С. Репкин, А.К. Новохрестов
Разработка сценария кибератаки на веб-портал предприятия 65
Н.С. Егошин, Н.И. Сермавкин, В.С. Репкин, А.Д. Калякин
Моделирование сценариев кибератак на базе киберполигогна Ampire 69
А.Ю. Якимук, С.А. Устинов, Т.П. Лазарев, А.С. Коваленко
Методы формализации описания сценариев кибератак 73

Секция 16

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В УПРАВЛЕНИИ И ПРИНЯТИИ РЕШЕНИЙ

*Председатель секции – **Сенченко Павел Васильевич**, проректор по УР, доцент каф. АОИ, к.т.н.*

А.Б. Кураков, Р.С. Кульшин, Г.А. Волокитин, С.В. Коновалов
Разработка языка шаблонов кодогенерации для информационной системы автоматизации разработки мобильных приложений 79

Г.А. Волокитин, Р.С. Кульшин, А.А. Сидоров, П.В. Сенченко
Использование решающих деревьев для задачи выбора товара в системах цифровых двойников 81

С.Е. Лопатин, Р.С. Кульшин, Г.А. Волокитин
Разработка системы тестирования мобильных приложений 84

Р.С. Кульшин, Г.А. Волокитин, А.А. Сидоров, П.В. Сенченко
Программное обеспечение для исследования алгоритмов рекомендательных систем 88

Р.С. Кульшин, И.Д. Тикшаев, А.А. Сидоров, П.В. Сенченко
Использование скоринговой модели оценки значимости действий пользователя для формирования персонализированных рекомендаций 91

Секция 17

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ОБУЧЕНИИ

*Председатели секции – **Дмитриев Вячеслав Михайлович**, профессор каф. КСУП, д.т.н.;*
***Ганджа Тарас Викторович**, профессор каф. КСУП, д.т.н.*

В.К. Шабалин, Д.А. Чебыкин, Д.Д. Пимонов, И.В. Климов, С.А. Краснов
Структура словаря веб-приложения для изучения японского языка 97

А.С. Болденков, Т.В. Ганджа
Применение дистанционных виртуальных лабораторий в обучении 100

Т.Н. Мосунова, В.А. Хамзина, Е.С. Костюк
Формирование процесса разработки корпоративных обучающих электронных тренажеров для сотрудников 103

А.А. Захарова, А.М. Аверьянова, К.Д. Глухих
Проблема автоматизации проведения нормоконтроля студенческих работ в вузах 106

Ф.А. Кун, Р.С. Герасимов, А.В. Цымбал, Д.Ю. Заигрин
Оценка эффективности достижений студентов 110

В.А. Семиглазов, Н.А. Романенко, В.А. Хамзина, Т.Н. Мосунова
Подбор инструментов для онлайн-продвижения курсов и оценки затрат по их применению для реализации проекта 113

Секция 18

ВИДЕОИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ЦИФРОВОЕ ТЕЛЕВИДЕНИЕ

*Председатели секции – **Курячий Михаил Иванович**, доцент каф. ТУ, к.т.н.;*
***Капустин Вячеслав Валериевич**, доцент каф. ТУ, к.т.н.*

С.А. Забуга, В.В. Капустин, А.Н. Шлыкович
Оценка эффективности методов повышения кадровой частоты видеопотока 119

В.С. Чебанов
Трансферное обучение модели SSD-MobileNet с использованием микрокомпьютера Jetson Nano 122

М.Е. Сукотнова, Г.Е. Эдель, В.В. Капустин
Идентификация дорожных знаков с использованием нейронных сетей 126

А.В. Поднебесный, Н.С. Герасимова, А.С. Захлебин
Исследование влияния изменения размера многоэлементной апертуры пятикаскадного фильтра на его быстродействие 130

А.А. Киселева
Исследование методов построения карт глубин по стереоизображениям 133

И.Д. Мусихин, В.В. Капустин
Исследование методов поиска особых точек для автоматического слежения за объектами интереса 137

С.А. Сальников
Обучение моделей нейронных сетей для автоматического обнаружения дорожных знаков 141
С.В. Сай, В.С. Никонов
Нейросетевой классификатор блоков изображения по признакам детальности 145
А.Н. Шлыкович, С.А. Забуга
Сравнение классических методов интерполяции изображения
с моделью искусственной нейронной сети SRGAN 148
Г.Е. Эдель, М.Е. Сукотнова, В.В. Капустин
Трансферное обучение нейронной сети для повышения точности распознавания детей
с использованием NVIDIA Jetson Nano 151
С.Е. Бояркин, А.В. Каменский
Адаптация метода Канни к работе с биомедицинскими изображениями 155

Секция 19
ЦИФРОВАЯ СВЯЗЬ И ВЕЙВЛЕТ-ФРАКТАЛЬНАЯ ОБРАБОТКА СИГНАЛОВ И ИЗОБРАЖЕНИЙ
*Председатель секции – **Голиков Александр Михайлович**, доцент каф. РТС, к.т.н.*

А.С. Дернова
Высокоскоростная система сотовой связи на базе беспилотной аэроплатформы 161
М.М. Муруева
Исследование системы мобильной связи UMTS на базе беспилотной аэроплатформы 164
А.М. Голиков, Д.С. Кирпичников, А.Н. Жариков
Обнаружение и оценка координат малоразмерных наземных объектов MIMO РЛС
с вейвлет-обработкой сигналов 167
Д.В. Скурихин, А.М. Голиков
Система передачи телеметрической информации LoRaWAN на базе беспилотной аэроплатформы 171

Секция 20
ВЫЧИСЛИТЕЛЬНЫЙ ИНТЕЛЛЕКТ И МАШИННОЕ ОБУЧЕНИЕ
*Председатель секции – **Ходашинский Илья Александрович**, профессор каф. КСУП, д.т.н.*

М.Б. Бардамова
Выбор меры оценки качества классификации для построения нечеткого классификатора
на несбалансированных данных 175
А.В. Герасимов
Исследование эффективности метаэвристики Lion Algorithm
при построении нечеткого классификатора несбалансированных данных 179
П.Г. Букина, Д.В. Грунвальд, С.С. Харченко
Определение психоэмоционального состояния человека на основе анализа речи
по частоте основного тона 183
К.Э. Горбунова, С.Г. Букина, С.С. Харченко
Обзор методов анализа речевых сигналов для оценки психоэмоционального состояния человека 186
С.С. Харченко, К.А. Качалов
Проблемы распознавания схожих изображений 189
Р.Е. Коломников, К.С. Сарин
Программный комплекс для аутентификации пользователя
по динамической рукописной подписи с помощью методов машинного обучения 192
Н.П. Корышев
Построение с помощью алгоритма «китов» нечёткого классификатора вторжений
по данным аудита жёсткого диска под управлением операционной системы Linux 196
Р.О. Остапенко, И.А. Ходашинский
Обнаружение сетевых атак с помощью нечёткого классификатора, построенного
с помощью метаэвристического алгоритма «саранчи» 199
А.А. Паскаль
Модификации алгоритма Future Search Algorithm для отбора признаков
в нечетком классификаторе 203

Е.Ю. Печенина, Н.В. Рузанов, В.А. Печенин
Разработка информационной системы подбора научных статей с использованием глубокого обучения 207

Д.С. Савельев
Отбор признаков с помощью метаэвристического алгоритма пустынных канюков 211

М.О. Светлаков
Применение нечеткого классификатора для аутентификации пользователя на основе данных электроэнцефалограммы 214

В.П. Турнаева
Отбор признаков с помощью метаэвристического алгоритма Rock Hyraxes Swarm / Алгоритм каменных даман 218

В.С. Ваняшин
Анализ существующих актуальных библиотек для выполнения задачи распознавания рукописного ввода 222

Д.А. Жуков, М.Ю. Катаев
Разработка программы анализа естественно-языковых ответов в программных системах тестирования 226

Секция 21

РОБОТОТЕХНИКА И МЕХАТРОНИКА

*Председатели секции – **Шурыгин Юрий Алексеевич**, директор департамента управления и стратегического развития ТУСУРа, зав. каф. КСУП, профессор, д.т.н.;*
***Коцубинский Владислав Петрович**, доцент каф. КСУП, к.т.н.*

И.Н. Булатникова, Н.Н. Гершунина, Е.С. Яковлева
Автоматизация проектирования алгоритмов для микропроцессорных систем 233

В.А. Казанин, И.Н. Минин, С.А. Ситников, А.В. Бадьин
Автономная сеть устройств удаленного мониторинга почвенно-климатических параметров в сельском хозяйстве 235

Е.В. Ланин, В.А. Казанин, А.В. Бадьин, К.В. Дорожкин
Концепция использования беспилотного автоматизированного дрона для организации беспроводной сети по протоколу LoraWAN в сельском хозяйстве 238

Секция 22

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ВЕБ-СИСТЕМ И МОБИЛЬНЫХ ПРИЛОЖЕНИЙ

*Председатель секции – **Зариковская Наталья Вячеславовна**, доцент каф. АОИ, к.ф.-м.н.*

С.А. Селезнев, Р.Е. Богачев, Н.В. Зариковская
Реализация клиентской части административной панели для системы онлайн-бронирования жилищных объектов 243

Н.С. Вершинина, З.А. Чепурова, Н.В. Зариковская
Процесс обеспечения качества мобильного приложения 246

Р.Е. Богачев, С.А. Селезнев, Н.В. Зариковская
Разработка модуля администратора серверной части веб-системы для терминалов 249

А.С. Стецко
Графическая среда программирования «ДРАКОН-Си». Редактор ДРАКОН-схем 253

А.Б. Виноградов, Е.В. Горелкина, К.К. Ермаков
Система мониторинга соблюдения техники безопасности и состояния здоровья сотрудников 256

Ю.О. Лобода, Н.С. Баулина, А.К. Алымкулова
Сравнительный анализ сайтов инновационной компании 259

Научное издание

ЭЛЕКТРОННЫЕ СРЕДСТВА И СИСТЕМЫ УПРАВЛЕНИЯ

XVIII Международная научно-практическая конференция

16–18 ноября 2022 г.

Материалы докладов

В двух частях

Часть 2

Electronic Devices and Control Systems

XVIII International Scientific-practical Conference
November 16–18, 2022
Part 2

Верстка В.М. Бочкаревой
Корректор В.Г. Лихачева

Оригинал-макет подготовлен в ООО «Издательство В-Спектр»
ИНН 7017129340
Сдано на верстку 11.07.2022. Заказ 17.

634055, Томск, пр. Академический, 13-24. Эл. почта: bvm-1@list.ru

---

Тираж отпечатан в типографии ТУСУРа
Подписано к печати 21.11.2022.
Формат $70\times100^{1}/_{16}$. Печать трафаретная.
Печ. л. 16,6. Тираж 100 экз.

634050, г. Томск, пр. Ленина, 40, к. 205, ГК.
Тел. (382-2) 70-15-24 (орг. комитет конф.).