

ПРИКАСПИЙСКИЙ ЖУРНАЛ



УПРАВЛЕНИЕ И ВЫСОКИЕ
ТЕХНОЛОГИИ

2024
№2 (66)



ISSN 2074-1707

АСТРАХАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ В. Н. ТАТИЩЕВА

ПРИКАСПИЙСКИЙ ЖУРНАЛ: управление и высокие технологии

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

2024

№ 2 (66)

Журнал включен в перечень рецензируемых научных изданий, рекомендованных ВАК России для публикации основных научных результатов диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук по следующим научным специальностям.

Группа специальностей 1.2 «Компьютерные науки и информатика»:

1.2.2 – Математическое моделирование, численные методы и комплексы программ (технические науки).

Группа специальностей 2.2 «Электроника, фотоника, приборостроение и связь»:

2.2.4 – Приборы и методы измерения (по видам измерений) (технические науки);

2.2.11 – Информационно-измерительные и управляющие системы (технические науки);

2.2.12 – Приборы, системы и изделия медицинского назначения (технические науки).

Группа специальностей 2.3 «Информационные технологии и телекоммуникации»:

2.3.1 – Системный анализ, управление и обработка информации (технические науки);

2.3.4 – Управление в организационных системах (технические науки);

2.3.5 – Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей (технические науки);

2.3.6 – Методы и системы защиты информации, информационная безопасность (технические науки).

Журнал входит в базу данных Ulrich's Periodicals Directory.

Астрахань

Астраханский государственный университет имени В. Н. Татищева

2024

Рекомендовано к печати редакционно-издательским советом
Астраханского государственного университета имени В. Н. Татищева

**ПРИКАСПИЙСКИЙ ЖУРНАЛ:
управление и высокие технологии**
НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

**2024
№ 2 (66)**

Редакционная коллегия

И.М. Азмухамедов, доктор технических наук, профессор, декан факультета цифровых технологий и кибербезопасности, профессор кафедры «Информационная безопасность» Астраханского государственного университета имени В. Н. Татищева (главный редактор)

И.В. Аникин, доктор технических наук, профессор, заведующий кафедрой «Системы информационной безопасности» Казанского национального исследовательского технического университета им. А.Н. Туполева – КАИ

А.А. Большаков, доктор технических наук, профессор, профессор кафедры «Системы автоматизированного проектирования и управления» Санкт-Петербургского государственного технологического института (технического университета)

Л.А. Демидова, доктор технических наук, профессор, профессор кафедры «Вычислительной и прикладной математики» Рязанского государственного радиотехнического университета (г. Рязань)

А.С. Катасёв, доктор технических наук, профессор, профессор кафедры систем информационной безопасности ФГБОУ ВО «Казанский национальный исследовательский технический университет им. А.Н. Туполева – КАИ» (г. Казань)

И.Ю. Квятковская, доктор технических наук, профессор, директор Института информационных технологий и коммуникаций Астраханского государственного технического университета

А.Г. Кравец, доктор технических наук, профессор, профессор кафедры «Системы автоматизированного проектирования и поискового конструирования» Волгоградского государственного технического университета

В.Ю. Кузнецова, кандидат технических наук, старший преподаватель кафедры информационной безопасности Астраханского государственного университета имени В. Н. Татищева

Ю.В. Литовка, доктор технических наук, профессор, профессор кафедры «Системы автоматизированной поддержки принятия решений» Тамбовского государственного технического университета

А.М. Лихтер, доктор технических наук, профессор, заведующий кафедрой «Общая физика» Астраханского государственного университета имени В. Н. Татищева

А.А. Лобатый, доктор технических наук, профессор, заведующий кафедрой «Информационные системы и технологии» Белорусского национального технического университета (Республика Беларусь, г. Минск)

Е.В. Никольцев, доктор технических наук, профессор, профессор кафедры «Управление и моделирование систем» Московского технологического университета (МИРЭА) (г. Москва)

В.О. Осипян, доктор физико-математических наук, доцент, профессор кафедры «Информационные технологии» Кубанского государственного университета (г. Краснодар)

И.Ю. Петрова, доктор технических наук, профессор, первый проректор Астраханского государственного архитектурно-строительного университета, заведующая кафедрой САПР Астраханского государственного архитектурно-строительного университета

А.В. Рыбаков, кандидат физико-математических наук, директор «Физико-математического института» Астраханского государственного университета имени В. Н. Татищева; доцент кафедры электротехники, электроники и автоматизации Астраханского государственного университета имени В. Н. Татищева

А.В. Скрипаль, доктор физико-математических наук, профессор, заведующий кафедрой «Медицинская физика» Саратовского национального исследовательского государственного университета имени Н.Г. Чернышевского

И.Б. Старченко, доктор технических наук, профессор, ООО «Параметрика», научный руководитель (г. Таганрог Ростовской области)

Т.Л. Тен, доктор физико-математических наук, профессор кафедры «Информационно-вычислительные системы» Карагандинского экономического университета (Республика Казахстан, г. Караганда)

Е.Н. Тищенко, доктор экономических наук, профессор, заведующий кафедрой «Информационные технологии и защита информации» Ростовского государственного экономического университета (РИНХ) – г. Ростов-на-Дону

С.А. Филлист, доктор технических наук, профессор, профессор кафедры «Биомедицинская инженерия» Юго-Западного государственного университета (г. Курск)

Л.Р. Фионова, доктор технических наук, профессор, декан факультета Вычислительной техники, заведующая кафедрой «Информационное обеспечение управления и производства» Пензенского государственного университета

В.А. Цимбал, заслуженный деятель науки РФ, доктор технических наук, профессор, профессор кафедры «Автоматизированные системы управления» (Филиал Военной академии РВСН им. Петра Великого МО в г. Серпухов Московской области)

Н.К. Юрков, заслуженный деятель науки РФ, доктор технических наук, профессор, заведующий кафедрой «Конструирование и производство радиоаппаратуры» Пензенского государственного университета

N.A. Kolesova, PhD, Check Point Software Technologies LTD, Tel-Aviv, Israel

Serg Miranda, PhD (Toulouse University, France), – Master thesis at UCLA (University of California, Los Angeles with an INRIA Scholarship), Professor of Computer Science, University of Nice – Sophia Antipolis (Nice, France), Director of the CS dept. and MBDS innovation lab (www.mbds-fr.org)

Журнал выходит 4 раза в год
Все материалы, поступающие в редколлегию журнала,
проходят независимое рецензирование

© Астраханский государственный университет,
имени В.Н. Татищева, 2024
© Гайфитдинова С. Ю., дизайн обложки, 2024

ASTRAKHAN TATISHCHEV STATE UNIVERSITY

**PRIKASPIYSKIY ZHURNAL:
Upravlenie i Vysokie Tekhnologii**

**CASPIAN JOURNAL:
Control and High Technologies**

A SCIENTIFIC AND TECHNICAL JOURNAL

**2024
No. 2 (66)**

The journal is included in the list of the reviewed scientific journals recommended by VAK of Russia for the publication of the main scientific results of theses for the candidate of science degree, for the doctor of science degree on the following scientific specialties.

Group of specialties 1.2 “Computer science and informatics”:

1.2.2 – Mathematical modelling, numerical methods and complexes of programmes (technical sciences).

Group of specialties 2.2 “Electronics, photonics, instrument engineering and communication”:

2.2.4 – Instruments and methods of measurement (by type of measurement) (technical sciences);

2.2.11 – Information-measuring and control systems (technical sciences);

2.2.12 – Medical devices, systems and products (technical sciences).

Group of specialties 2.3 “Information technologies and telecommunications”:

2.3.1 – System analysis, information control and processing (technical sciences);

2.3.4 – Management in organizational systems (technical sciences);

2.3.5 – Mathematical software and software for computing systems, complexes and computer networks (technical sciences);

2.3.6 – Information security methods and systems, information security (technical sciences).

The journal is included into the database Ulrich’s Periodicals Directory.

Astrakhan
Astrakhan Tatishchev State University
2024

Recommended by the Editorial and Publishing Board
of Astrakhan Tatishchev State University

**CASPIAN JOURNAL:
Control and High Technologies**

A SCIENTIFIC AND TECHNICAL JOURNAL

**2024
No. 2 (66)**

Editorial Board

I.M. Azhmukhamedov, Doct. Sci. (Engineering), Professor, Dean of the Faculty of Digital Technologies and Cybersecurity, Professor of Information Security Department, Astrakhan Tatishchev State University (**Editor-in-Chief**)

I.V. Anikin, Doct. Sci. (Engineering), Professor, Head of Information Security System Department, Federal State Budgetary Educational Institution of Higher Education «Kazan National Research Technical University named after A.N. Tupolev – KAI»

A.A. Bolshakov, Doct. Sci. (Engineering), Professor of «Systems of Automated Design Engineering and Control» department, St. Petersburg State Technological Institute (Technical University)

L.A. Demidova, Doct. Sci. (Engineering), Professor, Professor of the Computational and Applied Mathematics Department, Ryazan State Radio Engineering University (Ryazan)

A.S. Katasev, Doct. Sci. (Engineering), Associate Professor, Professor of the Department of Information Security Systems, Kazan National Research Technical University. A.N. Tupolev - KAI "(Kazan)

I.Yu. Kvyatkovskaya, Doct. Sci. (Engineering), Professor, Head of "Information Technologies and Communications" Institute of the Astrakhan State Technical University

A.G. Kravets, Doct. Sci. (Engineering), Professor, Professor of the Automated Design Engineering Systems and Search Constructing Department, Volgograd State Technical University

V.Yu. Kuznetsova, Cand. Sci. (Engineering), Senior Lecturer of Information Security Department, Astrakhan State University named after V.N. Tatishchev

Yu.V. Litovka, Doct. Sci. (Engineering), Professor, Professor of the Department of Automated Support System for Decision-Making, Tambov State Technical University

A.M. Likhner, Doct. Sci. (Engineering), Professor, Head of the Department of General Physics, Astrakhan Tatishchev State University

A.A. Lobaty, Doct. Sci. (Engineering), Professor, Head of Information Systems and Technologies Department, Belarusian National Technical University (Belarus, Minsk)

E.V. Nikulchev, Doct. Sci. (Engineering), Professor, Professor of the System Management and Modeling Department, Moscow Technological University (Moscow)

V.O. Osipyan, Doct. Sci. (Physics and Mathematics), Professor of the Kuban State University (Krasnodar)

I.Yu. Petrova, Doct. Sci. (Engineering), Professor, First Vice-Rector of the Astrakhan State Architectural and Construction University, Head of the CAD department of Astrakhan State Architectural and Construction University

A.V. Rybakov, Cand. Sci. (Physics and Mathematics), Director of the Institute of Physics and Mathematics, Astrakhan Tatishchev State University

A.V. Skripal, Doct. Sci. (Physics and Mathematics), Professor, Head of Medical Physics Department of the Saratov national research State University named after N.G. Chernyshevsky

I.B. Starchenko, Doct. Sci. (Engineering), Professor, OOO «Parametrica» (Taganrog, Rostov Oblast), Research Supervisor

T.L. Ten, Doct. Sci. (Engineering), Professor, Karaganda Economic University (Republic of Kazakhstan, Karaganda)

E.N. Tishchenko, Doct. Sci. (Economics), Professor, Head of the Information Technologies & Information Security Department, Rostov State University of Economics, Rostov-on-Don

S.A. Filist, Doct. Sci. (Engineering), Professor, Professor of Biomedical Engineering Department, Southwest State University (Kursk)

L.R. Fionova, Doct. Sci. (Engineering), Professor, Dean of the Computer Technology Faculty, Head of the Department «Information Support of Management and Production, Penza State University

V.A. Tsimbal, Doct. Sci. (Engineering), Honored Worker of Science of the Russian Federation, Professor, Professor of the Automated Control Systems Department (Branch of the Military Academy of the Russian Strategic Missile Forces named after Peter the Great of the Moscow Oblast, Serpukhov, Moscow Oblast)

N.K. Yurkov, Honored worker of science of the Russian Federation, Doct. Sci. (Engineering), Professor, Head of the department «Designing and production of the radio equipment», Penza State University

N.A. Kolesova, PhD, Check Point Software Technologies LTD, Tel-Aviv, Israel

Serg Miranda, PhD (Toulouse University, France), – Master thesis at UCLA (University of California, Los Angeles with an INRIA Scholarship), Professor of Computer Science dept., University of Nice – Sophia Antipolis (Nice, France), Director of the CS department and MBDS innovation lab (www.mbd-fr.org)

The journal is published four times a year
All materials that come to the Editorial Board of the journal
are subject to independent peer-review

© Astrakhan Tatishchev State University, 2024
© S. Yu. Gayfitdinova, cover design, 2024

СОДЕРЖАНИЕ

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

А. А. Пономарев, Д. С. Соловьев, И. А. Соловьева

Реализация спектрального анализа в задачах обработки аудиосигнала на микроконтроллерах семейства AVR 9–15

К. Н. Кургузкин, А. Н. Марьенков

Сравнительный анализ методик сопоставления информации в контексте интеграции схем данных 16–31

Р. И. Нежников, А. Н. Марьенков

Сравнительный анализ моделей трансформера для классификации неструктурированной текстовой информации 32–38

УПРАВЛЕНИЕ В ОРГАНИЗАЦИОННЫХ СИСТЕМАХ

А. А. Зубова, И. М. Ажмухамедов

Проблема формирования проектных команд в IT-сфере 39–45

О. И. Евдошенко, Н. И. Евдошенко

Разработка модели автоматического классификатора обращений граждан 46–52

К. Е. Лебедева, И. М. Ажмухамедов

Информационная поддержка процесса оказания благотворительной помощи 53–63

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ

Т. Е. Тюлькова, П. Ф. Чернавин, Н. П. Чернавин

Диагностика туберкулеза без бактериовыделения методами случайного леса и комитетов с минимизацией числа входных признаков 64–70

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

В. И. Петренко, Ф. Б. Тебуева, Д. Г. Волошин

Метод построения траектории полета беспилотных летательных аппаратов в условиях кибератак GPS/Глонасс спуфинга 71–80

М. Г. Огур

Математическая модель реализации многовекторных атак на IoT-системы на основе анализа потока сетевого трафика 81–92

МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ МАШИН, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ СЕТЕЙ

Л. Э. Семенова, А. В. Хайтул, Т. М. Ишкин

Программное обеспечение для автоматизированного подбора заданий студентам по предмету «Начертательная геометрия» 93–101

Ю. А. Смирнова, В. В. Скрипник, Ю. А. Головко
Алгоритм генерации тестов для олимпиадных задач
по программированию 102–110

И. И. Гордеев, В. А. Черкасова, И. А. Бубенщикова, С. Е. Алферин
Улучшенный программный компонент визуализации
графов фреймворка GOLEM 111–119

Ю. Ю. Калинина, Ю. А. Смирнова, Ю. А. Головко
Информационная система формирования ведомостей
учета работы преподавателей высших учебных заведений 120–126

ПРИБОРОСТРОЕНИЕ, МЕТРОЛОГИЯ
И ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ
ПРИБОРЫ И СИСТЕМЫ

ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ
И УПРАВЛЯЮЩИЕ СИСТЕМЫ

К. Г. Деримедведь
Развитие информационных систем
для локальных электроэнергетических объектов 127–132

Д. В. Старов, В. А. Корякова
Интеллектуальная система управления поливом
сельскохозяйственных участков 133–141

ПРАВИЛА ДЛЯ АВТОРОВ 142

CONTENTS

INFORMATICS, COMPUTER TECHNIQUE AND CONTROL

SYSTEM ANALYSIS, CONTROL AND INFORMATION PROCESSING

A. A. Ponomarev, D. S. Solovjev, I. A. Solovjeva

Implementation of spectral analysis in audio signal processing tasks on avr family microcontrollers9–15

K. N. Kurguzkin, A. N. Marenkov

Comparative analysis of comparison methods information in the context of data schema integration16–31

R. I. Nezhnikov, A. N. Marenkov

Comparative analysis of transformer models for classification of unstructured text information32–38

MANAGEMENT IN ORGANIZATIONAL SYSTEMS

A. A. Zubova, I. M. Azhmukhamedov

The problem of forming project teams in the IT field39–45

O. I. Evdoshenko, N. I. Evdoshenko

Development of a model of an automatic classifier of citizens' appeals46–52

K. E. Lebedeva, I. M. Azhmukhamedov

Information support for the process of providing charitable assistance53–63

MATHEMATICAL MODELLING, NUMERICAL METHODS AND PROGRAM SYSTEMS

T. Y. Tyulkova, P. F. Chernavin, N. P. Chernavin

Diagnostics bacteriologically non-confirmed tuberculosis by the forest method and the committee machine method with a decrease in the number of injected signs64–70

METHODS AND SYSTEMS OF INFORMATION PROTECTION, INFORMATION SECURITY

V. I. Petrenko, F. B. Tebueva, D. G. Voloshin

Method for planing an unmanned aerial vehicle flight path in conditions of gps/ghonass spoofing cyberattacks71–80

M. G. Ogur

A mathematical model for implementing multi-vector attacks on IoT systems based on analysis of network traffic flow81–92

MATHEMATICAL SOFTWARE AND SOFTWARE FOR COMPUTING MACHINES, COMPLEXES AND COMPUTER NETWORKS

L. E. Semenova, A. V. Khaytul, T. M. Ishkin

Software for automated selection of assignments for students in the subject «Descriptive geometry»93–101

Yu. A. Smirnova, V. V. Skripnik, Yu. A. Golovko
Algorithm of test generation for olympiad programming tasks 102–110

I. I. Gordeev, V. A. Cherkasova, I. A. Bubenschikova, S. E. Alferin
Improved graph visualization software component
of the golem framework 111–119

Yu. Yu. Kalinina, Yu. A. Smirnova, Yu. A. Golovko
Information system for formation of work reports of teachers
of higher educational institutions 120–126

**INSTRUMENT ENGINEERING, MEASUREMENT SCIENCE,
INFORMATION AND MEASURING DEVICES AND SYSTEMS**

INFORMATION-MEASURING AND CONTROL SYSTEMS

K. G. Derimedved
Development of information systems
for local electric power facilities 127–132

D. V. Starov, V. A. Koryakova
Intelligent watering system for agricultural areas 133–141

RULES FOR THE AUTHORS 142

ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ

СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ

УДК 004.02

РЕАЛИЗАЦИЯ СПЕКТРАЛЬНОГО АНАЛИЗА В ЗАДАЧАХ ОБРАБОТКИ АУДИОСИГНАЛА НА МИКРОКОНТРОЛЛЕРАХ СЕМЕЙСТВА AVR

Пономарев Артур Анатольевич, Тамбовский государственный университет имени Г.П. Державина, 392036, Российская Федерация, г. Тамбов, ул. Интернациональная, 33, магистрант, ORCID: 0009-0002-7372-5363, e-mail: caprimores@gmail.com

Соловьев Денис Сергеевич, Тамбовский государственный университет имени Г.П. Державина, 392036, Российская Федерация, г. Тамбов, ул. Интернациональная, 33, кандидат технических наук, доцент, ORCID: 0000-0001-6613-3218, e-mail: solovjevdenis@mail.ru

Соловьева Инна Александровна, Тамбовский государственный университет имени Г.П. Державина, 392036, Российская Федерация, г. Тамбов, ул. Интернациональная, 33, кандидат технических наук, старший преподаватель, ORCID: 0000-0002-1798-1859, e-mail: good.win32@yandex.ru

В настоящее время наблюдается активный рост применения микроконтроллеров при изготовлении прототипов устройств. Целью исследования является исследование возможностей микроконтроллера AVR ATmega328P в задачах спектрального анализа аудиосигнала. В качестве методов спектрального анализа использовались преобразования Фурье и Хартли, а также аппаратные фильтры на базе интегральных микросхем MSGEQ7 и BA3834. Рассмотрены детали процесса оцифровки аудиосигнала с учетом архитектурных особенностей ATmega328P. Продемонстрированы ограниченные возможности микроконтроллера при использовании программных методов и перспективы реализации систем динамического анализа с малым количеством частот.

Ключевые слова: микроконтроллер, методы спектрального анализа, амплитудно-частотные ряды, оцифровка аудиосигнала, быстрое преобразование Фурье, быстрое преобразование Хартли, ATmega328P, MSGEQ7, BA3834

IMPLEMENTATION OF SPECTRAL ANALYSIS IN AUDIO SIGNAL PROCESSING TASKS ON AVR FAMILY MICROCONTROLLERS

Ponomarev Arthur A., Derzhavin Tambov State University, 33 Internatsionalnaya St., Tambov, 392036, Russian Federation, master's student, ORCID: 0009-0002-7372-5363, e-mail: caprimores@gmail.com

Solovjev Denis S., Derzhavin Tambov State University, 33 Internatsionalnaya St., Tambov, 392036, Russian Federation,

Cand. Sci. (Engineering), Associate Professor, ORCID: 0000-0001-6613-3218, e-mail: solovjevdenis@mail.ru

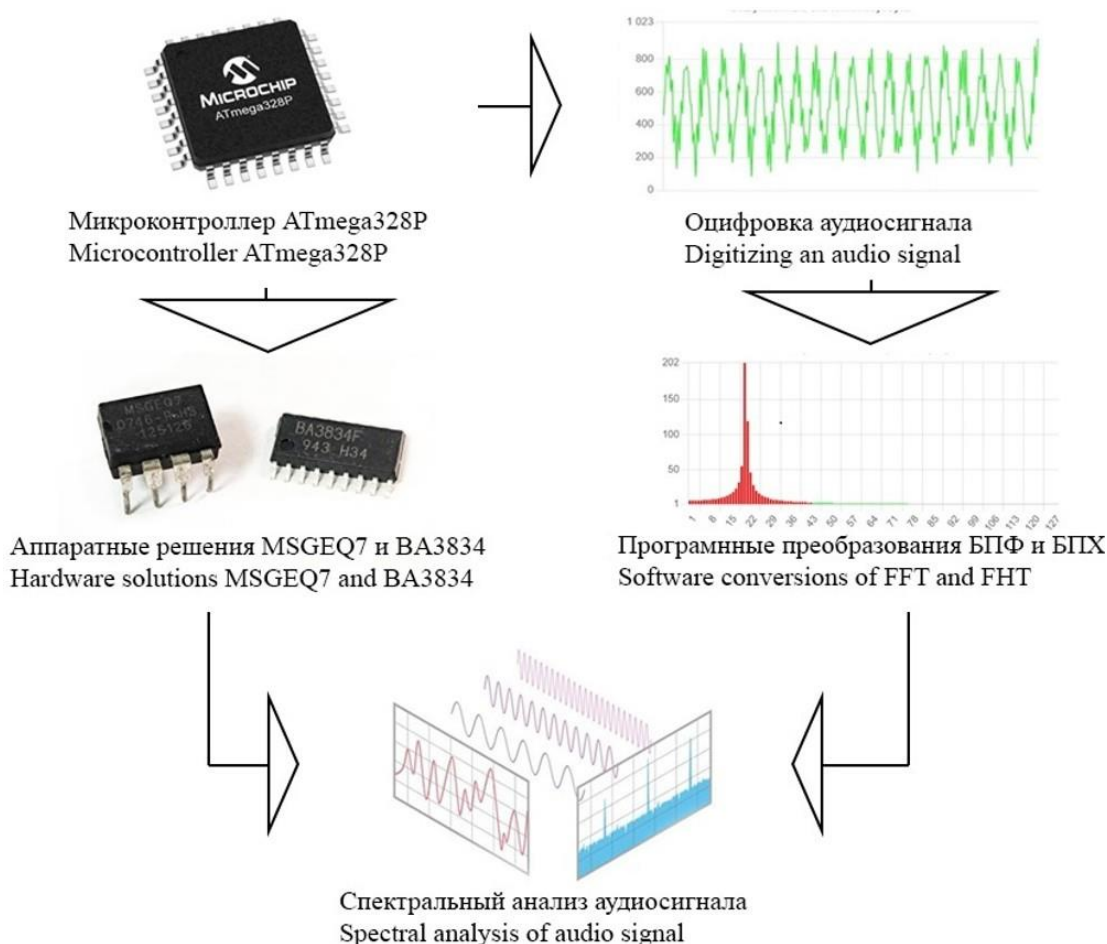
Solovjeva Inna A., Derzhavin Tambov State University, 33 Internatsionalnaya St., Tambov, 392036, Russian Federation,

Cand. Sci. (Engineering), Senior Lecturer, ORCID: 0000-0002-1798-1859, e-mail: good.win32@yandex.ru

Currently, there is an active growth in the use of microcontrollers in the manufacture of prototype devices. The aim of the study is to investigate the capabilities of the AVR ATmega328P microcontroller in the tasks of spectral analysis of an audio signal. Fourier and Hartley transforms, as well as hardware filters based on integrated circuits MSGEQ7 and BA3834, were used as spectral analysis methods. The details of the audio signal digitization process are considered, taking into account the architectural features of the ATmega328P. The limited capabilities of the microcontroller when using software methods and the prospects for the implementation of dynamic analysis systems with a small number of frequencies are demonstrated.

Keywords: microcontroller, spectral analysis methods, amplitude-frequency series, audio signal digitization, fast Fourier transform, fast Hartley transform, ATmega328P, MSGEQ7, BA3834

Graphical annotation (Графическая аннотация)

**ВВЕДЕНИЕ**

В последние десятилетия наблюдается активный рост применения микроконтроллеров (МК) в различных областях науки и исследований. Это обусловлено не только развитием технологий в области микроэлектроники, но и расширением возможностей МК для автоматизации и контроля экспериментов. Так, в работе [1] приводится пример реализации электронного блока управления для машины Атвуда на базе STM32, который возможно применять при педагогической деятельности и проведении экспериментов. Также перспективной областью применения МК является создание различного рода датчиков для сельского хозяйства, метрологии и производств. Разработка датчика влажности и температуры воздуха при использовании AVR ATTINY2313, который выполняет функции измерения аналоговых показаний и реализует обработку полученных данных, производя фильтрацию, а также генерирует выходной широтно-импульсно модулированный сигнал (ШИМ-сигнал) для конечных устройств, описана в работе [2]. МК в том числе применяются при создании цифровых сигнальных процессоров (DSP), анализирующих и обрабатывающих сигналы в режиме реального времени. В работе [3] продемонстрирована разработка анализатора звукового спектра на базе МК, который оцифровывает и очищает от шума аналоговый сигнал, а также с помощью быстрого преобразования Хартли получает частотный спектр, выводимый на светодиодный контроллер (LED-драйвер).

Спектральный анализ аудиосигнала представляет собой метод изучения частотной составляющей звука, в результате которого получают амплитудно-частотную характеристику (ряды) или спектрограммы, отображающие зависимость амплитуд составляющих частот, а в случае спектрограмм – и времени. Он активно применяется в обработке речевой информации, анализе музыкальных композиций, а также при разработке и проектировании аудио и светомузыкальных систем. Для получения спектральных характеристик речи традиционно применяются методы дискретного и быстрого преобразования Фурье (ДПФ и БПФ), которые позволяют получить амплитудный спектр голосового сигнала [4]. В работе [5] приводится пример созданного прибора для идентификации механических колебаний, состоящего в том числе из анализатора спектра звука на базе микросхемы MSGEQ7, данные от которой впоследствии применяются в нейронной сети.

Исходя из роста популярности применения встраиваемых систем на базе МК AVR ATmega328p, необходимо исследование возможностей её применения для реализации задач частотного анализа аудиосигнала.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Для проведения спектрального анализа аудиосигнала традиционно применяются два способа получения частотных характеристик: программные методы или аппаратные решения. При использовании программных методов необходима предварительная оцифровка, в ходе которой непрерывные значения аудиосигнала с помощью аналого-цифровых преобразователей (АЦП) преобразуются в дискретную форму и сохраняются в виде выборок в оперативной памяти. Важными характеристиками процесса оцифровки являются: частота дискретизации, размер выборки сигнала, разрядность (битность) сигнала. Согласно теореме Котельникова, частота дискретизации сигнала должна быть в два раза выше его максимальной частоты, вследствие чего для аудиосигнала составляет величину, равную 40000 Гц [6, 7].

Для программного преобразования дискретного сигнала к спектральным характеристикам, помимо ДПФ и БПФ, применяют дискретное преобразование Хартли (ДПХ) и быстрое преобразование Хартли (БПХ), которые имеют идентичные алгоритмы, за исключением ядра преобразования [6]. Фазовый коэффициент, используемый в обоих преобразованиях, определяется согласно:

$$\omega = \frac{2\pi}{T}, \quad (1)$$

где ω – фазовый коэффициент, рад;

T – период сигнала, с.

В преобразованиях Фурье используется экспоненциальное ядро вида:

$$e^{-\omega t i} = \cos(\omega t) - i \sin(\omega t), \quad (2)$$

где t – момент времени, с;

i – мнимая единица.

В преобразованиях Хартли ядром является сумма косинуса и синуса одного аргумента:

$$\text{cas}(\omega t) = \cos(\omega t) + \sin(\omega t). \quad (3)$$

Из ядер преобразований (2), (3) видно, что в ядре Фурье используется коэффициент i , отвечающий за фазу сигнала, а в ядре Хартли он отсутствует, в связи с этим преобразование Хартли способны производить трансформации только в области рациональных чисел, но за меньшее время из-за меньшего числа расчетов. Однако возможность работы с фазой сигнала может предоставлять дополнительные возможности при анализе периодичности сигнала, а в случае аудиосигнала – интерференции в сложных средах, определении направления в пространстве. В связи с вышесказанным преобразования Фурье имеют большую гибкость и универсальность, но и более сложные алгоритмы реализации за счет применения комплексных чисел. Дискретные преобразования Фурье и Хартли описываются согласно:

$$F_k = \sum_{n=0}^{N-1} x_n e^{-\omega k n}, \quad (4)$$

$$H_k = \sum_{n=0}^{N-1} x_n \text{cas}(\omega k n), \quad (5)$$

где F_k – амплитуда Фурье по индексу k , дБ;

H_k – амплитуда Хартли по индексу k , дБ;

x_n – дискретное значение сигнала по индексу n , дБ;

N – размер выборки сигнала, шт.

Быстрые методы преобразований Фурье и Хартли отличаются от дискретных в применении «бабочки», которая позволяет избежать повторных расчетов, используя прореживание по времени или частотам, вследствие чего асимптотика сложности алгоритма изменяется с $O(N^2)$ на $O(N \log N)$ [7]. Однако у них присутствует ограничение на размер выборки сигнала, равного $N = 2^k$, от которого зависит шаг частоты спектральных рядов, вычисляемый согласно:

$$S = \frac{F}{N}, \quad (6)$$

где S – шаг частоты, Гц;

F – частота дискретизации, Гц;

Альтернативой программным методам получения спектральных характеристик являются аппаратные решения в виде аналоговых фильтров, на вход которых подается исходный сигнал, а выход подключается к АЦП, считывающему не значение сигнала, а амплитудную характеристику. Итоговые системы могут быть достаточно громоздкими по размеру и лишены гибкости и требуют также значительное количество аналоговых входов МК. Главный плюс применения аналоговых фильтров – крайне малые вычислительные нагрузки, проявляющиеся только в необходимости оцифровки полученных значений амплитуд. Существует класс интегральных микросхем в виде многополосных фильтров с мультиплексированным выходом, которые решают как проблему громоздкости, так и необходимости большого количества АЦП [9, 10]. Данные микросхемы хоть и имеют ограниченное количество аналоговых фильтров, но компактность и наличие в них мультиплексора, позволяющего переключаться между фильтрами с помощью логических выводов и считывать значения с фильтров, лишь используя один аналоговый вывод, обеспечивает их применимость.

ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

Для программного анализа аудиосигнала на основе преобразований Фурье и Хартли МК программировался в среде разработки Arduino IDE, в которой по умолчанию частота дискретизации АЦП равна 9,6 кГц. Изменив значения регистра ADCSRA, частота дискретизации была повышена до 66 кГц, а использование микросекундных задержек между единичными считываниями позволило понизить её к 40 кГц [9]. Для сохранения выборки аудиосигнала использовался массив, состоящий из 256 значений типа `int16_t`, который по итогу занимал 512 байт (25 % всей оперативной памяти МК). Выход аудиосигнала в устройствах реализован таким образом, что его положительные и отрицательные значения измеряются относительно нуля, а АЦП МК ATmega328P может считывать положительные значения только в диапазоне 0–5 В [8], потому при процессе оцифровки нижняя полуволна считываться не будет (рис. 1). Для смещения сигнала использован компаратор на базе интегральной схемы LM393.

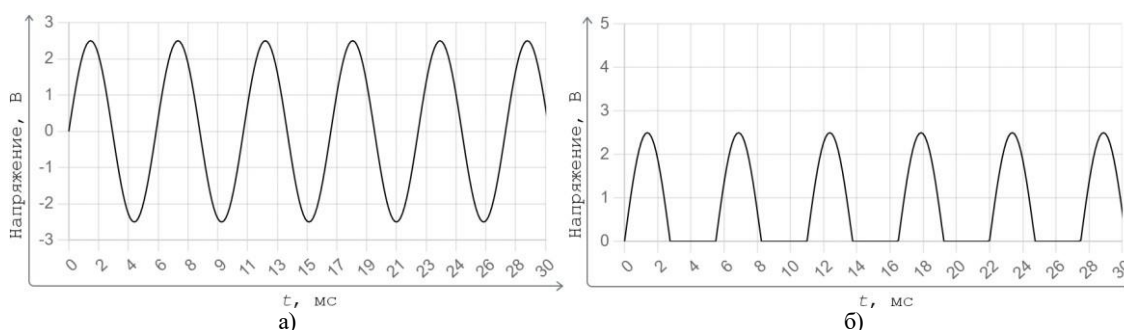


Рисунок 1 – Реальный (а) и считываемый (б) микроконтроллером сигнал без компаратора

Для преобразований сигнала к амплитудно-частотным рядам применялись библиотеки: `ArprohFFT` – приближенное БПФ, использующее в расчетах только целочисленные типы, в том числе в тригонометрических функциях; `ExactFFT` – точное БПФ, в основе расчетов которого используются значения с плавающей точкой; `FHT` – реализация БПХ на языке ассемблера, обладающая высокой скоростью вычисления. Алгоритм получения амплитуд в библиотеке `ExactFFT` реализован на основе операций с вещественными числами (`float`), которые МК вычисляет с помощью программной эмуляции, что обуславливается отсутствием у него модуля для операций с плавающей точкой (FPU) [8]. Перечисленные решения были протестированы с целью определения скорости расчета спектральных характеристик и процента ошибок, возникающих при преобразованиях.

Для аппаратного получения спектральных характеристик МК ATmega328p был соединен с интегральными микросхемами `MSGEQ7` и `BA3834`, имеющими по 7 аналоговых фильтров и настроенными на схожие частоты (усредненно: 65 Гц; 165 Гц; 410 Гц; 1 кГц; 2,5 кГц; 6 кГц; 15 кГц), но отличающимися по размеру, способу подключения и алгоритму переключения частот [9, 10]. Разработано программное обеспечение (ПО) для получения частотных характеристик от данных интегральных схем, где `MSGEQ7` выделилась более простым алгоритмом взаимодействия из-за циклического переключения между аналоговыми фильтрами, а для `BA3834` были применены побитовые операции (рис. 2). Ее выводы *A*, *B*, и *C* при переключении фильтров получают значения 3, 2 и 1 бита числа. Например, число 6 (соответствует 6 каналу) имеет двоичное представление 110, где третий разряд соответствует каналу *A* (логическая единица), второй – каналу *B* (логическая единица), первый – каналу *C* (логический ноль).

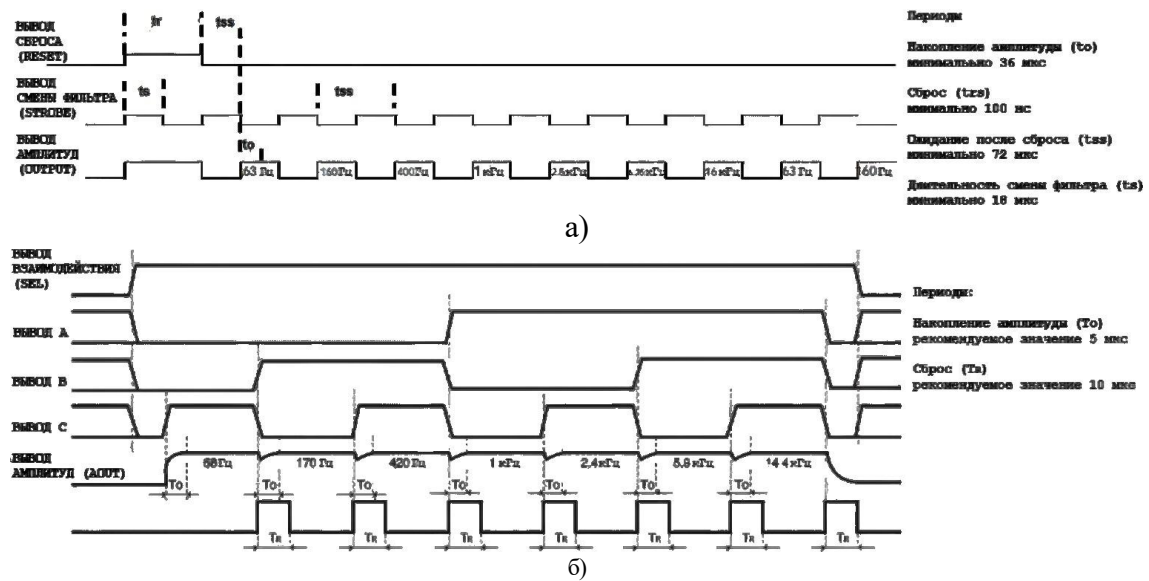


Рисунок 2 – Осциллограммы управления микросхемами MSGEQ7 (а) и VA3834 (б)

Для наглядного сравнения пиковых значений амплитуд и наличия постоянной составляющей, на представленные аппаратные фильтры с помощью генератора подавался аудиосигнал, а частоты выбирались исходя из характеристик фильтров с фиксацией значений.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В процессе оцифровки требуется учитывать смещение и усиление сигнала, так, чем выше разрешение получаемых амплитуд, тем ниже процент ошибок при дальнейших расчетах. Для усиления сигнала можно понижать опорное напряжение АЦП, но в таком случае будет необходимо изменять смещение сигнала. Слишком высокая частота дискретизации может стать проблемой при анализе низких частот, так как для анализа частоты 40 Гц при частоте дискретизации 44,1 кГц потребуется 1024 значения сигнала, которые при типе данных int16_t для хранения займут всю оперативную память МК. К тому же, БПФ и БПХ либо вовсе не способны будут обработать такое количество данных, либо сделают выполнение этой задачи на порядки дольше. Оптимальным размером выборки сигнала для данного МК является 256 значений. Однако в силу такой небольшой выборки необходимо подстраивать частоту дискретизации под анализ конкретных частот. Для оптимального частотного анализа всего звукового спектра, частоту дискретизации требуется перенастроить на 20 кГц. В таком случае шаг частот, согласно (3) составит около 80 Гц, а верхний порог по теореме Котельникова – 10 кГц.

Результаты тестирования скорости преобразования библиотек ApproxFFT, ExactFFT, FHT представлены в таблице 1. В таблице 2 приводятся данные о проценте ошибок, возникающих в ходе вычислений.

Таблица 1 – Длительность расчета реализаций БПФ и БПХ в зависимости от размера выборки сигнала

Размер выборки, шт	ApproxFFT, мс	ExactFFT, мс	FHT, мс
64	9,3	25,4	0,9
128	22,1	55,1	2,1
256	52,1	117,4	4,7

Таблица 2 – Процент ошибки результатов БПФ и БПХ в зависимости от размера выборки сигнала

Размер выборки, шт	ApproxFFT, %	ExactFFT, %	FHT, %
64	$5 \cdot 10^{-3}$	$1 \cdot 10^{-15}$	9,8
128	$1,3 \cdot 10^{-6}$	$3,1 \cdot 10^{-15}$	12,3
256	$3,9 \cdot 10^{-6}$	$2,5 \cdot 10^{-16}$	15,7

В тех случаях, когда спектральный анализ необходимо проводить достаточно быстро, например при создании светомузыкальных установок, от которых требуется оперативно реагировать на изменения в музыке световыми эффектами, – лучше использовать библиотеку FHT, а остальные будут серьезно замедлять систему (табл. 1). Для высокой точности вычислений с низким процентом ошибок лучше воспользоваться библиотекой ExactFFT, а для баланса скорости и точности – ApproxFFT (табл. 2).

При создании динамичных светомузыкальных систем и многополосных графических анализаторов предпочтительнее использовать аппаратные фильтры в виде микросхем MSGEQ7 или ВА3834, так как они позволяют получать амплитуды без компаратора и практически не используя вычислительные ресурсы МК. На рисунке 3 представлены пиковые значения амплитуд.

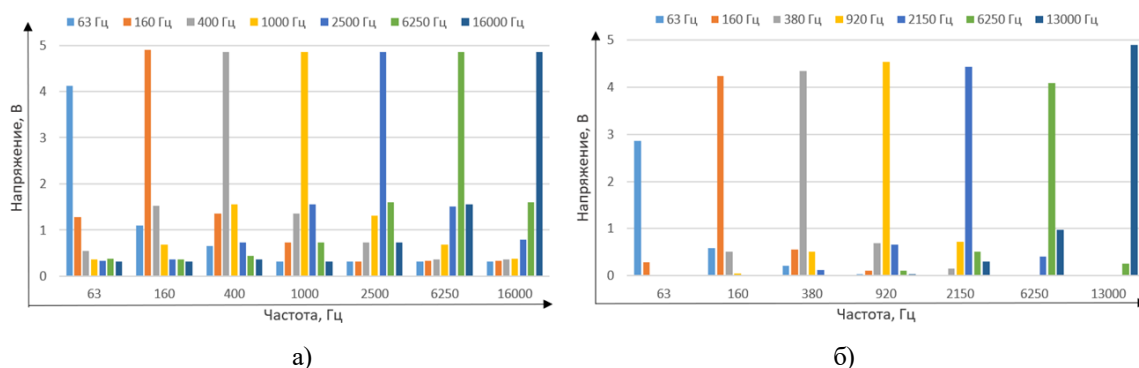


Рисунок 3 – Пиковые значения амплитуд фильтров MSGEQ7 (а) и ВА3834 (б)

Анализ значений показывает, что микросхема ВА3834 позволяет получать амплитуды с меньшей постоянной составляющей и низким взаимопроникновением каналов в сравнении с MSGEQ7. Поскольку обе микросхемы имеют ограниченное число фильтров и алгоритм их получения статичен, минимальная длительность получения всех амплитуд для MSGEQ7 составляет 504 мкс, а для ВА3834 – 70 мкс.

ЗАКЛЮЧЕНИЕ

В настоящее время МК ATmega328P может применяться при спектральном анализе аудиосигнала как с помощью программных реализаций в виде БПФ и БПХ, так и с применением аналоговых фильтров в виде интегральных микросхем. Однако области его применения значительно ограничены вычислительными возможностями при использовании преобразований Фурье и Хартли, вследствие чего разработка устройств с быстрым и точным анализом большого числа частот затруднительна. Использование МК ATmega328P в задачах спектрального анализа в режиме реального времени оправдано для аппаратных решений из класса интегральных микросхем многополосных фильтров с мультиплексированным выходом. Так, микросхема ВА3834 показывает лучшие результаты в сравнении с MSGEQ7, поскольку обладает низким показателем зашумленности выходного сигнала, имеет более гибкий алгоритм управления и меньшую длительность расчета.

Список источников

1. Сухомлинов, Н. М. Применение микроконтроллерных систем в исследованиях (на примере машины Атвуда) / Н. М. Сухомлинов, А. Г. Финогеев, Т. В. Смирнова, В. Д. Иващенко, Д. С. Парыгин // Прикаспийский журнал: управление и высокие технологии. – 2022. – № 3 (59). – С. 112–121.
2. Вострухин, А. В. Микроконтроллерный датчик влажности и температуры воздуха / А. В. Вострухин, В. С. Ядыкин, М. А. Ерина, Л. Н. Королькова, П. Ю. Пташкин, М. Ф. Горяинов // Прикаспийский журнал: управление и высокие технологии. – 2011. – № 2 (14). – С. 21–25.
3. Zhyvaha, Y. Cost Efficient Audio Spectrum Analyzer: Design and Study / Y. Zhyvaha, H. Klym, R. Dunets // 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS). – 2021. – Vol. 1. – P. 432–437.
4. Евсюков, М. В. Методы защиты в современных системах голосовой аутентификации / М. В. Евсюков, М. М. Пулято, А. С. Макарян, В. О. Немчинова // Прикаспийский журнал: управление и высокие технологии. – 2022. – № 3 (59). – С. 84–92.
5. Liang, J. S. Vibration feature extraction using audio spectrum analyzer based machine learning / J. S. Liang, K. Wang // 2017 International Conference on Information, Communication and Engineering (ICICE). – Xiamen, China, 2017. – P. 381–384.
6. Тропченко, А. Ю. Цифровая обработка сигналов. Методы предварительной обработки. / А. Ю. Тропченко, А. А. Тропченко. – Санкт-Петербург : СПбГУ ИТМО, 2009. – 100 с.
7. Сергиенко, А. Б. Цифровая обработка сигналов / А. Б. Сергиенко. – Санкт-Петербург : БХВ-Петербург, 2011. – 768 с.
8. Техническая спецификация ATmega48PA/88PA/168PA/328P. – 2009. – URL: <https://files.amperka.ru/datasheets/ATmega328.pdf> (дата обращения: 21.01.2024)
9. Техническая спецификация MSGEQ7. – 2004. – URL: <https://www.sparkfun.com/datasheets/Components/General/MSGEQ7.pdf> (дата обращения: 21.01.2024).
10. Техническая спецификация ВА3834. – URL: <https://labkit.ru/userfiles/file/documentation/Audioprocessor/BA3834.pdf> (дата обращения: 21.01.2024).

References

1. Sukhomlinov, N. M., Finogeev, A. G., Smirnova, T. V., Ivashchenko, V. D., Parygin, D. S. Application of microcontroller systems in research (using the example of Atwood's machine). *Caspian Journal: Control and High Technologies*, 2022, no. 3 (59), pp. 112–121 (In Russ.).
2. Vostrukhin, A. V., Yadykin, V. S., Erina, M. A., Korolkova, L. N., Ptashkin, P. Yu., Goryainov, M. F. Microcontroller humidity and air temperature sensor. *Caspian Journal: Control and High Technologies*, 2011, no. 2, pp. 21–25. (In Russ.).
3. Zhyvaha, Y., Klym, H., Dunets, R. Cost Efficient Audio Spectrum Analyzer: Design and Study. *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 2021, pp. 432–437.
4. Evsyukov, M. V., Putyato, M. M., Makaryan, A. S., Nemchinova, V. O. Protection methods in modern voice authentication systems. *Caspian Journal: Control and High Technologies*, 2022, no. 3 (59), pp. 84–92. (In Russ.).
5. Liang, J. S., Wang, K. Vibration feature extraction using audio spectrum analyzer based machine learning. *2017 International conference on Information, Communication and Engineering (ICICE)*. Xiamen, China, 2017, pp. 381–384.
6. Tropchenko, A. Yu., Tropchenko, A. A. Digital signal processing. Pre-processing methods. St. Petersburg, SPbGU ITMO, 2009. 100 p. (In Russ.).
7. Sergienko, A. B. Digital signal processing. St. Petersburg: BHV-Petersburg, 2011. 768 p. (In Russ.).
8. Datasheet ATmega48PA/88PA/168PA/328P. Available at: <https://files.amperka.ru/datasheets/ATmega328.pdf> (accessed 21.01.2024) (In Russ.).
9. Datasheet MSGEQ7. Available at: <https://www.sparkfun.com/datasheets/Components/General/MSGEQ7.pdf> (accessed 21.01.2024) (In Russ.).
10. Datasheet BA3834. Available at: <https://labkit.ru/userfiles/file/documentation/Audioprocessor/BA3834.pdf> (accessed 21.01.2024) (In Russ.).

Статья поступила в редакцию 16.02.2024; одобрена после рецензирования 18.03.2024; принята к публикации 22.03.2024.

The article was submitted 16.02.2024; approved after reviewing 18.03.2024; accepted for publication 22.03.2024.

УДК 004.622

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МЕТОДИК СОПОСТАВЛЕНИЯ ИНФОРМАЦИИ В КОНТЕКСТЕ ИНТЕГРАЦИИ СХЕМ ДАННЫХ

Кургузкин Кирилл Николаевич, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
аспирант, ORCID: 0009-0007-2983-0613, e-mail: kirill_kurguzkin@mail.ru

Марьенков Александр Николаевич, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат технических наук, доцент, ORCID: 0000-0003-1378-3553, e-mail: marenkovan17@gmail.com

Процесс сопоставления схем считается одним из наиболее важных этапов интеграции и реструктуризации данных в информационных системах. Основная цель этого процесса заключается в выявлении взаимосвязей между схемами. Одной из основных проблем в области сопоставления схем является обеспечение поддержки принятия решений при возникающей синтаксической и семантической неоднородности в источниках данных, а также при наличии сложной структуры сопоставления. Было предпринято множество попыток выявления соответствия атрибутов в процессе сопоставления схем, на что указывают опубликованные многочисленные исследования в данной области. В данной статье представлена классификация методик в области сопоставления схем данных с акцентом на выделенный в классификации графовый подход сопоставления. Также были проанализированы техники сопоставления схем и произведено тестирование наиболее популярных методик и программных средств с применением различных сценариев. Средняя результативность проанализированных методик сопоставления составила приблизительно 26 процентов, однако в ряде случаев некоторые методики («OpenII», «Coma-3.0», «Rondo») показали удовлетворительные результаты – от 40 до 60 процентов верно определенных связей. Рассмотренные в ходе экспериментов решения недостаточно хороши, таким образом требуются дальнейшие исследования, направленные на оптимизацию и модернизацию существующих методик и алгоритмов.

Ключевые слова: управление данными, интеграция баз данных, реструктуризация схем, сопоставление схем, графовый подход, синтаксический анализ сущностей, семантический анализ сущностей, классификация методик сопоставления, практическое тестирование

COMPARATIVE ANALYSIS OF COMPARISON METHODS INFORMATION IN THE CONTEXT OF DATA SCHEMA INTEGRATION

Kurguzkin Kirill N., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

graduate student, ORCID: 0009-0007-2983-0613, e-mail: kirill_kurguzkin@mail.ru

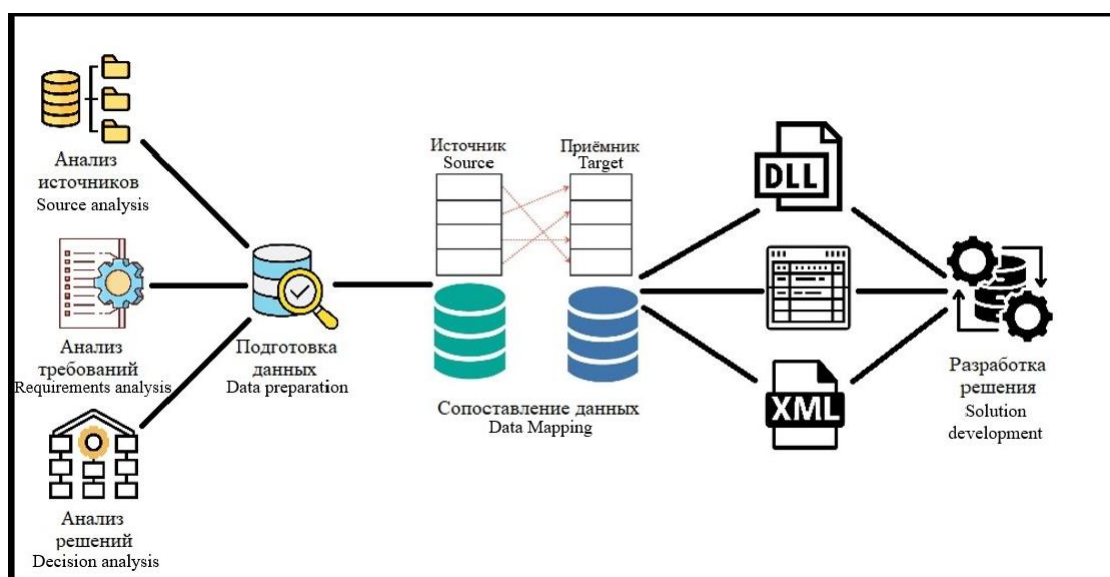
Marenkov Alexander N., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

Cand. Sci. (Engineering), Associate Professor, ORCID: 0000-0003-1378-3553, e-mail: marenkovan17@gmail.com

Schema matching is considered a crucial stage in data integration and restructuring within information systems. Its primary aim is to uncover relationships between schemas. One of the key challenges in schema matching is ensuring decision support amid syntactic and semantic disparities across data sources and complex matching structures. Numerous attempts have been made to identify attribute correspondences during schema matching, as evidenced by various published studies. This article presents a classification of data schema matching techniques, with a focus on the high-lighted graph-based matching approach. Additionally, it analyzes schema matching techniques and conducts testing on popular methods and tools across different scenarios. The average effectiveness of the analyzed matching techniques was approximately 26 %. However, in some instances, certain methods such as "OpenII," "Coma-3.0," and "Rondo" demonstrated satisfactory results, with correct link identification ranging from 40 to 60 %. The solutions explored during experiments were deemed not good enough, indicating a need for further research aimed at optimizing and modernizing existing techniques and algorithms.

Keywords: data management, database integration, schema restructuring, schema matching, graph-based approach, syntactic entity analysis, semantic entity analysis, classification of matching techniques, practical testing

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

В свете современной информационной эпохи и интенсивно развивающихся процессов информатизации стремительно растет объем информации, эффективный анализ и обработка которой зачастую является нетривиальной и трудоемкой задачей.

Согласно опубликованному исследованию в июне 2023 г. в журнале Fortune Business Inside [1], прогнозируется значительный рост рынка аналитики больших данных. Прогнозируемый среднегодовой темп роста рынка составит 13,5 % как минимум до 2030 г.

Другое исследование, проведенное изданием Actioner [2], указывает на прогнозируемый рост рынка интеграции данных на 12,5 % ежегодно до 2032 г. Подчеркивается, что почти 40 % проектов сталкиваются с проблемами, связанными с интеграцией различных наборов данных.

Соответственно, учитывая данные различных исследований, все более остро встает вопрос об ускорении и удешевлении процесса аналитики данных.

Ключевым этапом в аналитике зачастую является процедура сопоставления реляционных схем данных. Современные реляционные хранилища данных в основном построены с применением на стандарте ANSI SQL или его ответвлениях. Фактически является общепринятым для всех современных промышленных систем управления базами данных [3].

Процесс анализа и классификации сущностей в процессе интеграции реляционных баз данных представляет собой трудоемкую задачу, требующую внимательного изучения. Эта задача становится особенно сложной в контексте того, что различные источники могут использовать разные схемы данных. Анализ сущностей включает в себя выделение ключевых элементов данных, определение их типов, а также выявление связей между ними. Классификация данных подразумевает группировку сущностей по общим характеристикам для последующего создания единой структуры данных. Этот процесс требует не только технического понимания структуры данных, но и контекстуального понимания бизнес-логики, чтобы обеспечить корректное сопоставление и взаимодействие между различными источниками информации. Таким образом, трудоемкость этой задачи вытекает из необходимости подробного анализа, тщательной обработки данных и стратегического подхода к созданию устойчивой и эффективной системы интеграции баз данных.

Актуальность задачи сопоставления сущностей подтверждает ежегодное проведение конкурса «Ontology Alignment Evaluation Initiative» [4]. Основным направлением конкурса является оценка сильных и слабых сторон систем согласования и сравнение эффективности методик согласования. Также на актуальность исследований в данном направлении указывает ежегодное проведение соревнований по сопоставлению данных [5].

На основании всего вышесказанного можно с уверенностью отметить, что задача оптимизации методов анализа, интеграции и реструктуризации данных является актуальной.

Цель работы: проанализировать существующие методики и алгоритмы сравнения сущностей схем баз данных.

КРАТКИЙ АНАЛИЗ РАБОТ В ОБЛАСТИ СОПОСТАВЛЕНИЯ ДАННЫХ

Проблемы в области сопоставления рассматриваются учеными с конца прошлого века и до сих пор остаются актуальными.

Первые работы в данном направлении, например [6], рассматривали фундаментальные принципы управления данными, одним из которых является принцип интеграции различных данных через границы приложений и организаций.

Интеграция схем базы данных – это работа по объединению схем существующих или предполагаемых баз данных в общую, объединенную схему.

В начале XXI века накопленный учеными в данной области опыт позволил создать ряд классических методик интеграции схем баз данных, в основе которых лежат методики сравнения сущностей.

В исследовании [7] приведена обзорная статья основных методик, а также классификация методик по различным параметрам, такими как уровни и типы сопоставлений. Общей чертой указанных в данной статье методик является наличие в основе каждой анализа входящих сущностей как взаимосвязанных структурных элементов-графов.

Современные исследования, такие как [8–13], подтверждают актуальность работ в данном направлении и указывают на то, что проблема сопоставления и интеграции схем баз данных остается открытой.

Чтобы приступить к практическому анализу методик сравнения схем, необходимо составить их общую классификацию, учитывая современные тенденции и опыт указанных выше исследований.

КЛАССИФИКАЦИЯ МЕТОДИК СОПОСТАВЛЕНИЯ

На основе анализа работ по интеграции схем базы данных была проведена классификация методик сравнения схем, данная классификация представлена на рисунке 1. Рассмотрим подробно каждый элемент данной классификации.

Экспертный метод сопоставления сущностей представляет собой подход, при котором процесс сопоставления выполняется с участием экспертов-аналитиков в соответствующей области. Аналитики вручную рассматривают схемы данных или онтологии, выявляют сходства и устанавливают соответствия между соответствующими элементами. Однако, несмотря на высокую степень точности и контроля, при таком подходе характерен высокий уровень трудоемкости. Процесс требует значительных усилий и времени со стороны экспертов, особенно при работе с обширными и сложными схемами данных. Такие методы могут быть эффективными в случаях, когда высокая точность сопоставления критически важна, но их применение может быть ограничено из-за затрат на ресурсы.

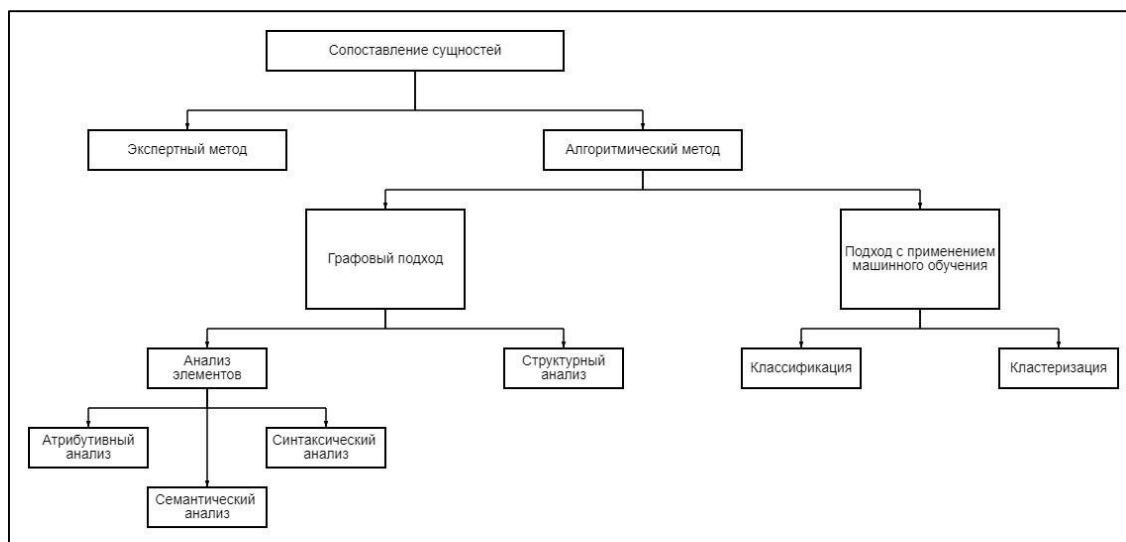


Рисунок 1 – Классификация методик сопоставления сущностей

Алгоритмический метод применяется в совокупности с экспертным методом и позволяет значительно упростить процедуры анализа, интеграции и реструктуризации данных.

Все основные алгоритмические методики сопоставления сущностей на сегодняшний день делятся на две большие группы: методики, применяющие графовый подход к анализу сущностей, и методики, применяющие машинное обучение.

Графовый подход включает в себя возможность анализа связей между сущностями. Если исходные и целевые схемы содержат достаточное количество информации об ограничениях, это может помочь алгоритму или методике сопоставления идентифицировать схожесть между схемами

и обеспечить точное соответствие атрибутов схемы. Таким образом, данный подход уделяет внимание структуре схемы для выявления схожести между атрибутами.

Графовый подход также может включать в себя анализ непосредственно самих сущностей. Основная идея заключается в том, чтобы использовать лингвистическую информацию, такую как имена атрибутов и их описания, для определения соответствия между схемами.

Семантический анализ сущностей – это подход, который уделяет внимание смысловому содержанию данных и использует формальные методы для интерпретации семантических отношений между элементами данных.

В качестве примера можно привести латентный семантический анализ, который использует сопоставление слов с семантической схожестью. Латентно-семантический анализ отображает документы и отдельные слова в так называемое «семантическое пространство», в котором и производятся все дальнейшие сравнения.

Также в качестве семантического анализа можно рассмотреть применение различных семантических баз данных, например WordNet/Тезаурус или Google Similarity.

Синтаксический анализ сущностей – это построение структуры, которая позволяет приблизиться к некоторому формализованному представлению смысла текста. Но, в отличие от «глубокой» семантической структуры, которая строится в результате семантического анализа, синтаксическая структура обычно не связывает конструкции естественного языка с их значениями в некоторой предметной области. Большинство моделей синтаксической структуры предложения опираются на грамматику составляющих текста. В ходе синтаксического анализа исходный текст преобразуется в структуру данных, обычно – в дерево, которое отражает синтаксическую структуру входной последовательности и хорошо подходит для дальнейшей обработки [14].

В качестве примера синтаксического анализа можно привести синтаксические N-граммы, включающие последовательное разделение слов или текстов на токены, а также регулярные выражения, используемые для определения шаблонов для атрибутов схемы и сопоставления схем на основе этих классов шаблонов.

Атрибутивный анализ сущностей представляет собой анализ каждого атрибута по дополнительным признакам. В рамках стандарта SQL такими дополнительными атрибутами могут являться типы данных, размерность, а также ограничения типа «NULL/NOT NULL» или установленные значения атрибута по умолчанию. Подходы к определению связей при таком анализе могут быть позаимствованы из синтаксического или семантического анализа сущностей.

При структурном анализе элементов графов сущности рассматриваются в контексте их взаимоотношения «родитель – потомки». Схожесть узлов вычисляется на том основании, что два нелистовых элемента схемы структурно схожи, если множества их детей являются схожими. Также при таком виде анализа учитываются ограничения типа «внешние ключи».

Структурный анализ также может рассматривать входные данные графов с учетом их семантической интерпретации. Примерами являются методы решения задачи выполнимости булевой формулы и методы рассуждения в описательной логике. Чтобы сопоставить две схемы, используются так называемые метаданные, описывающие эти схемы, такие как имя схемы, имя таблицы, количество атрибутов, максимальная длина пути и т. д. Далее эти показатели анализируются и агрегируются в один коэффициент, который оценивает сходство между ними.

Подход с применением машинного обучения в области сопоставления схем представляет собой инновационный метод, в котором алгоритмы обучения используются для автоматического извлечения и обобщения закономерностей в данных. Этот подход позволяет анализировать разнообразные структуры схем, улучшая точность и эффективность сопоставления. С использованием обученных моделей машинного обучения можно достичь более глубокого понимания контекста данных.

Классификация сущностей – это построение модели машинного обучения, способной присваивать новые объекты к заранее определенным категориям или классам. То есть классификация занимается прогнозированием меток классов для новых данных на основе обучающего набора данных. Классификация сущностей может проводиться по таким же принципам, как и анализ сущностей в графовом подходе: семантически и синтаксически. Примерами данного подхода сущностей является классификация на основе построения деревьев решений синтаксического анализа или нахождения расстояния Левенштейна для семантического анализа.

Кластеризация сущностей – это группировка объектов в кластеры на основе их схожести без использования предварительно определенных меток классов. В общем случае, в рамках сравнения сущностей, выбирается лучший размер кластера, который измеряет, насколько объект похож на свой собственный кластер по сравнению с другими кластерами. Примером кластеризации может являться подход, описанный в [12] с применением метода Кохонена – нейронной сети, обучаемой методом без учителя для кластеризации схожих паттернов во входных данных.

Также следует отметить, что большинство методик используют комбинированные подходы для достижения более эффективного результата.

Теперь, когда была разработана классификация методик сравнения сущностей, выберем наиболее популярные методики и составим их описание с учетом составленной классификации.

АНАЛИЗ МЕТОДИК СОПОСТАВЛЕНИЯ

Рассмотрим методику сопоставления сущностей под названием «Similarity Flooding» [15]. Данная методика сопоставления данных основана на вычислении контрольных точек, которые можно использовать в различных сценариях, не только при сравнении элементов баз данных. Алгоритм принимает два графа (схемы, каталоги или другие структуры данных) в качестве входных данных и создает в качестве выходных данных отображение между соответствующими узлами графов. В зависимости от цели сопоставления с помощью фильтров выбирается подмножество сопоставления.

Опишем подробнее данный алгоритм. Он представляет собой последовательность шагов для определения соответствий между таблицами и столбцами в двух SQL-схемах данных (S_1 и S_2).

Первый шаг – это преобразование SQL-схем в графы, которое осуществляется по формуле (1):

$$G_1 = \text{SQL2Graph}(S_1); G_2 = \text{SQL2Graph}(S_2), \quad (1)$$

где S_1 и S_2 – реляционные схемы данных;

«SQL2Graph» – функция преобразования реляционной схемы в граф.

Каждая таблица и столбец представлены узлами в графе, а связи между ними отражают отношения.

Второй шаг: применение алгоритма сопоставления строк («String Matching») для создания начального отображения между графами G_1 и G_2 :

$$\text{initialMap} = \text{StringMatch}(G_1, G_2), \quad (2)$$

где «StringMatch» – функция синтаксического сравнения сущностей на основе сходства суффиксов и префиксов.

Таким образом, начальное отображение представляет собой предположительные соответствия между элементами двух графов на основе схожести строк.

Третий шаг: производится объединение графов G_1 и G_2 на основании начального отображения, согласно формуле (3):

$$\text{product} = \text{SFJoin}(G_1, G_2, \text{initialMap}), \quad (3)$$

где «SFJoin» – это функция объединения, где узлы и связи соединяются в единый граф для определения сходства между элементами схем.

Алгоритм определения сходства основан на предположении, что при обнаружении сходства между двумя узлами графа, схожесть смежных элементов увеличивается. Данный алгоритм называется авторами [15] алгоритмом вычисления неподвижной точки.

Четвертый шаг: применяются пороговые значения для выбора соответствий согласно формуле (4), которые считаются достаточно сильными:

$$\text{result} = \text{SelectThreshold}(\text{product}), \quad (4)$$

где «SelectThreshold» – это функция подбора пар узлов на основе результата вычисления схожих элементов, которые являются «наиболее правдоподобными».

Авторы [15] предлагают определять такие пары узлов путем решения задачи стабильных браков.

В итоге данная методика автоматизирует процесс сопоставления таблиц и столбцов в двух схемах данных с использованием методов преобразования в графы, сопоставления строк и определения соответствий с использованием пороговых значений. Согласно разработанной классификации, данную методику можно отнести к графовому подходу с применением структурного анализа и синтаксического анализа элементов.

Несмотря на то, что описанный алгоритм нахождения сходства между сущностями представляет собой интересный и эффективный подход к сопоставлению моделей данных, у него присутствуют некоторые недостатки. Например, ограниченность обработки сложных случаев. В некоторых сложных сценариях, при наличии большого числа перекрывающихся структурных элементов, алгоритм сталкивается с трудностями в правильном сопоставлении. Ограниченность в типах данных и структурах: алгоритм менее эффективен для определенных типов данных или сложных структур, которые не поддерживаются внутренней моделью данных.

Далее рассмотрим методику «S-MATCH» [16]. «S-MATCH» – это система сопоставления схем на семантической основе. Она принимает две графоподобные структуры, такие как XML-схемы или онтологии, и возвращает семантические отношения между узлами графов, которые семантически соответствуют друг другу.

«S-MATCH» анализирует смысл текста в элементах и структурах схем/онтологий. Основная концепция методики заключается в переводе элементов входящих схем на внутренний язык,

используемый для выражения связей между элементами. В качестве внутреннего авторами был выбран логический предикативный, где атомарные формулы – концепции, записанные как одиночные слова, а сложные формулы получаются путем комбинирования атомарных концепций с использованием связок теории множеств. Авторы предлагают следующие определения для возможных семантических отношений: эквивалентность ($=$), более общее (\sqsubseteq), менее общее (\supseteq), несоответствие (\perp), перекрытие (\cap). На каждом шаге работы методики для всех элементов схем/онтологий устанавливаются или уточняются перечисленные семантические отношения.

Алгоритм поиска сопоставлений организован в четыре основных шага.

Шаг 1. Для всех атрибутов L в двух деревьях вычислить CL (концепцию атрибута). На данном этапе применяется поиск сопоставлений согласно словарю WordNet – лексической базе данных, разработанной в Принстонском университете, для работы с текстом, составленным на английском языке.

Шаг 2. Для всех узлов N в двух деревьях вычислить CN (концепцию сущности). На данном этапе все отношения между сущностями переводятся в предикативные связки, после чего по формуле (5) производится проверка на допустимость:

$$\text{Context} = \text{rel}(C_i, C_j), \quad (5)$$

где C_i – концепция узла в первом графе;

C_j – концепция узла во втором сопоставляемом графе;

rel – семантическое отношение, преобразованное в предикативную связку.

Таким образом «Context» – конъюнкция всех отношений (удачно переведенных в предикативную связку) между концепциями меток C . Ключевая идея заключается в том, что проблема сопоставления была трансформирована в проблему допустимости, где проверяется наличие семантического отношения между концепциями двух узлов, предполагая в качестве контекста всё, что мы смогли вывести относительно отношений между концепциями меток в двух графах.

Процесс создания атомарных концепций происходит следующим образом. Токены, представляющие предлоги, знаки препинания, союзы и т. д., переводятся в логические связки и используются для построения сложных концепций на основе атомарных концепций из первого шага. Например, запятые и союзы переводятся в дизъюнкции, предлоги типа «of», «in» – в конъюнкции. После первой фазы все метки были переведены в предложения внутреннего языка концепций. Далее предполагается, что концепция, обозначенная меткой, является самой меткой.

Шаг 3. Для всех пар атрибутов в двух деревьях вычислить отношения между соответствующими концепциями атрибутов.

Шаг 4. Для всех пар сущностей в двух деревьях вычислить сильнейшие отношения между соответствующими концепциями сущностей.

Таким образом, 3 и 4 производят уточнение предикативных связок, полученных на первых двух шагах.

Данный алгоритм, согласно разработанной классификации, можно отнести к графовому подходу с применением структурного анализа и семантического анализа элементов. Основным недостатком описанного выше подхода является сложность перевода конструкций языка в логические связки, поскольку не все отношения между словами можно точно выразить с использованием таких связок. Также дополнительным ограничением может служить необходимость проверки семантических связей по словарю «WordNet», в нем заключены данные только лишь по английскому языку, однако зачастую термины описания схем данных могут быть представлены в формате транслитерации буквенных сочетаний.

Рассмотрим методику сопоставления «Harmony» [17]. «Harmony» – это методика, работающая на уровне элементов схемы на основе языковых характеристик. Общий алгоритм сопоставления состоит из нескольких шагов.

Первый шаг – нормализация текста. Алгоритм токенизирует все текстовые строки в исходных и целевых схемах, разделяя фразы, которые не разделены пробелами, на отдельные слова, образуя так называемые токены. В процессе токенизации удаляются все знаки препинания и текст содержит только буквы, цифры и пробелы. Дополнительно все заглавные буквы заменяются строчными буквами. Во время предварительной обработки алгоритм также подсчитывает частоту каждого нормализованного слова, появляющегося где-либо в элементе исходной или целевой схемы:

$$\text{freq}(wd) \rightarrow N, \quad (6)$$

где wd – слово, а v в контексте задачи сопоставления схем, атрибут или сущность;

N – количество появления слова в тексте.

Далее по формуле (7) определяется вес, связанный с каждым словом:

$$wt = \frac{1}{\text{freq}(wd)}, \quad (7)$$

где wt – вес слова.

Вес, связанный с каждым словом, обратно пропорционален количеству его появлений в исходных и целевых схемах. В идеальном случае слово появляется ровно один раз в исходной и один раз в целевой схеме или дважды в общей сложности.

Второй шаг – лингвистический поиск. Основной стратегией поиска является определение того, насколько много общих слов есть у пары представленных схем.

Третий шаг – оценка сопоставления. Оценка сопоставления создается при рассмотрении некоторого набора доказательств (общих наблюдаемых доказательств и положительных наблюдаемых доказательств). Оценка сопоставления лежит в пределах $[-1; 1]$. Третий шаг тесно связан со вторым. Положительные доказательства основываются на пересечении соответствующих рассматриваемых токенов, а общие доказательства – на объединении токенов.

Четвертый шаг – объединение оценок. В Harmony запускается несколько процессов сопоставления параллельно, каждый из которых генерирует оценку сопоставления для каждой пары исходных и целевых элементов. Основной алгоритм объединения голосов – это взвешенное среднее оценок сопоставления, сгенерированных каждым из процессов.

Данный алгоритм, согласно разработанной классификации, можно отнести к графовому подходу с применением синтаксического структурного анализа и комбинации семантического, синтаксического и атрибутивного анализа элементов. Основным недостатком данного подхода является сложность процесса токенизации. Не всегда стандартизированные подходы могут позволить графотно осуществить разделение фраз на токены.

«Match Execution» – ядро сопоставления, разработанное в Лейпцигском университете [18]. Процесс сопоставления состоит из нескольких этапов.

Первый этап – этап сопоставления на основании различных типов сопоставителей.

Аффиксы. Этот сопоставитель ищет общие префиксы и суффиксы.

N-граммы. Строки сравниваются на основе их набора n-грамм, то есть последовательностей из n символов.

Расстояние редактирования. Схожесть строк вычисляется на основе количества операций редактирования, необходимых для преобразования одной строки в другую (метрика Левенштейна [19]).

«Soundex». Этот сопоставитель вычисляет фонетическую схожесть между атрибутами.

Синонимы. Этот сопоставитель оценивает схожесть между наименованиями атрибутов на основании словаря («WordNet»).

«DataTуре». Этот сопоставитель использует таблицу синонимов, определяющую степень совместимости между набором predetermined типов данных.

Второй этап – операция «MathCompose», производящая умножение сходств индивидуальных значений между собой. «MathCompose» используется для создания нового результата сопоставления на основе двух существующих результатов, предполагая транзитивность сходства элементов. Сходство элементов, то есть «match» рассчитывается по следующей формуле:

$$match = MatchCompose(match1, match2), \quad (8)$$

где match1 и match2 – предыдущие результаты сопоставления.

Операция предполагает транзитивность отношения сходства между элементами, то есть если PO1 похож на PO2 и PO2 на PO3, то PO1 (весьма вероятно) также похож на PO3, транзитивность элементов рассматривается согласно формуле (9):

$$match_1: PO1 \leftrightarrow PO2, match_2: PO2 \leftrightarrow PO3, \quad (9)$$

где PO1, PO2, PO3, PO4 – различные элементы схем.

Третий этап – естественное объединение между двумя таблицами ввода. В этих таблицах каждый кортеж указывает соответствие 1:1 между элементами соответствующих схем вместе с их значениями сходства. То есть производится объединение двух таблиц по всем общим столбцам с одинаковыми именами.

Основным недостатком «Match Execution» является высокая чувствительность к качеству входных данных. Результаты имеют большую зависимость от качества подготовки схем для сопоставления. Данный алгоритм, согласно разработанной классификации, можно отнести к графовому подходу с применением структурного анализа и комбинации семантического, синтаксического и атрибутивного анализа элементов.

«Cupid» [20] – гибридная методика сопоставления, которая включает в себя лингвистические и структурные техники. «Cupid» вычисляет коэффициенты сходства атрибутов, применяя для сопоставления специализированные для конкретной области словари.

Методика сопоставления состоит из трех этапов и работает только с древовидными структурами, к которым сводятся недревовидные случаи.

Первый этап – лингвистическое сопоставление, вычисляет лингвистические коэффициенты сходства между именами элементов схемы (метками) на основе морфологической нормализации, категоризации, техник на основе строк (тесты общего префикса, суффикса) и поиска в тезаурусе.

Второй этап – структурное сопоставление. Алгоритм вычисляет структурные коэффициенты сходства с весами листьев, измеряющими сходство между контекстами, в которых встречаются элементарные элементы схемы. Структурная схожесть выделяется для поддеревьев, если существует хотя бы одна крепкая связь, то есть связь, имеющая вес выше установленного порога, который определяется по формуле (10):

$$ssim_{s,t} = \frac{|\{x \mid x \in leaves(s) \wedge \exists y \in leaves(t), stronglink(x,y)\} \cup \{x \mid x \in leaves(t) \wedge \exists y \in leaves(s), stronglink(x,y)\}|}{|leaves(s) \cup leaves(t)|}, \quad (10)$$

где $leaves(s)$ и $leaves(t)$ – наборы листьев в поддереве, имеющем корень s и t соответственно.

Третий этап – генерация элементов сопоставления. Алгоритм вычисляет взвешенные коэффициенты сходства и создает окончательное соответствие, выбирая пары элементов схемы с взвешенными коэффициентами сходства, превышающими порог.

Основным недостатком данного алгоритма является необходимость решать задачу определения порога сходства на третьем этапе. Неправильно выбранный порог может привести к недостаточной или избыточной генерации соответствий.

Данный алгоритм, согласно разработанной классификации, можно отнести к графовому подходу с применением структурного анализа и комбинации семантического и атрибутивного анализа элементов.

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ И ТЕСТИРОВАНИЕ МЕТОДИК

Теперь, когда были составлены описания наиболее популярных алгоритмов, использующие графовый подход в решении проблем сопоставления, произведем сравнительный анализ ряда существующих программных продуктов, имеющих в открытом доступе, которые разработаны на основе описанных выше методик. Общая информация о каждой из рассматриваемых программ представлена в таблице 1. В данной сравнительной таблице указан ряд параметров каждого программного продукта: методика сопоставления данных, тип входной информации, тип интерфейса, язык интерфейса, визуализация выходной информации.

Таблица 1 – Сравнительная таблица программных продуктов сопоставления схем данных

ПО /параметры	Методика	Тип входной информации	Тип интерфейса	Язык интерфейса	Визуализация отчетов
«Rondo»	«Similarity Flooding»	DDL, XML	GUI+BAT	Английский	Визуализированные связи между сущностями
«S-MATCH»	«Semantic match»	XML	GUI+BAT	Английский	Набор продукционных правил
«OpenII»	«Harmony»	DDL, XML	GUI	Английский	Визуализированные связи между сущностями
«Coma-3.0»	«Match Execution»	XML	GUI	Английский	Визуализированные связи между сущностями
«Reproducing-schema-matching»	«Cupid»	DDL	Source code	Английский	Отчет с результатами сопоставления

Также стоит отметить, что, помимо программных решений, существуют коммерческие системы интеграции баз данных, подвергнутые анализу в [21]. Однако эти системы не пригодны для решения задач, рассматриваемых в данном исследовании. Они предназначены не для аналитических задач, а скорее для ряда задач системного администрирования. Разница в подходах к интеграции баз данных для аналитики и системного администрирования заключается в том, что в первом случае основное внимание уделяется обеспечению эффективного анализа данных с целью извлечения ценной информации, во время как во втором случае акцент делается на обеспечении стабильной работы системы и управлении её инфраструктурой.

Часть анализируемых программных продуктов работает со входными данными в формате XML, что при анализе сущностей баз данных, которые на практике зачастую представляются в форматах, совместимых со стандартами ANSI SQL, создает необходимость преобразований входной информации для приведения ее к поддерживаемой программами структуре. Авторы «Rondo» и «S-MATCH» предусмотрели автоматический конвертер DDL-скрипта с описанием структуры сущностей в XML-формат. Это тоже может быть не удобно, так как в случае, если код написан для какой-

либо из промышленных СУБД («Oracle», «PostgreSQL», «MSSQL» и т.д.), его также необходимо предварительно привести к общепринятому формату ANSI SQL. Разработчики «OpenII» и «Coma-3.0» принимают код DDL любого формата вне зависимости от стандарта написания. Для тестирования алгоритма «Cupid» в представленной программной реализации необходимо преобразовать DDL-файл в формат «csv»-файла со структурой «таблица-атрибут».

Во всех программных продуктах присутствует GUI-интерфейс для визуализации связей между сущностями, в «Rondo» и «S-MATCH» есть возможность работать непосредственно с исполняемыми файлами без GUI-интерфейса. Исключением является алгоритм Cupid – для него отсутствует готовый программный продукт. Для тестирования была использована реализация алгоритма из открытого репозитория на «Github» [22]. Все рассматриваемые программные продукты имеют англоязычный интерфейс.

Результаты работы программ «Rondo», «OpenII», «Coma-3.0» представляются в формате прорисованных связей между сущностями, с описанием рассчитанной вероятности схожести тех или иных атрибутов. Все программные продукты позволяют пользователю сохранять полученные результаты. «S-MATCH» в результате работы выдает перечень производственных правил, которые указывают на вероятность соотношений тех или иных атрибутов между собой. «Reproducing-schema-matching» рассчитывает статистику и выводит полученные результаты в текстовый файл.

Для того чтобы проанализировать работу данных программных продуктов в совокупности с работой применяемых в них методик, были разработаны сценарии тестирования, которые основывались на реально существующем хранилище данных, с измененными для сохранения конфиденциальности сущностями и атрибутами.

Хранилище данных организовано по принципу Data Vault [23]. При данном подходе все сущности делятся на три класса: хабы, линки, сателлиты.

Хабы содержат ограниченное число атрибутов и позволяют в общих чертах обозначить состояние экземпляра рассматриваемой сущности.

Сателлиты представляют собой подробное описание экземпляра сущности, содержа в себе ключевые записи показателей.

Линки обозначают связи между хабами и сателлитами и по своей сути формируют в хранилище так называемые отношения «много ко многим», то есть отношения между двумя сущностями, где одна сущность может соответствовать нескольким экземплярам другой, и наоборот.

Такая архитектура хранилища обладает существенным преимуществом, предоставляя легкость в последующем масштабировании, например, при создании сателлитов из новых систем-источников для хабов. Необходимо также выделить относительную простоту разработки ETL-процессов (процессов извлечения данных, преобразования данных и загрузки в целевую базу данных) в соответствии с данной архитектурой. Но при такой архитектуре достаточно сложно осуществлять процесс анализа сущностей и приведения новых сущностей к общей структуре.

В целом сценарии охватывают три вида потенциальных связей между сущностями:

1. «Много ко многим» – тип связей, описанный выше.

2. «Один ко многим» – это тип связи между двумя сущностями в базе данных, где одна запись в одной таблице (родительская таблица) может иметь несколько соответствующих записей в другой таблице (дочерней таблице), но каждая запись в дочерней таблице может иметь только одну связанную запись в родительской таблице.

3. «Один ко одному» – это тип связи между двумя сущностями в базе данных, при котором одна запись в одной таблице соответствует не более чем одной записи в другой таблице и наоборот. В этом случае каждая запись в каждой из связанных таблиц имеет уникальное отношение к другой записи.

Для проверки эффективности выбранных методик были разработаны сценарии, охватывающие различные варианты взаимосвязей между сущностями в реляционных базах данных. Таблица со сценариями представлена ниже:

Таблица 2 – Сравнительная таблица программных продуктов сопоставления схем данных

Сценарий / Схема	Схема 1	Схема 2
Много ко многим	Таблица «публичные отчеты параметры», Таблица «публичные отчеты данные»	Таблица «хаб_отчеты», Таблица «линк_отчеты», Таблица «сателлиты отчеты»
Один ко многим	Таблица «сотрудники»	Таблицы хабы («сотрудники», «телефоны», «электронные почты», «отделы», «должности») Таблицы линки («сотрудники_телефоны», «сотрудники_почты», «должности_отделы») Таблица «сателлиты_сотрудники»
Один к одному	Таблица «продукция»	Таблица «продукция»

В таблице 3 рассмотрим пример сущностей баз данных из третьего сценария, результаты тестирования которого отображены далее на рисунках 2–10.

Таблица 3 – Сущности источника и приемника в сценарии № 3

Таблица источника «production»	Таблица приемника «ods_report_prd»
id SERIAL PRIMARY KEY	id serial NOT NULL PRIMARY KEY
datetime TIMESTAMP NOT NULL	dt_of_report varchar(32) NULL
service_id INT NOT NULL	datetime TIMESTAMP NOT NULL
date TIMESTAMP	service_id int4 NOT NULL
prod VARCHAR(10)	structure_path1 varchar(100) NOT NULL
plan_item_val NUMERIC(100, 3)	structure_path2 varchar(100) NULL
plan_item_add_val NUMERIC(100, 3)	structure_path3 varchar(100) NULL
plan_manufacture_val NUMERIC(100, 3)	head_name varchar(100) NOT NULL
plan_manufacture_add_val NUMERIC(100, 3)	dimension varchar(16) NULL
plan_artifact_val NUMERIC(100, 3)	equation varchar(200) NULL
plan_artifact_add_val NUMERIC(100, 3)	plan_value varchar(100) NULL
plan_artifact_obj_val NUMERIC(100, 3)	plan_date varchar(100) NULL
plan_artifact_obj_add_val NUMERIC(100, 3)	addition_value varchar(100) NULL
plan_commodity_val NUMERIC(100, 3)	dscrptn varchar(1000) NULL
plan_commodity_add_val NUMERIC(100, 3)	done_time TIMESTAMP NOT NULL
done_time TIMESTAMP NOT NULL	done_state INT DEFAULT 0 NOT NULL
done_state INT DEFAULT 0 NOT NULL	

Часть связей этого сценария определяется абсолютно однозначно. К примеру, атрибуты с наименованиями «id», «datetime», «done_state». Однако присутствуют атрибуты, определение связей между которыми могут составить сложность для программных продуктов, например «plan_item_val», «plan_artifact_val», «plan_artifact_obj_val», «plan_commodity_val» и «plan_value».

Поочередно проведем тестирование каждого из представленного в таблице 1 программного продукта на основании сценариев из таблицы 2.

Интерфейс программного продукта «Rondo» представлен на рисунках 2 и 3. Для демонстрации был выбран пример третьего сценария. В результате работы программы были выделены 7 связей между атрибутами таблиц, которые посчитались алгоритмом как однозначное совпадение (рис. 2).

При нажатии на каждую из связей отображается вероятность соотношения атрибутов (рис. 3). Для каждого атрибута одной сущности подбираются три наиболее похожих на него из другой сущности.

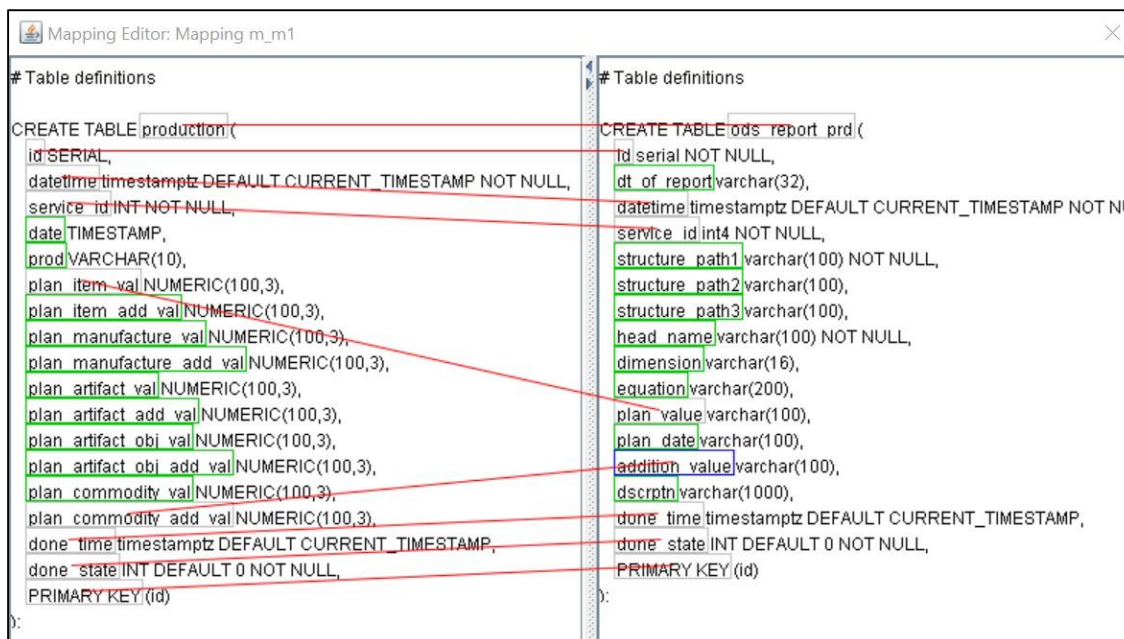


Рисунок 2 – Тестирование третьего сценария в ПО «Rondo»

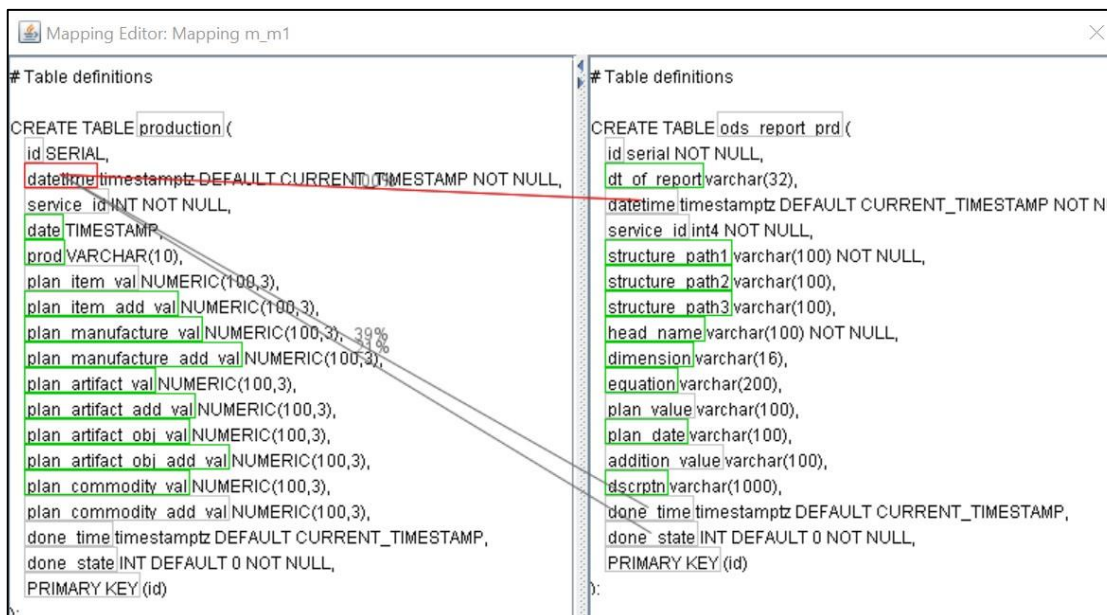


Рисунок 3 – Отображения вероятностей сходства в ПО «Rondo»

Интерфейс программного продукта «S-MATCH» представлен на рисунке 4. Для третьего сценария программа определила 2 верных связи из 17. Было выявлено, что программное решение и методика, лежащая в его основе, подходит для сравнения онтологий, позволяя проводить детальный анализ структуры данных и обеспечивая сопоставление между различными онтологическими моделями. При этом «S-MATCH» показывает низкий процент эффективности при решении задач сопоставления схем данных.

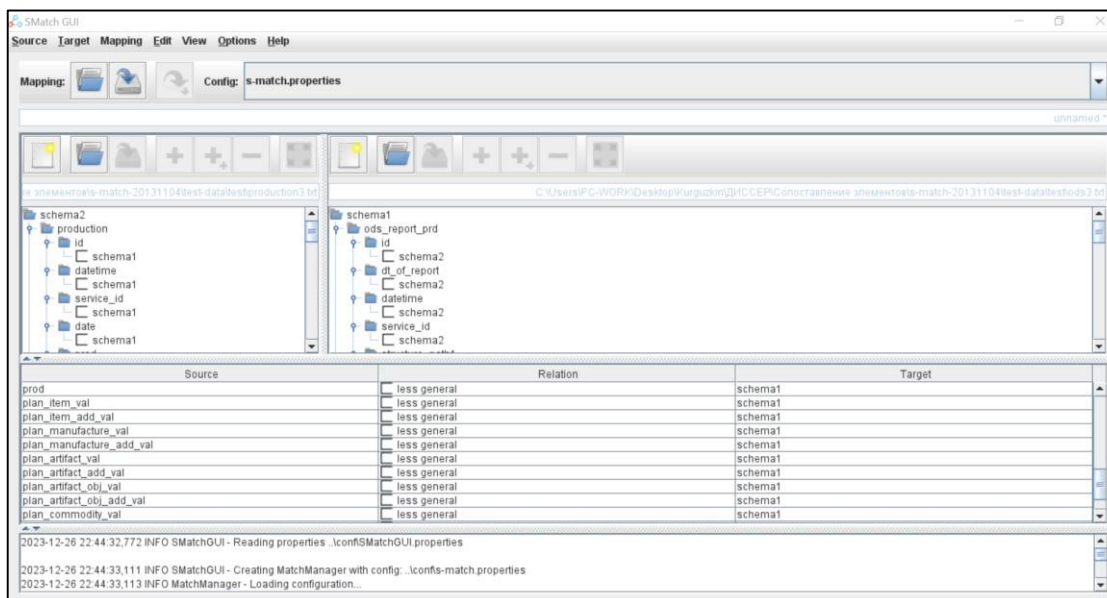


Рисунок 4 – Тестирование третьего сценария в ПО «S-MATCH»

Интерфейс программного продукта «OpenII» [24] представлен на рисунке 5. «OpenII» определил возможные связи для 16 из 17 источников, представленного в скрипте третьего сценария.

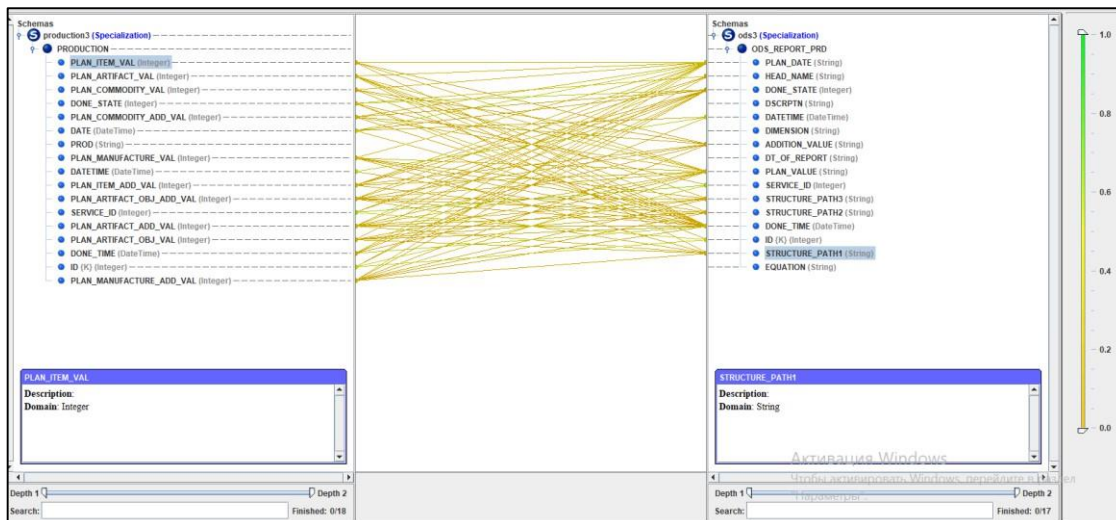


Рисунок 5 – Тестирование третьего сценария в ПО «OpenII»

При нажатии на атрибут сущности показываются возможные связи этого атрибута. В колонке, расположенной на правом краю панели программы, отображается вероятность соотношения атрибутов (рис. 6). Количество отображаемых связей не ограничено. Процесс сравнения атрибутов настраивается в параметрах. Имеется возможность выбрать различные методы сопоставления – схожесть имен, схожесть структуры и т. д. Для получения более точных результатов были выбраны все доступные методы сравнения.

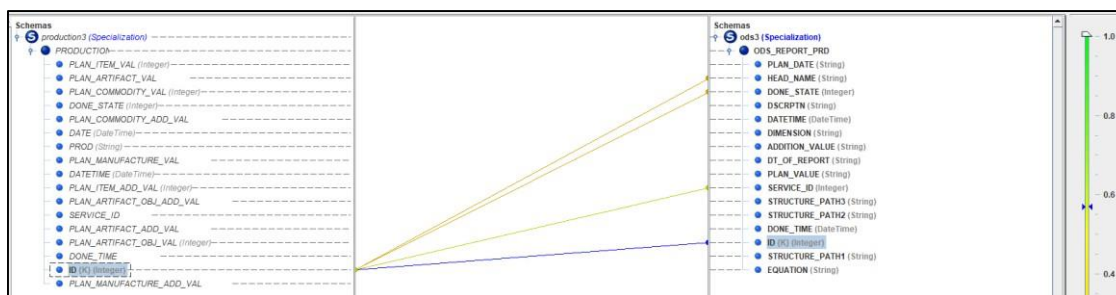


Рисунок 6 – Отображения вероятностей сходства в ПО «OpenII»

Интерфейс программного продукта «Сота-3.0» представлен на рисунке 7. Алгоритм определил 6 связей для атрибутов из третьего сценария. Стоит отметить, что все определенные связи верны. Также очевидно, что «Сота-3.0» не стремится выделить взаимосвязи для каждого атрибута из источника, если нет определенной уверенности в правильности выявленного отношения.

При нажатии на атрибут источника программа отображает вероятность связи этого атрибута с конкретным атрибутом в приемнике (рис. 8). Для одного атрибута источника – один атрибут приемника. По аналогии с «OpenII» процесс сравнения атрибутов настраивается в параметрах. Наиболее точные результаты выявлены при методе сравнения «наименования узлов».

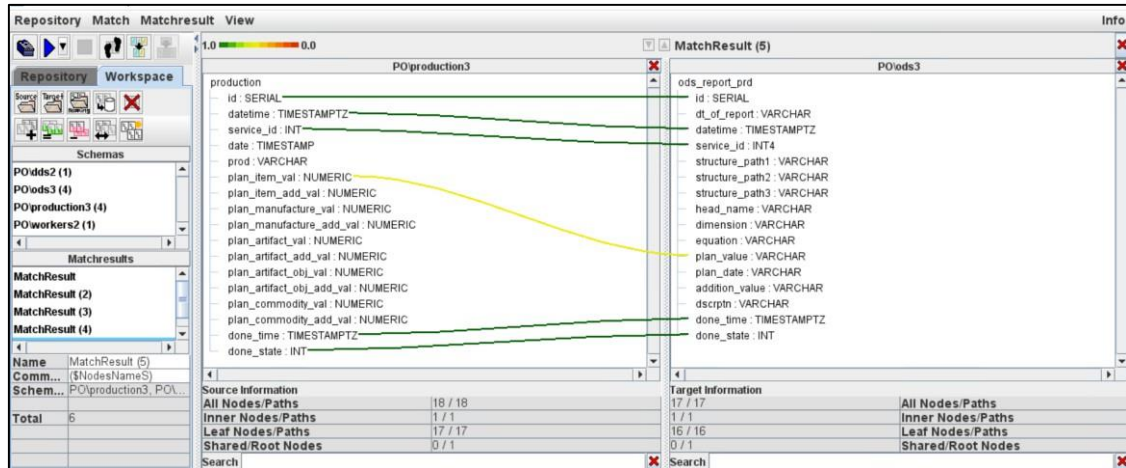


Рисунок 7 – Тестирование третьего сценария в ПО «Coma-3.0»

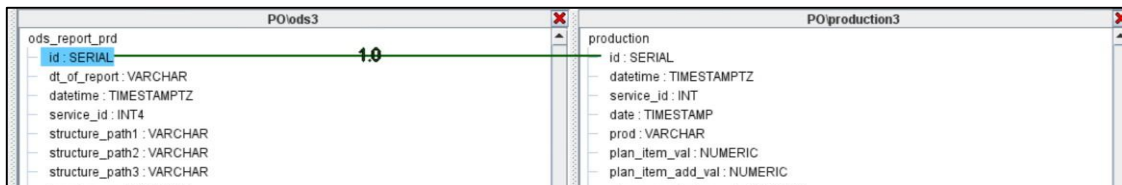


Рисунок 8 – Отображения вероятностей сходства в ПО «Coma-3.0»

Алгоритм для методики «Cupid» тестировался редактированием параметров исходного кода напрямую (рис. 9). Результаты работы алгоритма для третьего сценария отображены на рисунке 10. Было найдено 2 взаимосвязи между сущностями. Отчеты по умолчанию сохраняются в текстовый файл. Вероятности отношений между атрибутами не выводятся.

Следует отметить, что данный продукт показал худшие результаты по соотношению качество результатов / время работы. При тестировании третьего сценария время работы алгоритма составило 10 минут, остальные алгоритмы справлялись с данным сценарием менее чем за минуту.

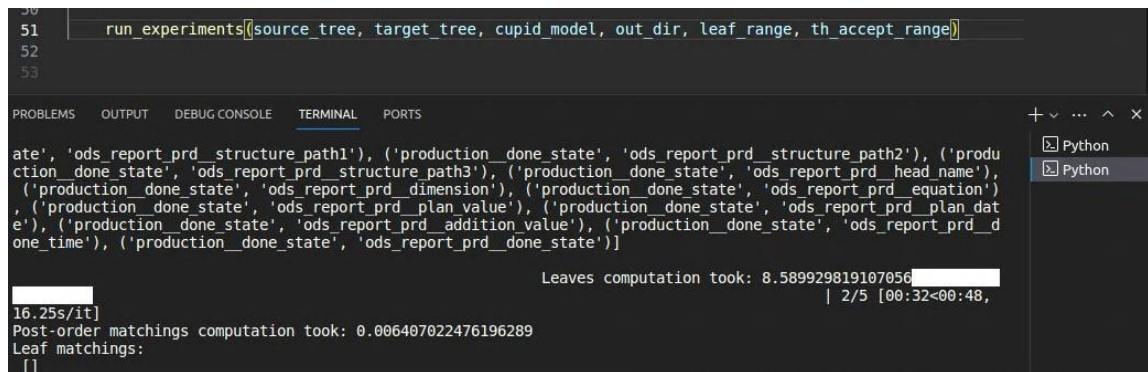


Рисунок 9 – Тестирование третьего сценария в ПО «Reproducing-schema-matching»



Рисунок 10 – Отображения вероятностей сходства в ПО «Reproducing-schema-matching»

АНАЛИЗ РЕЗУЛЬТАТОВ

Результаты проведенного анализа программных средств, осуществляющих сопоставления схем данных, представлены в таблице 4. Программные продукты «Rondo», «OpenII» и «Coma-3.0» в целом показали сопоставимые результаты по третьему сценарию, верно определив 6–7 связей между атрибутами. Наилучшие результаты по второму сценарию показал программный продукт «Coma-3.0», верно было определено 16 взаимосвязей. По первому сценарию лучший результат показал программный продукт «OpenII» – верно обнаружил 12 связей.

Таблица 4 – Сравнительная таблица программных продуктов сопоставления схем данных

ПО / Сценарии	«Много ко многим»	«Один ко многим»	«Один к одному»
«Rondo»	5 – верно, 10 – неоднозначно, 5 – неверно	8 – верно, 28 – неоднозначно, 4 – неверно	7 – верно, 8 – неоднозначно, 2 – неверно
«S-MATCH»	2 – верно, 18 – неоднозначно	4 – верно, 36 – неоднозначно	2 – верно, 15 – неоднозначно
«OpenII»	12 – верно, 7 – неоднозначно, 1 – неверно	10 – верно, 23 – неоднозначно, 7 – неверно	6 – верно, 9 – неоднозначно, 2 – неверно
«Coma-3.0»	6 – верно, 14 – неоднозначно	16 – верно, 24 – неоднозначно	6 – верно, 15 – неоднозначно
««Reproducing-schema-matching»»	4 – верно, 4 – неоднозначно, 12 – неверно	5 – верно, 35 – неоднозначно	2 – верно, 15 – неоднозначно

Стоит отметить, что у всех программных средств большой процент неоднозначно выделенных связей, когда алгоритм не смог определить точное сопоставление для того или иного атрибута. Как уже отмечалось ранее, программный продукт «S-MATCH» показал неоднозначные результаты и, очевидно, не подходит для рассматриваемой задачи сравнения сущностей схем данных.

На рисунке 11 отображены сводные диаграммы по каждому из рассмотренных сценариев. Как видно на диаграммах, только в одном случае процент верно определенных связей превысил 50 процентов. С каждым из сценариев различные программные продукты справились с разной эффективностью. Наиболее эффективными в среднем оказались «OpenII» и «Coma-3.0» – данные программные продукты верно определил 36 процентов связей между атрибутами.

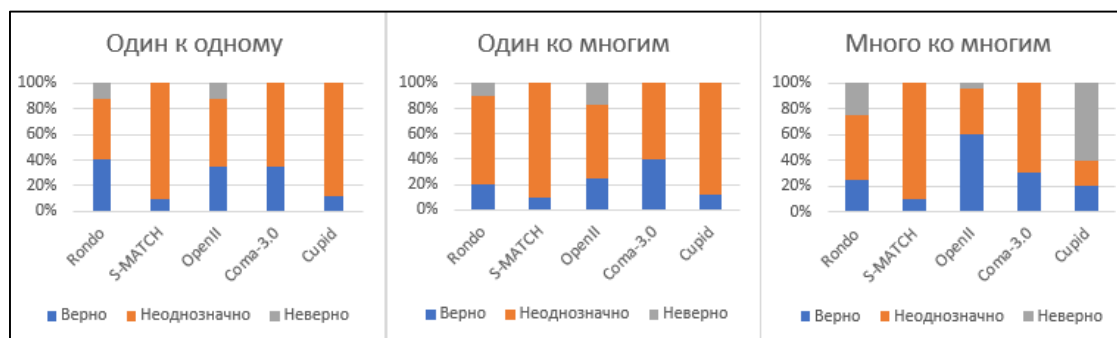


Рисунок 11 – Сравнительная диаграмма результатов сопоставления

Также следует отметить, что с каждым из сценариев программные продукты справились с разной эффективностью. Так, по первому сценарию «Rondo» показал лучшие результаты сопоставления – 40 процентов верно определенных связей. По второму сценарию лучшие результаты у «Coma-3.0» – 40 процентов верных связей. По третьему сценарию – «OpenII» и 60 процентов верно определенных связей.

ЗАКЛЮЧЕНИЕ

В результате проведенного анализа были сделаны следующие выводы. Результативность проанализированных методик сопоставления составила от 15 до 60 процентов, что сигнализирует о необходимости оптимизации существующих методик.

В связи с тем, что рассмотренные существующие решения недостаточно эффективно справляются с поставленными задачами, требуются дальнейшие исследования в области сопоставления

схем данных. Опираясь на результаты практического эксперимента, рассмотренного в данной статье, целесообразно уделить внимание развитию одной из наиболее проявивших себя методик («OpenII», «Coma-3.0», «Rondo»).

Большинство программных продуктов и методик, ориентированных на графовый анализ сущностей с применением семантических или синтаксических элементов анализа атрибутов и ограничений, справляются только с простыми вариантами сопоставлений. Большое число связей выделяется неоднозначно: алгоритмы зачастую либо стремятся показать связи для всех представленных атрибутов, либо не могут определиться, к какому из атрибутов приемника отнести тот или иной атрибут источника данных. Для большинства современных схем данных, в том числе для той, что была представлена в сценариях сравнения в настоящем исследовании, очень сложно, не зная контекста непосредственных данных, которые будут помещены в каждый из атрибутов схемы, однозначно определить взаимосвязи схем друг с другом.

Таким образом, существующие классические методики сопоставления схем данных необходимо модернизировать и оптимизировать при помощи методов анализа контекста, то есть непосредственного анализа самих данных, которые должны быть размещены в анализируемых схемах, а также дать возможность аналитику вносить производные правила, позволяющие вручную определять ряд взаимосвязей для получения более точных результатов сравнения. Для решения данных задач подходят современные методы работы с данными, основанные на машинном обучении. Соответственно, в дальнейшем планируется проанализировать научные исследования и разработки, касающиеся применения механизмов машинного обучения в области интеграции, реструктуризации и управления данных.

Список источников

1. Big Data Analytics. Fortune Business Inside. 2023. – URL: <https://www.fortunebusinessinsights.com/big-data-analytics-market-106179>.
2. Data integration in 2023. Actioner. 2023. – URL: <https://actioner.com/guides/data-integration-statistics>.
3. Хайдарова, С. Создание SQL-запросов в реляционных базах данных / С. Хайдарова // Вестник РГТУ. Серия: информатика. Информационная безопасность. Математика. – 2020. – С. 8–19.
4. Ontology Alignment Evaluation Initiative. – URL: <https://oei.ontologymatching.org/>.
5. Semantic Web Challenges. – URL: <https://iswc2023.semanticweb.org/semantic-web-challenges/>.
6. Batini, C. A Comparative Analysis of Methodologies for Database Schema Integration / C. Batini, M. Lenzerini, S. Navathe // ACM Computing Surveys. – 1986. – 18 (4). – P. 323–364.
7. Shvaiko, P. A Survey of Schema-Based Matching Approaches / P. Shvaiko, J. Euzenat // Data Semantics (JoDS). – 2005. – Vol. IV. – P. 1–29.
8. Sutanta, E. Survey: Models and Prototypes of Schema Matching / E. Sutanta, R. Wardoyo, K. Mustofa, E. Winarko // International Journal of Electrical and Computer Engineering (IJECE). – 2016. № 6 (3). – P. 1011–1022.
9. Alwan, A. A Survey of Schema Matching Research using Database Schemas and Instances / A. Alwan, A. Nordin, M. Alzeber, A. Abualkishik // International Journal of Advanced Computer Science and Applications. – 2017. – Vol. 8 (10). – P. 102–111.
10. Martono G., Azhari S. Review implementation of linguistic approach in schema Matching / G. Martono, S. Azhari // International Journal of Advances in Intelligent Informatics. – 2017. – Vol. 3, № 1. – P. 1–9.
11. Fernandez, R. Seeping Semantics: Linking Datasets using Word Embeddings for Data Discovery / R. Fernandez, E. Mansour, A. Ali Qahtan, A. Elmagarmid, I. Ilyas, S. Madden, M. Ouzzani, M. Stonebraker, N. Tang // IEEE 34th International Conference on Data Engineering (ICDE). – 2018. – P. 2–12.
12. Sahay, T. Schema Matching using Machine Learning / T. Sahay, A. Mehta, S. Jadon ; College of Information and Computer Sciences University of Massachusetts. – Amherst, 2019. – P. 1–7.
13. Sun, Z. What Makes Entities Similar? A Similarity Flooding Perspective for Multi-sourced Knowledge Graph Embeddings / Z. Sun, J. Huang, X. Xu, Q. Chen, W. Ren, W. Hu // ICML'23: Proceedings of the 40th International Conference on Machine Learning. – 2023. – P. 1–11.
14. Шелманов, А. О. Исследование методов автоматического анализа текстов и разработка интегрированной системы семантико-синтаксического анализа : дис. / А. О. Шелманов. – Москва, 2015. – С. 1–210.
15. Melnik, S. Similarity flooding: A versatile graph matching algorithm and its application to schema matching / S. Melnik, H. Garcia-Molina, E. Rahm // Data Engineering : 18th International Conference. – 2002. – P. 1–12.
16. Giunchiglia, F. S-match: an algorithm and an implementation of semantic matching / F. Giunchiglia, P. Shvaiko, M. Yatskevich // European Semantic Web Symposium. – Springer, 2004. – P. 61–75.
17. Mork, P. The Harmony Integration Workbench / P. Mork, L. Seligman, A. Rosenthal, J. Korb, C. Wolf. – The MITRE Corporation McLean, VA, USA, 2008. – P. 3–31.
18. Do, H. COMA – A system for flexible combination of schema matching approaches / H. Do, E. Rahm // Proceedings of the 28th VLDB Conference. Hong Kong, China, 2002. – P. 1–12.
19. Hall, P. Levenshtein: Approximate String Matching / P. Hall, G. Dowling // Computing Survey. – 1980. – Vol. 12, № 4. – P. 381–402.
20. Madhavan, J. Generic Schema Matching with Cupid / J. Madhavan, P. Bernstein, E. Rahm // Proceedings of the 27th VLDB Conference. – Roma, Italy, 2001. – P. 1–10.
21. Дэвис, Р. Инструменты сравнения схемы баз данных // Открытые системы. – 2012. – URL: <https://www.osp.ru/winitpro/2012/17/13017909>.

22. Repository of Andra Denis Ionescu. Reproducing-schema-matching. – URL: <https://github.com/AndraIonescu/reproducing-schema-matching/>.
23. Попов, М. И. Data vault в качестве архитектуры хранилища данных банковской организации / М. И. Попов, Ф. Б. Лазарев // Auditorium. – 2022. – С. 1–5.
24. Seligman, L. OpenII: An Open Source Information Integration Toolkit / L. Seligman, P. Mork, A. Halevy, K. Smith // Proceedings of the ACM SIGMOD International Conference on Management of Data. – Indianapolis, Indiana, USA, 2010. – P. 1–3.

References

1. *Big Data Analytics. Fortune Business Inside*. 2023. URL: <https://www.fortunebusinessinsights.com/big-data-analytics-market-106179>.
2. Data integration in 2023. Actioner. 2023. URL: <https://actioner.com/guides/data-integration-statistics>.
3. Khaydarova S. Creating SQL queries in relational databases. *Bulletin of RHTU. Series: Informatics. Information Security. Mathematics*, 2020, pp. 8–19. (In Russ.).
4. *Ontology Alignment Evaluation Initiative*. URL: <https://oaei.ontologymatching.org/>.
5. *Semantic Web Challenges*. URL: <https://iswc2023.semanticweb.org/semantic-web-challenges/>.
6. Batini, C., Lenzerini, M., Navathe, S. A Comparative Analysis of Methodologies for Database Schema Integration. *ACM Computing Surveys*, 1986, no. 18 (4), pp. 323–364.
7. Shvaiko, P., Euzenat, J. A Survey of Schema-Based Matching Approaches. *Data Semantics (JoDS)*, 2005, vol. IV, pp. 1–29.
8. Sutanta, E., Wardoyo, R., Mustofa, K., Winarko, E. Survey: Models and Prototypes of Schema Matching. *International Journal of Electrical and Computer Engineering (IJECE)*, 2016, vol. 6 (3), pp. 1011–1022.
9. Alwan, A., Nordin, A., Alzeber, M., Abualkishik, A. A Survey of Schema Matching Research using Database Schemas and Instances. *International Journal of Advanced Computer Science and Applications*, 2017, vol. 8 (10), pp. 102–111.
10. Martono, G., Azhari, S. Review implementation of linguistic approach in schema Matching. *International Journal of Advances in Intelligent Informatics*, 2017, vol. 3, no. 1, pp. 1–9.
11. Fernandez, R., Mansour, E., Ali Qahtan, A., Elmagarmid, A., Ilyas, I., Madden, S., Ouzzani, M., Stonebraker M., Tang N. Seeing Semantics: Linking Datasets using Word Embeddings for Data Discovery. *IEEE 34th International Conference on Data Engineering (ICDE)*, 2018, pp. 2–12.
12. Sahay, T., Mehta, A., Jadon, S. Schema Matching using Machine Learning. College of Information and Computer Sciences University of Massachusetts, Amherst, 2019, pp. 1–7.
13. Sun, Z., Huang, J., Xu, X., Chen, Q., Ren, W., Hu, W. What Makes Entities Similar? A Similarity Flooding Perspective for Multi-sourced Knowledge Graph Embeddings. *ICML'23: Proceedings of the 40th International Conference on Machine Learning*, 2023, pp. 1–11.
14. Shelmanov, A. O. Research of methods of automatic analysis of texts and development of an integrated system of semantic-syntactic analysis : dissertation. Moscow, 2015, pp. 1–210. (In Russ.).
15. Melnik, S., Garcia-Molina, H., Rahm, E. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. *Data Engineering : 18th International Conference*, 2002, pp. 1–12.
16. Giunchiglia, F., Shvaiko, P., Yatskevich, M. S-match: an algorithm and an implementation of semantic matching. *European Semantic Web Symposium*. Springer, 2004, pp. 61–75.
17. Mork, P., Seligman, L., Rosenthal, A., Korb, J., Wolf, C. *The Harmony Integration Workbench*. The MITRE Corporation McLean, VA, USA, 2008. P. 3–31.
18. Do, H., Rahm, E. COMA – A system for flexible combination of schema matching approaches. *Proceedings of the 28th VLDB Conference*. Hong Kong, China, 2002, pp. 1–12.
19. Hall, P., Dowling, G. Levenshtein: Approximate String Matching. *Computing Survey*, 1980, vol. 12, no. 4, pp. 381–402.
20. Madhavan, J., Bernstein, P., Rahm, E. Generic Schema Matching with Cupid. *Proceedings of the 27th VLDB Conference*. Roma, Italy, 2001, pp. 1–10.
21. Davis, R. Tools for comparing database schema. *Open Systems*, 2012. URL: <https://www.osp.ru/winit-pro/2012/17/13017909>. (In Russ.).
22. *Repository of Andra Denis Ionescu. Reproducing-schema-matching*. URL: <https://github.com/AndraIonescu/reproducing-schema-matching/>.
23. Popov, M. I., Lazarev, F. B. Data vault as a data storage architecture for a banking organization. *Auditorium*, 2022, pp. 1–5. (In Russ.).
24. Seligman, L., Mork, P., Halevy, A., Smith, K. OpenII: An Open Source Information Integration Toolkit. *Proceedings of the ACM SIGMOD International Conference on Management of Data*. Indianapolis, Indiana, USA, 2010, pp. 1–3.

Статья поступила в редакцию 10.04.2024; одобрена после рецензирования 26.04.2024; принята к публикации 28.04.2024.

The article was submitted 10.04.2024; approved after reviewing 26.04.2024; accepted for publication 28.04.2024.

УДК 004.827

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МОДЕЛЕЙ ТРАНСФОРМЕРА ДЛЯ КЛАССИФИКАЦИИ НЕСТРУКТУРИРОВАННОЙ ТЕКСТОВОЙ ИНФОРМАЦИИ

Нежников Ратибор Игоревич, Астраханский государственный университет имени В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
аспирант, ORCID: 0009-0000-2042-4028, e-mail: nezchnikov1998@gmail.com

Марьенков Александр Николаевич, Астраханский государственный университет имени В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат технических наук, ORCID: 0000-0003-1378-3553, e-mail: marenkovan17@gmail.com

В данной статье представлен обзор подхода к классификации неструктурированной текстовой информации с использованием модели трансформера. Трансформеры, такие как BERT, GPT и RoBERTa, предлагают существенные преимущества в обработке текста и анализе данных благодаря своей мощной архитектуре и возможности к файн-тюнингу. Описываются основные этапы использования трансформеров в задаче классификации текстов, включая предобработку данных, выбор и настройку архитектуры трансформера, а также методы оценки производительности. В статье описан обзор проблемы классификации текстовой информации, основных методов ее решения, а также преимущества использования трансформеров в этой области. Приводится краткий обзор основных компонентов архитектуры трансформера, таких как механизм позиционного кодирования, многослойные перцептроны и механизм внимания. Анализируются методы подготовки данных для классификации, настройки и файн-тюнинга модели трансформера, а также методы оценки производительности классификации с использованием таких показателей, как точность, полнота и F1-мера. Приводятся практические результаты исследования нейросетевых моделей для решения задачи классификации, а также приведен анализ архитектур, размерности модели, регуляризации, оптимизатора, скорости обучения и размера батча, которые могут влиять на эффективность использования трансформеров для классификации текстов.

Ключевые слова: модель трансформера, BERT, RoBERTa, классификации текстовой информации, нейронные сети

COMPARATIVE ANALYSIS OF TRANSFORMER MODELS FOR CLASSIFICATION OF UNSTRUCTURED TEXT INFORMATION

Nezhnikov Ratibor I., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

graduate student, ORCID: 0009-0000-2042-4028, e-mail: nezchnikov1998@gmail.com

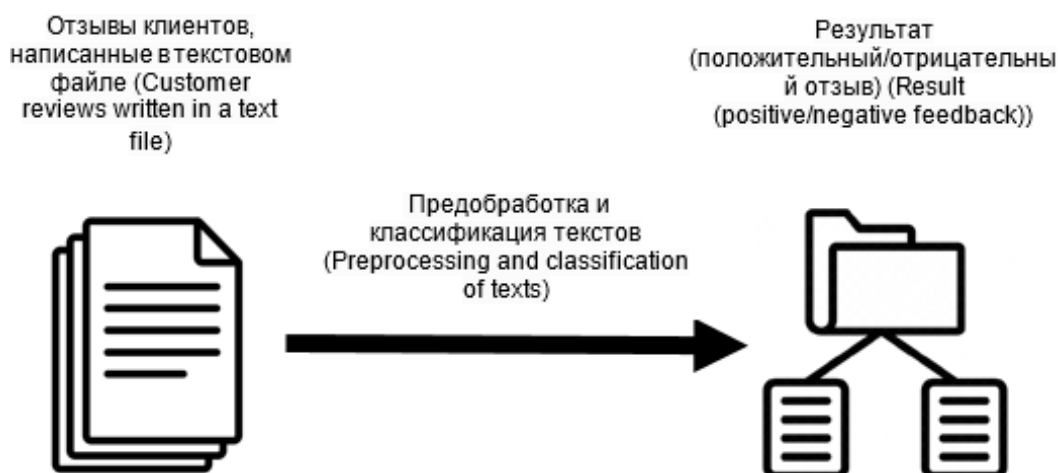
Marenkov Alexander N., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

Cand. Sci. (Engineering), Associate Professor, ORCID: 0000-0003-1378-3553, e-mail: marenkovan17@gmail.com

This article presents an overview of an approach to classifying unstructured textual information using the transformer model. Transformers such as BERT, GPT and RoBERTa offer significant advantages in text processing and data analysis due to their powerful architecture and fine-tuning capabilities. The main stages of using transformers in the task of text classification are described, including data preprocessing, selection and configuration of the transformer architecture, as well as methods for evaluating performance. The article describes an overview of the problem of classifying text information, the main methods for solving it, as well as the advantages of using transformers in this area. Transformer Architecture Overview: Provides a brief overview of the major components of the Transformer architecture, including the attention engine, multilayer perceptrons, and positional encoding engine. Text classification using the Transformer model: Methods for preparing data for classification, tuning and fine-tuning the Transformer model are described, as well as approaches to assessing classification performance using metrics such as accuracy, precision, recall and F1-measure. Practical aspects: Discusses key factors that can influence the effectiveness of using transformers for text classification, such as choice of architecture, model dimension, regularization, optimizer, learning rate and batch size.

Keywords: transformer model, BERT, RoBERTa, text information classification, neural networks

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

Обработка и анализ текстовых данных являются одними из наиболее важных задач в современном мире. С каждым днем количество неструктурированной текстовой информации увеличивается благодаря появлению интернета, социальных сетей, блогов и других цифровых платформ. В результате необходимы мощные инструменты для анализа и классификации текстов, которые могут быть применены в широком спектре областей, таких как маркетинг, журналистика, научные исследования, правоохранительные органы и т. д. [1].

За последнее время машинное обучение и искусственный интеллект стали основными инструментами для работы с текстовыми данными. Идея архитектуры трансформера, предложенная Vaswani и его коллегами в 2017 г. [11], является одним из важных прорывов в этой области. В работах [10, 12, 14] при решении задач обработки естественного языка (NLP), таких как машинный перевод, извлечение информации, генерация текста, определение тональности текста и т. д., архитектуры BERT, GPT и RoBERTa имеют низкий процент ошибок.

Целью данной статьи является исследование применения трансформеров для классификации неструктурированной текстовой информации. Для этого необходимо рассмотреть их основные принципы работы, архитектуру, а также способы применения для решения задач классификации. В рамках экспериментальной части необходимо обучить модель на реальном наборе данных и проанализировать полученные результаты с использованием различных метрик производительности. По результатам проведенного исследования можно будет оценить эффективность моделей и определить возможность для дальнейшего применения в задаче классификации текстов.

ТРАНСФОРМЕРЫ В ОБРАБОТКЕ ЕСТЕСТВЕННОГО ЯЗЫКА

Трансформер – это архитектура глубокого обучения. Авторы архитектуры революционизировали обработку естественного языка, и их разработка стала основой для создания множества моделей, таких как BERT, GPT-3, T5 и т. д.

Основными преимуществами трансформеров являются их способность к параллельной обработке данных и механизм внимания (attention), позволяющий модели эффективно обрабатывать информацию из разных частей текста [12].

Механизм внимания вычисляет взвешенную сумму значений (values) на основе сходства между запросами (queries) и ключами (keys). В контексте трансформера запросы, ключи и значения генерируются из векторных представлений слов в тексте. Формула для механизма внимания следующая:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{1}$$

- где Q – матрица запросов;
- K – матрица ключей;
- V – матрица значений;
- d_k – размерность ключей и запросов.

Важным элементом трансформера является механизм самовнимания (self-attention), который вычисляет взаимосвязи между словами в тексте. Самовнимание является частным случаем механизма внимания, где запросы, ключи и значения берутся из одного и того же источника (текущего входа или предыдущего слоя трансформера) [8].

Трансформеры используют многоголовое внимание для улучшения обработки информации с разных позиций и представлений. Многоголовое внимание состоит из нескольких параллельных механизмов внимания, называемых «головами». Результаты каждой головы объединяются и передаются на следующий слой. Формула для многоголового внимания:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O, \quad (2)$$

где $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$;
 W_i^Q, W_i^K, W_i^V – весовые матрицы для i -й головы [37].

ПОЗИЦИОННАЯ КОДИРОВКА (POSITIONAL ENCODING)

Трансформеры не имеют встроенной позиционной информации, поскольку они не используют рекуррентные или сверточные слои. Для добавления позиционной информации к входным векторам слов используется позиционная кодировка. Она добавляется к входным векторам перед подачей их на слой самовнимания.

Позиционная кодировка может быть статической или динамической. В оригинальной статье о трансформерах используется статическая позиционная кодировка, которая вычисляется по формуле:

$$\text{PE}(\text{pos}, i) = \sin\left(\frac{\text{pos}}{10000^{2i/d_{\text{model}}}}\right) \text{ если } i \text{ четное, } \cos\left(\frac{\text{pos}}{10000^{2i/d_{\text{model}}}}\right) \text{ если } i \text{ нечетное,} \quad (3)$$

где pos – позиция слова в последовательности;

i – индекс измерения вектора;

d_{model} – размерность вектора [38].

АРХИТЕКТУРА ТРАНСФОРМЕРА

На рисунке 1 показана архитектура трансформера, состоящая из двух основных компонентов: кодировщика (encoder) и декодера (decoder). Кодировщик состоит из N одинаковых слоев, каждый из которых содержит механизм многоголового самовнимания, а также промежуточные полносвязные слои и слои нормализации. Декодер также состоит из N одинаковых слоев, но включает дополнительный механизм многоголового внимания, предназначенный для взаимодействия с выходом кодировщика [8].

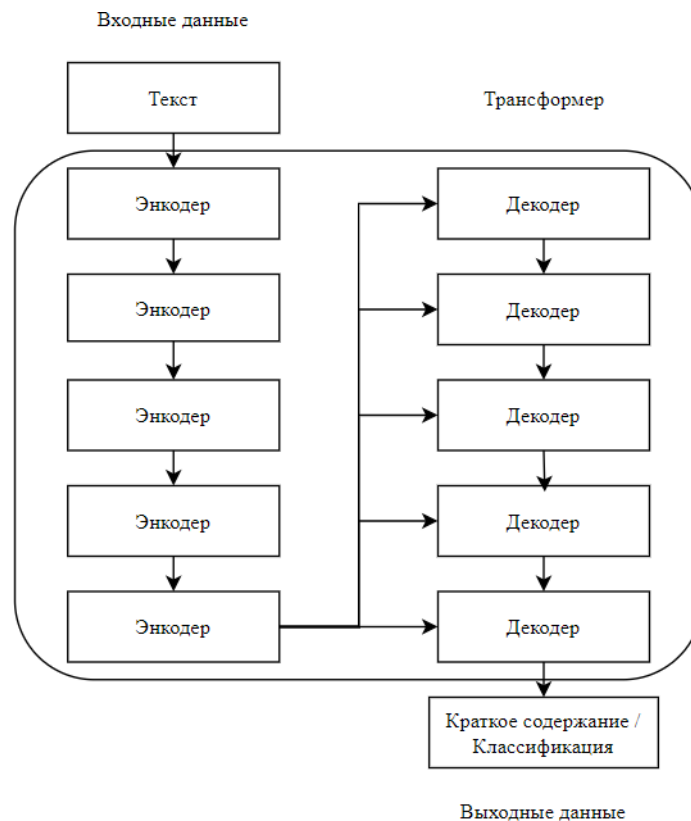


Рисунок 1 – Структура трансформера

Модели, основанные на трансформерах, могут быть предобучены на больших вырезках текста в режиме обучения без учителя (unsupervised learning) или полунатренированного обучения (semi-supervised learning). Затем они могут быть адаптированы для решения конкретных задач классификации с использованием методов, таких как fine-tuning или zero-shot learning [9].

ПОДГОТОВКА ДАННЫХ

Прежде всего, данные должны быть предобработаны и приведены к формату, подходящему для обучения модели трансформера. Входные данные представляют собой набор текстовых документов, каждый из которых содержит текст и соответствующий класс.

Текстовые данные могут быть предобработаны следующим образом:

- токенизация: текст разбивается на отдельные слова или подслова (токены);
- подготовка словаря: составляется словарь из уникальных токенов, присутствующих в наборе данных;

- конвертация токенов в индексы: токены заменяются соответствующими индексами из словаря;
- паддинг: последовательности дополняются нулями до максимальной длины для создания однородных размеров входных данных [9].

ФАЙН-ТЮНИНГ ТРАНСФОРМЕРА ДЛЯ КЛАССИФИКАЦИИ

Трансформеры, такие как BERT или RoBERTa, предобучены на больших вырезках текста и имеют хорошие знания о языке и контексте. Для адаптации предобученной модели к конкретной задаче классификации используется метод, называемый файн-тюнингом.

При файн-тюнинге модель дообучается на меньшем наборе размеченных данных, содержащих тексты и соответствующие классы. Затем используются выходные данные модели для получения вероятностей принадлежности к каждому классу.

Для адаптации архитектуры трансформера к задаче классификации можно добавить полносвязный слой с размерностью, равной количеству классов, к выходу модели. Затем применяется функция активации softmax (3).

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^K e^{z_j}} \text{ для } i = 1, \dots, K \text{ и } z = (z_1, \dots, z_K) \in \mathbb{R}^K, \quad (4)$$

где z_i – i -тый элемент входного вектора z ;

n – количество элементов в входном векторе;

e – основание натурального логарифма (приближенно равно 2.71828) [9].

Функция softmax гарантирует, что все выходные значения будут находиться в диапазоне от 0 до 1 и их сумма будет равна 1, что делает их интерпретируемыми как вероятности.

После обучения модели необходимо оценить ее производительность на тестовых данных. Для этого можно использовать метрики, такие как точность (accuracy), точность (precision), полнота (recall) и F1-меру (F1-score). Эти метрики позволяют измерить качество работы модели на задаче классификации.

Точность (accuracy) вычисляется как отношение правильно классифицированных текстов к общему числу текстов (4):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5)$$

Точность (precision) и полнота (recall) оценивают качество работы модели для каждого класса (5):

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (6)$$

где True Positives (TP) – количество правильно определенных положительных примеров;

False Positives (FP) – количество неправильно определенных положительных примеров;

False Negatives (FN) – количество неправильно определенных отрицательных примеров.

F1-мера представляет собой среднее гармоническое между точностью и полнотой (6):

$$F_1 = 2 \frac{(P \cdot R)}{(P + R)} \quad (7)$$

где P – Precision;

R – Recall [38].

F1-мера является полезной метрикой, когда классы несбалансированы или когда одновременно важны и точность, и полнота.

Для успешного применения трансформеров к задаче классификации неструктурированной текстовой информации следует учесть следующие аспекты:

- выбор архитектуры трансформера: необходимо определить, какая архитектура трансформера (BERT, GPT, RoBERTa и т. д.) лучше всего подходит для задачи и доступных данных;

- размерность модели и число слоев: более крупные модели и большее количество слоев могут привести к лучшей производительности, но также потребуют больше вычислительных ресурсов и времени на обучение;

– регуляризация: необходимо использовать методы регуляризации, такие как «dropout», для предотвращения переобучения модели;

– размер батча: необходимо определить оптимальный размер батча, чтобы обеспечить эффективное обучение и использование ресурсов [38].

ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

Для проведения эксперимента была выбрана задача классификации отзывов о продуктах на основе набора текстов, состоящего из 15 тысяч файлов. Данный набор содержит отзывы пользователей, размеченные на две категории: положительные и отрицательные.

Подготовка данных. Был разделён набор отзывов на обучающую и тестовую выборки в соотношении 80 % на 20 %. Далее проведена предобработка текста. Затем текст был преобразован в числовые последовательности. Также был собран словарь, состоящий из 5000 наиболее часто встречающихся слов. Максимальная длина последовательности была ограничена 128 словами.

ОБУЧЕНИЕ МОДЕЛИ

Была использована предобученная модель BERT и RoBERTa для классификации отзывов. Данные модели различаются в своих подходах к обучению. BERT использует две задачи предварительного обучения: задачу предсказания следующего предложения (Next Sentence Prediction, NSP) и задачу маскированного языкового моделирования (Masked Language Model, MLM). RoBERTa отказывается от задачи NSP, обучаясь только на задаче MLM. Это позволяет модели лучше изучить контекстуальные отношения между словами.

На вершине основной модели BERT был добавлен линейный слой для классификации. Модель была дообучена на обучающей выборке с использованием оптимизатора Adam, скорости обучения $2e-5$ и размера батча 32. Всего было проведено 3 эпохи обучения. В процессе обучения была использована кросс-энтропийная функция потерь.

Аналогично дообучалась и модель RoBERTa. Существенным различием является лишь большее количество эпох – 10.

После обучения моделей была проведена оценка их производительности на тестовой выборке, используя метрики Accuracy, Precision, Recall и F1-мера. Результаты для модели BERT и RoBERTa показаны на рисунке 2.

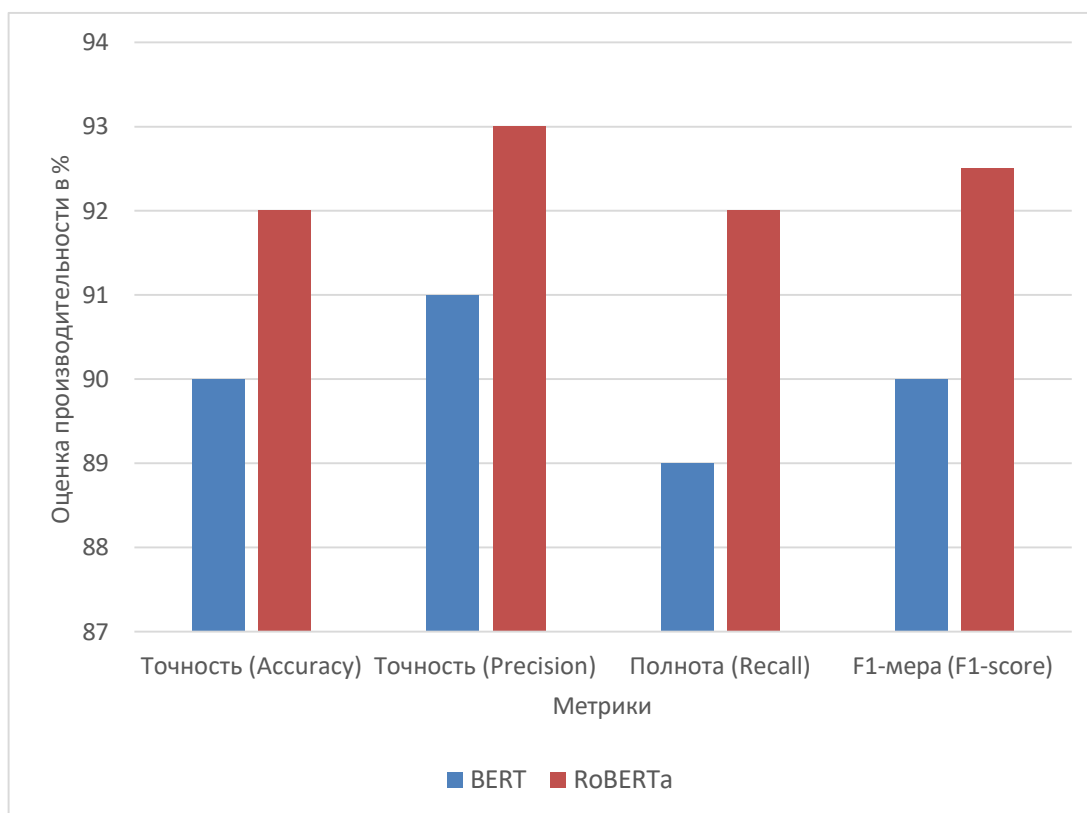


Рисунок 2 – Оценка производительности на основе метрик

Результаты эксперимента показывают, что предобученные модели BERT и RoBERTa способны эффективно классифицировать отзывы о продуктах на основе наборов текстовых отзывов. Для наглядности в таблице представлены показатели метрик моделей BERT, RoBERTa, а также сверточных (CNN) и рекуррентных (RNN) нейронных сетей.

Таблица – Показатели метрик нейросетевых моделей на тестовой выборке

Модель	Точность (Accuracy)	Точность (Precision)	Полнота (Recall)	F1-score
BERT	90 %	91 %	89 %	90 %
RoBERTa	92 %	93 %	92 %	92,5 %
CNN	85 %	86 %	84 %	85 %
RNN	83 %	85 %	82 %	83,5 %

Сверточные и рекуррентные нейронные сети были одними из первых успешно примененных моделей для обработки текстовых данных, и они действительно могут показывать хорошие результаты на многих задачах. Однако модели трансформеров, такие как BERT и RoBERTa, обычно превосходят их благодаря своим уникальным свойствам, таким как возможность эффективно моделировать долгосрочные зависимости в данных и лучшее использование контекстной информации.

Однако стоит отметить, что для улучшения производительности модели можно провести дополнительные эксперименты с различными гиперпараметрами, архитектурами трансформера и методами предобработки данных. Также можно использовать другие предобученные модели для сравнения результатов и определения наилучшей конфигурации для задачи классификации.

ЗАКЛЮЧЕНИЕ

В данной статье было рассмотрено применение моделей трансформера, в частности BERT и RoBERTa, для задачи классификации неструктурированной текстовой информации. Эксперименты показали, что модель BERT способна эффективно классифицировать отзывы о продуктах с высокими значениями метрик производительности.

Возможности и гибкость моделей трансформера открывают перспективы для дальнейших исследований и применения в различных задачах анализа текста. Благодаря использованию предобученных моделей и методам фэйн-тюнинга, разработчики и исследователи могут значительно сократить время обучения и требуемые вычислительные ресурсы, делая трансформеры доступными для широкого круга пользователей.

Список источников

- Гудфеллоу, И. Глубокое обучение / И. Гудфеллоу, Й. Бенжю, А. Курвилль. – Москва : Мир, 2018.
- Китов, В. В. Машинное обучение и анализ данных / В. В. Китов. – Москва : МЦНМО, 2016.
- Кудрин, Н. Д. Нейронные сети и глубокое обучение / Н. Д. Кудрин. – Москва : ФИЗМАТЛИТ, 2017.
- Лопатин, А. В. Технологии обработки и анализа текстовых данных с использованием глубокого обучения / А. В. Лопатин // Управление большими системами. – 2017. – № 66. – С. 136–152.
- Бухановский, А. В., Старостин А. М., Царёв А. В. Анализ текстовых данных с помощью нейросетевых алгоритмов / А. В. Бухановский, А. М. Старостин, А. В. Царёв // Известия ЮФУ. Технические науки. – 2016. – № 9 (180). – С. 15–27.
- Школа глубокого обучения. Физтех. URL: <https://www.dlschool.org/> (дата обращения: 12.05.2023).
- Специализация «Машинное обучение и анализ данных» // Coursera. – URL: <https://www.coursera.org/specializations/machine-learning-data-analysis> (дата обращения: 12.05.2023).
- Николенко, С. Нейронные сети и глубокое обучение / С. Николенко, А. Кадури, Е. Архангельская. – Санкт-Петербург : Питер, 2018.
- Барсегян, А. А. Технологии анализа текста / А. А. Барсегян, С. В. Купцов, А. А. Лошкарев // Системы и средства информатики. – 2011. – № 21. – С. 170–198.
- Специализация «Нейронные сети для обработки текстов» // Coursera. – URL: <https://www.coursera.org/specializations/neural-networks-for-text-processing> (дата обращения: 12.05.2023).
- Vaswani, A. Attention is all you need / A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez & I. Polosukhin // Advances in neural information processing systems. – 2017. – P. 5998–6008.
- Devlin, J. BERT: Pre-training of deep bidirectional transformers for language understanding / J. Devlin, M. W. Chang, K. Lee & K. Toutanova // arXiv preprint arXiv. – 2018. – 1810.04805.
- Radford, A. Improving language understanding by generative pre-training / A. Radford, K. Narasimhan, T. Salimans & I. Sutskever. – 2018. – URL <https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageunderstandingpaper.pdf>.
- Liu, Y. RoBERTa: A robustly optimized BERT pretraining approach / Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen & V. Stoyanov // arXiv preprint arXiv. – 2019. – 1907.11692.
- Wolf, T. Huggingface's transformers: State-of-the-art natural language processing / T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi & A. M. Rush // arXiv preprint arXiv. – 2020. – 1910.03771.
- Brown, T. B. Language models are few-shot learners / T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal & S. Agarwal // arXiv preprint arXiv. – 2020. – 2005.14165.

17. Pennington, J. Glove: Global vectors for word representation / J. Pennington, R. Socher & C. Manning // Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). – 2014. – P. 1532–1543.
18. Mikolov, T. Efficient estimation of word representations in vector space / T. Mikolov, K. Chen, G. Corrado & J. Dean // arXiv preprint arXiv. – 2013. – 1301.3781.
19. Lample, G. Cross-lingual language model pretraining / G. Lample & A. Conneau // arXiv preprint arXiv. – 2019. – 1901.07291.
20. Scikit-learn: Machine learning in Python, Pedregosa et al. // JMLR 12. – 2011. – P. 2825–2830.

References

1. Goodfellow, I., Bengio, Y., Courville, A. *Deep learning*. Moscow, Mir, 2018 (In Russ.).
2. Kitov, V. V. *Machine learning and data analysis*. Moscow, MTsNMO, 2016 (In Russ.).
3. Kudrin, N. D. *Neural networks and deep learning*. Moscow, FIZMATLIT, 2017 (In Russ.).
4. Lopatin, A. V. Technologies for processing and analyzing text data using deep learning. *Management of large systems*, 2017, no. 66, pp. 136–152 (In Russ.).
5. Bukhanovsky, A. V., Starostin, A. M., Tsarev, A. V. Analysis of text data using neural network algorithms. *News of the Southern Federal University. Technical science*, 2016, no. 9 (180), pp. 15–27 (In Russ.).
6. *School of deep learning. Phystech*. URL: <https://www.dlschool.org/> (accessed 05.12.2023) (In Russ.).
7. Specialization “Machine learning and data analysis”. *Coursera*. URL: <https://www.coursera.org/specializations/machine-learning-data-analysis> (accessed 05.12.2023) (In Russ.).
8. Nikolenko, S., Kadurin, A., Arkhangel'skaya, E. *Neural networks and deep learning*. St. Petersburg, Peter, 2018 (In Russ.).
9. Barseghyan, A. A., Kuptsov, S. V., Loshkarev, A. A. Technologies for text analysis. *Systems and means of informatics*, 2011, no. 21, pp. 170–198 (In Russ.).
10. Specialization “Neural networks for text processing”. *Coursera*. URL: <https://www.coursera.org/specializations/neural-networks-for-text-processing> (accessed 05.12.2023). (In Russ.).
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N. & Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 2017, pp. 5998–6008.
12. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv*, 2018, 1810.04805.
13. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. Improving language understanding by generative pre-training. 2018. URL <https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/languageunderstandingpaper.pdf>.
14. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D. & Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv*, 2019, 1907.11692.
15. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., & Rush, A. M. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv*, 2020, 1910.03771.
16. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P. & Agarwal, S. Language models are few-shot learners. *arXiv preprint arXiv*, 2020, 2005.14165.
17. Pennington, J., Socher, R., & Manning, C. Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
18. Mikolov, T., Chen, K., Corrado, G., & Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv*, 2013, 1301.3781.
19. Lample, G., & Conneau, A. Cross-lingual language model pretraining. *arXiv preprint arXiv*, 2019, 1901.07291.
20. Scikit-learn: Machine learning in Python, Pedregosa et al. *JMLR 12*, 2011, pp. 2825–2830.

Статья поступила в редакцию 08.04.2024; одобрена после рецензирования 26.04.2024; принята к публикации 14.05.2024.

The article was submitted 08.04.2024; approved after reviewing 26.04.2024; accepted for publication 14.05.2024.

УПРАВЛЕНИЕ В ОРГАНИЗАЦИОННЫХ СИСТЕМАХ

УДК 004.67:51-77

ПРОБЛЕМА ФОРМИРОВАНИЯ ПРОЕКТНЫХ КОМАНД В ИТ-СФЕРЕ

Зубова Анастасия Алексеевна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

аспирант, ORCID: 0000-0002-0042-5346, e-mail: an_vesnaa@mail.ru

Ажмухамедов Искандар Маратович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

доктор технических наук, профессор, ORCID: 0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

В статье подчеркнута актуальность проблемы формирования проектных команд для решения задач в ИТ-сфере. Рассмотренные статистические данные показали, что большая часть проектов в области ИТ является неуспешной. В результате анализа применимости в рамках поставленной задачи имеющихся подходов зарубежных и отечественных авторов был сделан вывод о том, что на данный момент отсутствует комплексная методика для формирования проектных команд для ИТ-проектов. Были выделены основные параметры, которые влияют на результат выполнения проекта в ИТ-сфере. В их число вошли: психологическая совместимость претендентов, их навыки, скорость работы сотрудников и т. п. Предложен показатель эффективности работы команды, который позволит руководителю ИТ-проекта обоснованно принимать решения при подборе исполнителей. На основе данного показателя предложена формальная постановка задачи.

Ключевые слова: ИТ-проект, команда проекта, формирование проектных команд, оценка эффективности работы команды

THE PROBLEM OF FORMING PROJECT TEAMS IN THE IT FIELD

Zubova Anastasia A., Astrakhan Tatishchev State University, Tatishchev St., 20a, Astrakhan, 414056, Russian Federation,

graduate student, e-mail: an_vesnaa@mail.ru

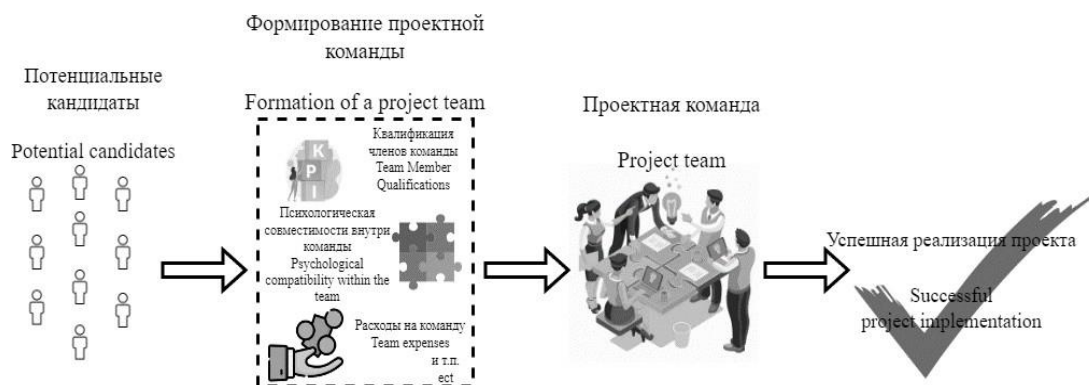
Azhmukhamedov Iskandar M., Astrakhan State University, Tatishchev St., 20a, Astrakhan, 414056, Russian Federation,

Doct. Sci. (Engineering), Professor, ORCID: 0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

The article emphasizes the relevance of the problem of forming project teams to solve problems in the IT field. The statistical data reviewed showed that most projects in the IT field are unsuccessful. As a result of analyzing the applicability of existing approaches of foreign and domestic authors within the framework of the task, it was concluded that at the moment there is no comprehensive methodology for forming project teams for IT projects. The main parameters that influence the outcome of a project in the IT field were identified. These included: the psychological compatibility of applicants, their skills, the speed of work of employees, etc. An indicator of team performance is proposed, which will allow the IT project manager to make informed decisions when selecting performers. Based on this indicator, a formal formulation of the problem is proposed.

Keywords: IT project, project team, formation of project teams, assessment of team performance

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

Кадровый ресурс является ключевым элементом в любой организации. Правильное распределение исполнителей в большинстве случаев напрямую влияет на показатель эффективности работы компании.

На сегодняшний день имеются два подхода к решению кадровой задачи. Первый – предусматривает наличие сформированного устойчивого коллектива, обладающего определенным опытом решения задач в той или иной сфере. В этом случае под сложившийся коллектив подбираются проекты, которые могут быть им успешно реализованы.

Другой подход заключается в том, что под имеющийся проект подбирается отдельная команда, существование которой ограничено сроком реализации данного проекта. Обычно такая команда называется «проектной». После достижения целей проекта команда расформировывается, и для следующего проекта подбирается новая команда.

В сфере разработки программного обеспечения в последние годы доминирующим является второй подход. Это связано как со спецификой проектов в IT-сфере, так и с широким набором сервисов, которые предоставляют современные средства коммуникации. Возможность вести работу над IT-проектами из любой точки мира, согласовывая работу команды через сеть Интернет, позволяет руководителю проекта привлекать к работе наиболее компетентных, с его точки зрения, сотрудников, фактически до бесконечности расширяя возможности кадрового отбора.

При этом, по данным информационного агентства International Data Corporation (IDC), в мире отмечается ежегодный рост расходов на информационно-коммуникационные технологии. Цифровизация затронула все сферы жизни, а следовательно, можно прогнозировать устойчивый рост спроса на проекты в сфере IT [1].

Однако, по данным компании Standish Group, лишь около 31 % IT-проектов завершаются успешно [2]. 69 % – это неуспешные (проваленные) проекты. Такая неутешительная статистика обусловлена тем, что проекты в области IT имеют ряд особенностей, таких как [3–5]:

- специфичность сферы информационных технологий как области знаний и сферы практической деятельности. В связи с данной особенностью заказчик не всегда может правильно формализовать собственное видение проекта, что влечет за собой нечеткое формирование технического задания;
- корректировка задач и работ в процессе выполнения проекта. В силу того, что разрабатываемые IT-системы встраиваются в среду организации заказчика, необходимо постоянно адаптировать систему к меняющимся условиям во время выполнения проекта. Такие корректировки могут возникнуть на любом этапе реализации проекта;
- функциональная разнородность работ в проекте. Область IT включает в себя большое количество различных технологий по обработке, хранению и передаче информации. В связи с этим для выполнения каждого отдельного проекта часто требуются узкоспециализированные специалисты;
- ставка дисконтирования для таких проектов должна быть выше средней ставки дисконтирования, используемой в организации, так как считается, что риск данных проектов превосходит средний показатель риска. Причинами этого является то, что, как правило, проекты IT являются непрофильными и уникальными для организации;
- процесс построения эффективных коммуникаций в данной сфере является непростой задачей, поскольку потенциальные участники команд – это специалисты с разным опытом и уровнем компетенций. Поэтому, по данным компании theStandishGroup, это становится частой причиной неуспешности проектов.

Исходя из вышеизложенного, разработка методики рационального подбора эффективной команды для реализации IT-проектов является весьма актуальной. Для решения данной задачи необходимо проанализировать сильные и слабые стороны подходов к формированию проектных команд, а также осуществить формальную постановку задачи, что и явилось **целью данной работы**.

АНАЛИЗ ИМЕЮЩИХСЯ ПОДХОДОВ К РЕШЕНИЮ ЗАДАЧИ ФОРМИРОВАНИЯ ПРОЕКТНЫХ КОМАНД

Исследованиями в области распределения трудовых ресурсов занимались как зарубежные, так и отечественные ученые. Среди зарубежных исследователей можно выделить работы R. Belbin, B. Tuckman и Jensen M., G. Dantzig, A. Costa, F. Ramos, M. Perkusich, A. Freire, H. Almeida и A. Perkusich.

Среди отечественных авторов наиболее известны работы Л. Канторовича, А. Катаева, А. Тынчинского, Л. Сабадош, Н. Косенко, М. Гаховой.

Одну из первых методик формирования проектной команды предложил в 1970-х годах М. Белбин [6–7]. Основная цель заключается в том, чтобы создать сбалансированную команду, где разные стили работы компенсируют друг друга. Лидер команды использует результаты тестирования для лучшего понимания сильных сторон и предпочтений участников, чтобы эффективно распределить роли, избежать конфликтов и повысить производительность.

В 1977 г. Б. Такманом и М. Дженсен также была предложена модель формирования проектной команды [8–9]. В основу создания модели были положены результаты исследований Такмана, проведенных в контексте наблюдения за поведением групп в различных ситуациях. Такмана интересовал процесс общения и сотрудничества в группе, а также как группы эффективно решают задачи и преодолевают препятствия. Такман, как и М. Белбин, составил собственную иерархию ролей для построения эффективной команды.

Представленные методики полностью основаны на вербальных оценках результатов наблюдений. Такое решение задачи сложно строго формализовать для дальнейшей автоматизации данного процесса.

Работы Джорджа Данцига, выдающегося американского математика, по различным аспектам оптимизации, также могут быть применены при формировании команд. Метод оптимизации известен как «симплекс-метод» [10–11]. Этот метод применяется для решения задач линейного программирования, что является формой математической оптимизации, при которой стремятся минимизировать или максимизировать линейную функцию от нескольких переменных с учетом линейных ограничений.

Теория оптимального распределения ресурсов, развитая Леонидом Канторовичем [12], посвящена проблемам эффективного распределения ресурсов в различных областях, таких как экономика, производство и логистика. Эта теория стала важным инструментом для оптимизации использования ресурсов с целью максимизации общей полезности или прибыли. Во многих случаях теория оптимального распределения использует методы линейного программирования.

Так, например, в работе А. В. Катаева и А. В. Тычинского [13] приводятся математические модели и алгоритмы для оптимизации выбора численности проектной команды с учетом затрат на выполнение работ и бюджета проекта. Для решения поставленной задачи авторы используют методы целочисленной оптимизации.

Основной акцент при формировании проектной команды исследователи делают на минимизацию количества участников. Для нахождения решения авторами используется симплекс-метод.

Однако данные подходы применимы только в случае, если возможна постановка задачи линейного программирования. То есть оптимизируемая функция, а также ограничения являются строго линейными. В случае, когда наблюдается взаимное влияние параметров задачи друг на друга, а при формировании проектной команды в IT-сфере именно так и происходит, данные условия не выполняются. И, как следствие, указанный подход напрямую не может быть использован.

Группа исследователей Федерального университета Кампина-Гранде опубликовала статью, в которой было описано применение генетического алгоритма для формирования проектных команд в организационных системах [14].

В статье авторы используют подход SBSE (Search-based software engineering) для формирования нескольких команд на основе Scrum-инструментов. Задача разбивается на две части. Прежде всего изучаются технические знания разработчиков на основе информации о проектах, в которых они ранее участвовали. Создаются профили сотрудников на основе тегов для разработчиков и списка проектов.

После этого создается структура данных для запуска генетического алгоритма распределения сотрудников по проектам. Таким образом, формируются команды с максимальной технической совместимостью с учетом требований проекта и навыков сотрудников.

Для решения поставленной задачи был использован генетический алгоритм, потому что, по мнению авторов, он один из наиболее подходящих методов комбинаторной оптимизации.

Авторы методики формируют проектную команду только на основе сопоставления навыков, которыми обладают потенциальные члены команд, и требований проекта. При этом такой подход не учитывает другие параметры, от которых зависит успешное завершение проекта. Например, психологическую совместимость исполнителей, затраты на оплату труда и т. п.

В работе Л. Сабадош, Н. Косенко, М. Гаховой [15] представлено исследование, направленное на создание системы поддержки принятия решений при формировании команды проекта.

Авторы используют принцип полезности П. Фишборна [16] для формирования функции полезности кандидата.

Предлагается использовать метод компараторной идентификации параметров модели оценивания К. Петрова [17]. Теоретической основой метода является положение теории полезности, согласно которому более предпочтительная альтернатива имеет более высокую полезность, а эквивалентные альтернативы – равную полезность.

Для решения поставленной задачи исследователи выполнили структурную и параметрическую идентификацию модели оценивания кандидатов. Сформировали аддитивную функцию полезности для оценки потенциального члена команды.

Однако авторы подчеркивают, что данная задача является задачей многокритериальной оптимизации и её решение требует учета трудноформализуемых факторов, что является препятствием для применения традиционных математических методов.

Таким образом, проведенный анализ позволяет выделить ряд значимых параметров, которые авторы рассмотренных выше работ считают приоритетными при формировании проектных команд. Среди них: психологическая совместимость претендентов, их навыки, бюджет проекта, временные рамки проекта и т. п.

При этом процесс формирования проектных команд для выполнения IT-проектов обладает специфическими особенностями. Большинство предлагаемых подходов не учитывают одновременно ряд значимых параметров, влияющих на эффективность проектных команд, а также не позволяют комплексно оценить качество того или иного набора исполнителей для реализации проекта.

ФОРМАЛЬНАЯ ПОСТАНОВКА ЗАДАЧИ РАЦИОНАЛЬНОГО ПОДБОРА КОМАНДЫ ДЛЯ IT-ПРОЕКТА

Исходя из вышеизложенного, можно констатировать, что задача формирования проектной команды является многокритериальной и слабоструктурированной, значительная часть параметров, влияющих на решение, часто носит экспертный характер и выражается в виде вербальных (нечисловых) оценок. Последнее, в свою очередь, приводит к тому, что задача становится трудно формализуемой.

Онтологическая схема предметной области приведена на рисунке.

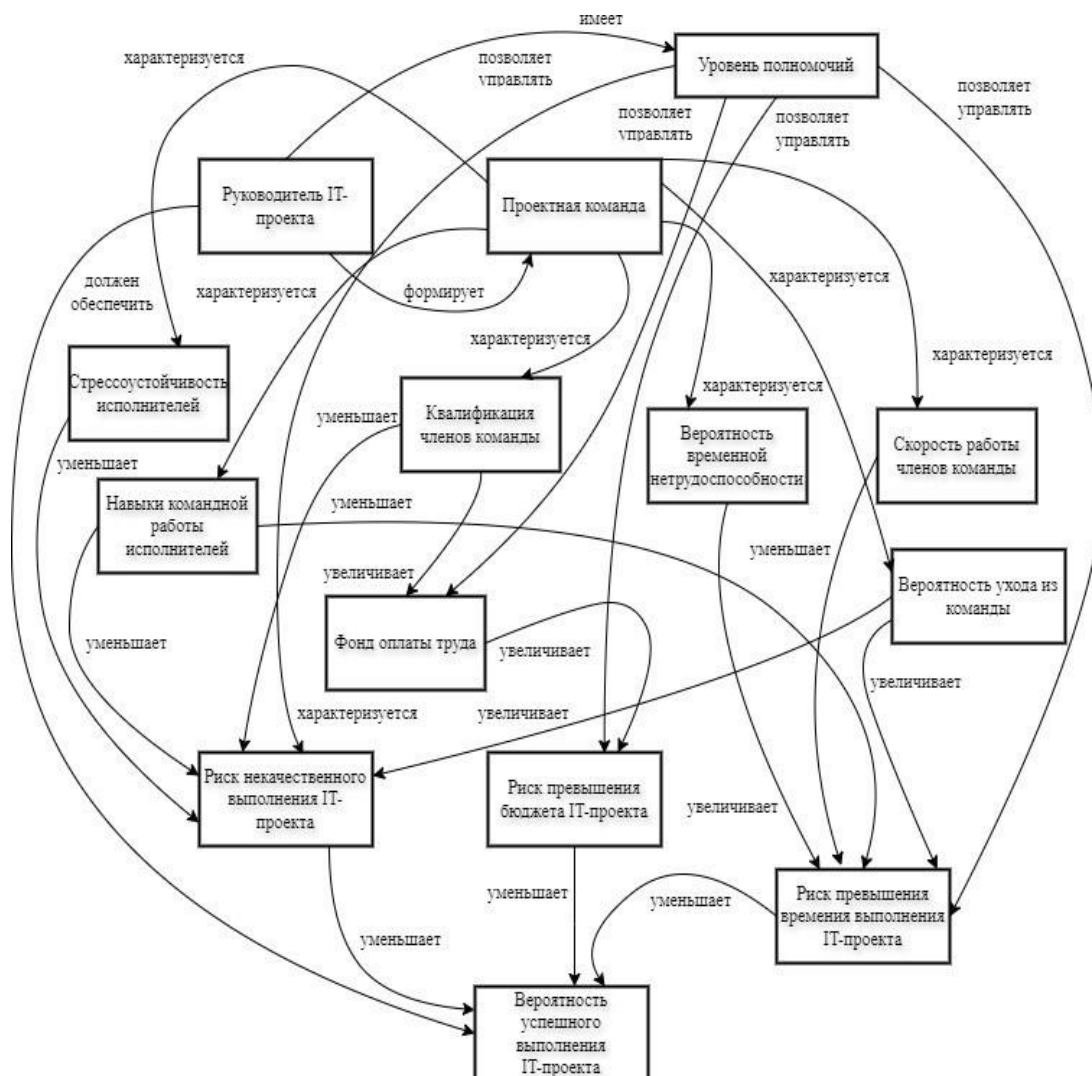


Рисунок – Онтологическая схема предметной области

Введем следующие обозначения:

I – множество потенциальных исполнителей;

$I = \{I_m\}$, ($1 \leq m \leq |I|$); в свою очередь, I_m характеризуется следующим набором параметров $(C_m; KS_m; Z_m; S_m; Q_m)$;

C_m – набор компетенций m -го члена команды;

K_m – навыки командной работы m -го члена команды;

ST_m – стрессоустойчивость m -го члена команды;

Z_m – заработная плата m -го члена команды;

S_m – скорость работы m -го члена команды;

Q_m – качество работы m -го члена команды;

$I^k \subseteq I$ – k -а команда; k -е подмножество множества I ($1 \leq |I^k| \leq |I|$); $|I| \equiv N$; $|I^k| \equiv N^k$;

$I^k = \{I_m^k\}$ – множество исполнителей в k -м наборе; в свою очередь, I^k характеризуется следующим набором параметров $(Q^k; R^k; Z^k; T^k)$;

Q^k – качество выполнения проекта k -й командой ($0 \leq Q^k \leq 1$); является функцией от Q_m : $Q^k = Q^k(Q_m) \quad m = \overline{1, N^k}$;

R^k – риск невыполнения проекта k -й командой ($0 \leq R^k \leq 1$); является функцией от C_m ; K_m ; $ST_m = R^k(C_m; K_m; ST_m) \quad m = \overline{1, N^k}$;

Z^k – стоимость выполнения проекта k -й командой ($0 \leq Z^k \leq +\infty$); является функцией от Z_m : $Z^k = Z^k(Z_m) \quad m = \overline{1, N^k}$;

T^k – время выполнения проекта k -й командой ($0 \leq T^k \leq +\infty$); является функцией от S_m : $T^k = T^k(S_m) \quad m = \overline{1, N^k}$;

T – допуск по времени выполнения проекта;

\mathcal{E}^k – эффективность k -й команды исполнителей при выполнении проекта ($0 \leq \mathcal{E}^k \leq 1$).

Для расчета показателя эффективности проектной команды с учетом введенных обозначений предлагается использовать следующую формулу (1):

$$\mathcal{E}^k = \frac{[1-R^k] \cdot [(1+T)-T^k] \cdot [Q^k]}{Z^k} \quad (1)$$

Рассчитанное по формуле 1 значение эффективности работы команд позволяет руководителю IT проекта ранжировать различные варианты подбора исполнителей. Предложенный критерий может быть положен в основу разработки соответствующей методики, в рамках которой необходимо будет предусмотреть модели, методы и алгоритмы, позволяющие адекватно оценивать все входящие в формулу 1 параметры.

Исходя из их природы, можно констатировать, что данная задача требует междисциплинарного подхода и может быть решена с использованием результатов и методов из различных областей знаний: IT-технологий, социологии, психологии, теории оценки рисков, менеджмента и т. д.

Подбор проектной команды, необходимой для успешного выполнения проекта, является задачей переборного типа. В комбинаторике количество сочетаний различных элементов можно найти по следующей формуле (2):

$$V = \sum_{k=1}^N C_N^k \quad (2)$$

где N – мощность множества потенциальных исполнителей;

k – количество членов в команде;

C_N^k – количество возможных сочетаний из N по k .

То есть данная задача является NP-полной, и ее решение методом полного перебора при больших значениях N и k затруднено.

Наиболее рациональным методом решения данного вида задач является эволюционно-генетический подход, который используется для построения алгоритмов поиска оптимальных решений с использованием механизмов, аналогичных естественному отбору в природе [18–19].

ЗАКЛЮЧЕНИЕ

Проведенный анализ показал, что на данный момент отсутствует комплексная методика, позволяющая решить задачу рационального подбора проектных команд для выполнения IT-проектов. В рамках формализации проблемы была предложена формула оценки эффективности команд, состоящих из различных наборов исполнителей.

Это позволяет перейти к разработке методики рационального подбора таких коллективов. Данная методика, опираясь на междисциплинарный подход, должна комплексно учитывать специфические особенности разработки проектов в IT-сфере, квалификацию и личностные характеристики исполнителей.

Список источников

1. Gartner прогнозирует рост мировых ИТ-расходов на 6,8 % в 2024 году // Интерфакс. – URL: <https://www.interfax.ru/business/940611> (дата обращения: 02.04.2024).
2. Жестокая статистика: 69 % ИТ-проектов проваливаются или не достигают цели. И вот почему // Myfin.by. – URL: <https://myfin.by/stati/view/zhestokaya-statistika-69-it-proektov-provalivayutsya-ili-ne-dostigayut-celi-i-vot-pochemu> (дата обращения: 02.04.2024).
3. Власова, Т. И. Особенности управления качеством в инновационных ИТ-проектах / Т. И. Власова // Вестник Государственного университета просвещения. Серия: Экономика. – 2020. – № 3. – С. 61–71.
4. Куликова, Л. Л. Особенности оценки эффективности ИТ-проектов / Л. Л. Куликова, В. Ю. Швакин // Вестник ИрГТУ. – 2010. – № 3 (43). – С. 153–159.
5. Минайлова, М. С. Особенности эффективных коммуникаций в ИТ-проекте / М. С. Минайлова // Экономика и социум. – 2020. – № 5–1 (72). – С. 855–860.
6. Belbin, R. M. *Team Roles at Work* Routledge / R. M. Belbin. – Routledge, 2010. – 153 p.
7. Belbin, R. M. *Management Teams. Why They Succeed or Fail* / R. M. Belbin. – London : Elsevier, 2004. – 193 p.
8. Tuckman, B. Developmental Sequence in Small Groups / B. Tuckman // *Psychological Bulletin*. – 1965. – Vol. 63. – P. 384–399.
9. Tuckman, B. Stages of Small Group Development / B. Tuckman, M. Jensen // *Group and Organizational Studies*. – 1977. – Vol. 2. – P. 419–427.
10. Dantzig, G. B. *Linear programming Vol. 1. Introduction* / G. B. Dantzig, M. N. Thapa. – Springer-Verlag, 1997. – Vol. 1. – 435 p.
11. Dantzig, G. B. *Linear programming. Vol. 2. Theory and Extensions* / G. B. Dantzig, M. N. Thapa // Springer-Verlag, 2003. – 209 p.
12. Канторович, Л. В. Оптимальные решения в экономике / Л. В. Канторович, А. Б. Горстко. – Москва : Наука, 1972. – 233 с.
13. Катаев, А. В. Формирование и оптимизация проектных команд: математические модели и алгоритмы / А. В. Катаев, А. В. Тычинский // Вестник ТИУиЭ. – 2021. – № 1 (33). – С. 57–60.
14. Costa, A., Team Formation in Software Engineering: A Systematic Mapping Study / A. Costa, F.Ramos, M. Perkusich, E. Dantas, E. Dilorenzo, F. Chagas, A. Meireles, D. Albuquerque, L. Silva, H. Almeida, A. Perkusich // *IEEE Access: The Multidisciplinary Open Access Journal*. – 2020. – Vol. 8. – P. 145687–145712.
15. Сабадош, Л. Ю. Система поддержки принятия решений по формированию проектной команды / Л. Ю. Сабадош, Н. В. Косенко, М. А. Гахова // Экономика. Информатика. – 2012. – № 19–1 (138). – С. 185–189.
16. Fishburn, P. C. *Utilitu theory for decision making* / P. C. Fishburn. New York, 1970.
17. Петров, К. Э. Компараторная структурно-параметрическая идентификация моделей скалярного многофакторного оценивания : монография / К. Э. Петров, В. В. Крючковский. – Херсон : Олди-плюс, 2009. – 294 с.
18. Батищев, Д. И. Применение генетических алгоритмов к решению задач дискретной оптимизации / Д. И. Батищев, Е. А. Неймарк, Н. В. Старостин. – Новгород : Изд-во ННГУ, 2007. – 88 с.
19. Синадский, Н. И. Автоматизация тестирования сетевых средств защиты информации на основе применения эволюционно-генетического подхода / Н. И. Синадский, А. В. Агафонов // МСим. – 2018. – № 2 (46). – С. 125–134.

References

1. Gartner forecasts global IT spending to grow by 6.8 % in 2024. *Interfax*. URL: <https://www.interfax.ru/business/940611> (accessed 04.02.2024) (In Russ.).
2. Cruel statistics: 69% of IT projects fail or do not achieve their goals. And here's why. *Myfin.by*. URL: <https://myfin.by/stati/view/zhestokaya-statistika-69-it-proektov-provalivayutsya-ili-ne-dostigayut-celi-i-vot-pochemu> (accessed 04.02.2024) (In Russ.).
3. Vlasova, T. I. Features of quality management in innovative IT projects. *Bulletin of the State University of Education. Series: Economics*, 2020, no. 3, pp. 61–71 (In Russ.).
4. Kulikova, L. L., Shvakin, V. Yu. Features of assessing the effectiveness of IT projects. *Bulletin of ISTU*, 2010, no. 3 (43), pp. 153–159 (In Russ.).
5. Minailova, M. S. Features of effective communications in an IT project. *Economy and society*, 2020, no. 5–1 (72), pp. 855–860 (In Russ.).
6. Belbin, R. M. *Team Roles at Work* Routledge. Routledge, 2010. 153 p.
7. Belbin, R. M. *Management Teams. Why They Succeed or Fail*. London, Elsevier, 2004. 193 p.
8. Tuckman, B. Developmental Sequence in Small Groups. *Psychological Bulletin*, 1965, vol. 63, pp. 384–399.
9. Tuckman, B., Jensen, M. Stages of Small Group Development. *Group and Organizational Studies*, 1977, vol. 2, pp. 419–427.
10. Dantzig, G. B., Thapa, M. N. *Linear programming Vol. 1. Introduction*. Springer-Verlag, 1997. Vol. 1. 435 p.
11. Dantzig, G. B., Thapa, M. N. *Linear programming. Vol. 2. Theory and Extensions*. Springer-Verlag, 2003. 209 p.
12. Kantorovich, L. V., Gorstko, A. B. *Optimal solutions in economics*. Moscow, Publishing House “Nauka”, 1972. 233 p. (In Russ.).
13. Kataev, A. V., Tychinsky, A. V. Formation and optimization of project teams: mathematical models and algorithms. *Bulletin of TIUE*, 2021, no. 1 (33), pp. 57–60 (In Russ.).

14. Costa, A., Ramos, F., Perkusich, M., Dantas, E., Dilorenzo, E., Chagas, F., Meireles, A., Albuquerque, D., Silva, L., Almeida, H., Perkusich A. Team Formation in Software Engineering: A Systematic Mapping Study. *IEEE Access: The Multidisciplinary Open Access Journal*, 2020, vol. 8, pp. 145687–145712.

15. Szabadosh, L. Yu., Kosenko, N. V., Gakhova M. A. Decision support system for forming a project team. *Economics. Computer Science*, 2012, no. 19–1 (138), pp. 185–189 (In Russ.).

16. Fishburn, P. C. *Utility theory for decision making*. New York, 1970.

17. Petrov, K. E., Kryuchkovsky, V. V. *Comparative structural-parametric identification of scalar multifactor assessment models : monograph*. Kherson, Oldie-plus, 2009. 294 p. (In Russ.).

18. Batishchev, D. I., Neimark, E. A., Starostin, N. V. Application of genetic algorithms to solving discrete optimization problems. Novgorod, Publishing House of Nizhny Novgorod State University, 2007. 88 p. (In Russ.).

19. Sinadsky, N. I., Agafonov, A. V. Automation of testing of network information security tools based on the use of an evolutionary-genetic approach. *MSiM*, 2018, no. 2 (46), pp. 125–134 (In Russ.).

Статья поступила в редакцию 27.04.2024; одобрена после рецензирования 17.05.2024; принята к публикации 17.05.2024.

The article was submitted 27.04.2024; approved after reviewing 17.05.2024; accepted for publication 17.05.2024.

УДК 004.855.5

РАЗРАБОТКА МОДЕЛИ АВТОМАТИЧЕСКОГО КЛАССИФИКАТОРА ОБРАЩЕНИЙ ГРАЖДАН

Евдошенко Олег Игоревич, МИРЭА – Российский технологический университет, 119454, Российская Федерация, г. Москва, пр-т Вернадского, 78,

кандидат технических наук, доцент, ORCID: 0000-0002-1318-2822, e-mail: evdoshenko@mirea.ru

Евдошенко Наиля Ильнуровна, Астраханский государственный архитектурно-строительный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 18,

магистрант, ORCID: 0000-0003-1787-0259, e-mail: mvnailya@mail.ru

Цель статьи – разработка модели автоматического классификатора обращений граждан на основе методов машинного обучения. Были изучены существующие алгоритмы классификации, такие как метод опорных векторов, деревья решений, наивный Байесовский классификатор, нейронные сети. Обоснован выбор алгоритма классификации, исходя из поставленной задачи. Создана модель структуры набора данных для обучения. Представлены основные этапы создания модели, включая выбор подходящего алгоритма классификации, сбор и предобработку данных, обучение. Описана предобработка и подготовка данных перед обучением с использованием лемматизации текста, удаления стоп-слов, цифр и иностранных символов. Для разбивки слов на ключевые слова рассмотрен алгоритм TF-IDF. За основу в разработке алгоритма классификации обращений по категориям и исполнителям был взят линейный метод опорных векторов. Описан метод векторизации текста для представления обращений в виде числовых признаков с использованием векторизатор. Представлена схема функционирования алгоритма классификации обращений граждан по категории и исполнителям. Рассмотренный подход позволяет автоматизировать процесс классификации обращений граждан по категориям и исполнителям, что способствует эффективности работы организации и качества обслуживания граждан.

Ключевые слова: обращение, модель, классификация, машинное обучение, категория, векторизация

DEVELOPMENT OF A MODEL OF AN AUTOMATIC CLASSIFIER OF CITIZENS' APPEALS

Evdoshenko Oleg I., MIREA – Russian Technological University, 78 Vernadsky Ave., Moscow, 119454, Russian Federation,

Evdoshenko Nailya I., Astrakhan State University of Architecture and Civil Engineering, 18 Tatishchev St., Astrakhan, 414056, Russian Federation,

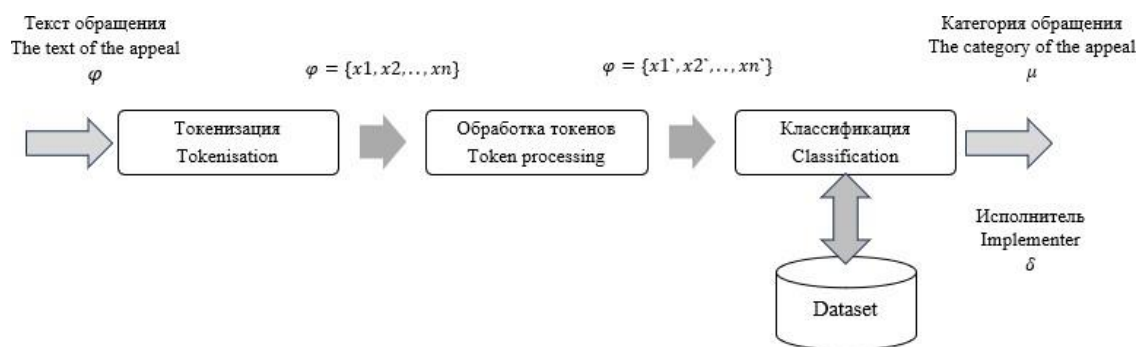
Cand. Sci. (Engineering), Assistant Professor, ORCID: 0000-0002-1318-2822, e-mail: evdoshenko@mirea.ru

undergraduate student, ORCID: 0000-0003-1787-0259, e-mail: mvnailya@mail.ru

The purpose of the article is to develop a model of an automatic classifier of citizens' appeals based on machine learning methods. Existing classification algorithms such as the support vector machine, decision trees, naive Bayesian classifier, and neural networks were studied. The choice of the classification algorithm based on the task is justified. A model of the training dataset structure has been created. The main stages of model creation are presented, including the selection of a suitable classification algorithm, data collection and preprocessing, and training. It describes the preprocessing and preparation of data before training using lemmatization of text, removal of stop words, numbers and foreign characters. The TF-IDF algorithm is considered for splitting words into keywords. The linear method of support vectors was used as a basis for the development of an algorithm for classifying appeals by categories and performers. A text vectorization method is described for representing appeals in the form of numerical signs using a vectorizer. The scheme of functioning of the algorithm for classifying citizens' appeals by category and performers is presented. The considered approach makes it possible to automate the process of classifying citizens' appeals by categories and performers, which contributes to the efficiency of the organization and the quality of service to citizens.

Keywords: appeal, model, classification, machine learning, category, vectorization

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

Современное социальное государство должно создавать возможность обеспечить своевременное получение обращений граждан и своевременное решение аварийных ситуаций. Целесообразно повысить внимание общественности к использованию различных каналов участия в управлении: сложившиеся позитивные примеры на практике позволяют повысить веру граждан в отзывчивость власти и увеличить область использования имеющихся инструментов связи [1].

Ежедневно население обращается по вопросам жилищно-коммунального хозяйства, связанные с отоплением, водоснабжением, газоснабжением, уборкой территории и другими коммунальными услугами.

На сегодняшний момент поступающие обращения подвергаются ручной обработке с присвоением категории аварийной ситуации и исполнительного органа, что затрудняет их оперативное выполнение.

Классификация обращений граждан с использованием машинного обучения является важным инструментом для организаций и государственных учреждений, которые занимаются обработкой большого объема запросов и жалоб от граждан. Такая классификация позволяет автоматизировать процесс обработки обращений, ускорить реакцию на них и повысить качество обслуживания.

Целью исследования является разработка модели автоматического классификатора обращений граждан для повышения эффективности обработки и повышения качества обслуживания.

МЕТОДЫ ИССЛЕДОВАНИЯ

Для автоматической классификации обращений граждан можно использовать различные алгоритмы машинного обучения, в зависимости от специфики задачи и типа данных. Рассмотрим несколько наиболее распространенных алгоритмов [2, 3]:

1. Наивный байесовский классификатор (Naive Bayes Classifier) – это алгоритм классификации, основанный на теореме Байеса. Он позволяет классифицировать текстовые данные, определяя, к какой категории они относятся, на основе анализа вероятностей соответствующих слов и фраз в тексте. Наивный байесовский классификатор часто используется для классификации электронных писем или текстовых сообщений.

2. Деревья решений (Decision Trees) – это алгоритм классификации, который создает дерево ветвлений, на каждом узле которого осуществляется разбиение данных на две или более ветви в зависимости от определенных условий. Деревья решений могут быть использованы для классификации обращений граждан по типу аварии или уровню приоритетности.

3. Метод опорных векторов (Support Vector Machines, SVM) – это алгоритм классификации, который позволяет разделять исходные данные на две категории. Данный метод может быть использован для классификации обращений граждан, основываясь на определенных признаках, таких как местоположение или время.

4. Нейронные сети (Neural Networks) – это алгоритмы машинного обучения, моделирующие работу человеческого мозга. Нейронные сети включают в себя множество взаимосвязанных нейронов, которые обрабатывают информацию и определяют, к какой категории она относится. Нейронные сети могут быть использованы для классификации обращений граждан, основываясь на большом количестве признаков.

В результате данного обзора в основу модели автоматического классификатора обращений по категориям и исполнителям был взят метод опорных векторов, а именно линейный метод опорных векторов, который строит гиперплоскость для разделения классов в многомерном пространстве признаков.

Он имеет несколько преимуществ:

- хорошая обобщающая способность: LinearSVC хорошо работает на различных типах данных и демонстрирует хорошие результаты на новых, неизвестных данных;
- эффективность: LinearSVC обладает высокой скоростью работы и эффективно обрабатывает большие объемы данных;

– хорошая интерпретируемость: LinearSVC позволяет легко интерпретировать результаты классификации и понять, какие признаки влияют на принятое решение;

– поддерживает многоклассовую классификацию: LinearSVC может быть использован для решения задач классификации с несколькими классами.

Перед изучением модели машинного обучения одним из важнейших шагов является предобработка данных.

Предобработка включает в себя несколько шагов, изображенных на рисунке 1.

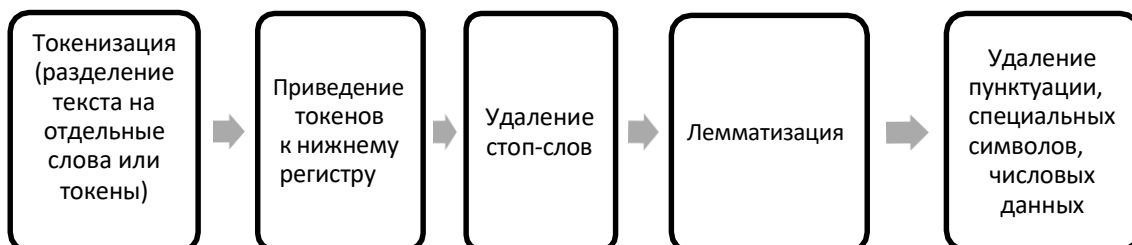


Рисунок 1 – Шаги предобработки текста обращений

Эти шаги помогают улучшить качество модели, уменьшить размер словаря и ускорить обучение.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Модель автоматического классификатора обращений представляет собой набор параметров для каждой аварийной ситуации, а также алгоритмы обработки информации для разбиения обращений граждан на категории, определения числа категорий аварийных ситуаций и ключевых слов.

Набор параметров для каждой аварийной ситуации представляет собой множество в формате:

$$O = \{\varphi, \mu, \delta\},$$

где φ – текст обращения;

μ – класс обращения (категория);

δ – исполнитель.

На рисунке 2 представлен фрагмент обработки входящего обращения и определения категории (вероятности отношения к определённой категории) обращения μ .

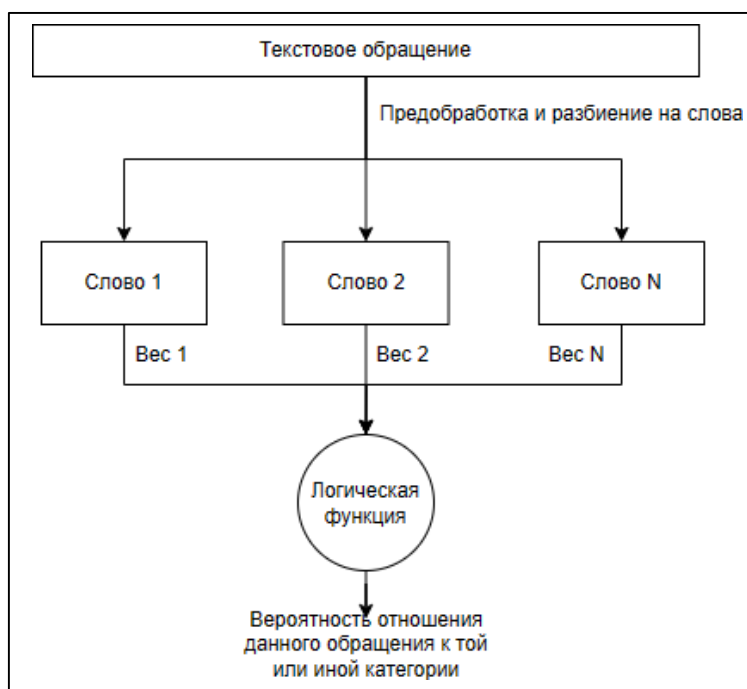


Рисунок 2 – Обработка входящего сообщения

Для того чтобы использовать сформированный набор данных для создания модели машинного обучения, его необходимо подготовить, выполнив все шаги в соответствии с рисунком 1.

Шаг 1. Текст сообщения φ разбить на токены.

В машинном обучении для подготовки данных при обработке естественного языка используются специальные обработчики исходного текста (токенизаторы). Они позволяют разбить текст на блоки для подачи в модель. При обработке текста первым шагом применяется токенизация, в результате которой на выходе получается набор токенов (подстрок), которые представляют из себя предложения, слова или отдельные символы [4].

Шаг 2. Удаление стоп-слов, лемматизация.

Токены из датасета приводятся к базовой форме (лемме) с помощью лемматизации, удаляются стоп-слова, цифры и иностранные символы. Новые данные сохраняются в новые файлы для дальнейшей работы при обучении.

Шаг 3. Классификация обращений.

Для решения задач классификации планируется использовать архитектуру TF-IDF и библиотеку SciKit-Learn (Sklearn).

Для классификации обращений использовались следующие классы:

Классификатор категорий:

- благоустройство территории;
- капитальный ремонт;
- газоснабжение;
- отопление;
- работа системы канализации;
- электроснабжение;
- подача холодного и горячего водоснабжения;
- работа лифтового оборудования.

Классификатор исполнителей:

- администрация;
- аварийно-диспетчерская служба;
- водоканал;
- госжилинспекция;
- управление капитального ремонта многоквартирных домов;
- управление благоустройства города;
- управление энергетики и жилищно-коммунального хозяйства.

В качестве основного метода классификации текстов использовался метод опорных векторов [5].

Для преобразования категорий и исполнителей в понятные для модели числовые данные используется класс LabelEncoder, который находится в библиотеке sklearn. Чтобы получить признак для первого столбца, необходимо обработать колонку функцией fit_transform и заменить существующие текстовые данные новыми закодированными [6]. LabelEncoder присваивает уникальное числовое значение каждой уникальной категории в столбце с метками, что позволяет моделям машинного обучения работать с категориальными данными. Он помогает преобразовать категориальные метки в числовые представления, что облегчает обработку данных алгоритмами машинного обучения. Процесс работы LabelEncoder заключается в следующем: каждой уникальной категории присваивается уникальное целое число.

После применения LabelEncoder к колонкам Категория и Исполнитель в датасете преобразованные данные сохраняются в новые файлы в формате pickle.

Итогом всех описанных действий является сохранение нового датасета, где категория и исполнитель представлены в виде чисел, а текст обращения нормализован.

Для обучения используем новый преобразованный датасет.

Для преобразования текста в числа, чтобы подать на обучение, используется TfidfVectorizer, который преобразует текст в вектора, где каждое слово представляется в виде числа в зависимости от важности слова для документа. Формула TF-IDF комбинирует понятия TF и IDF, чтобы вычислить важность каждого слова в каждом документе. Формально формула выглядит следующим образом [7]:

$$TF - IDF(t, d) = TF(t, d) * IDF(t),$$

где $TF(t, d)$ – частота термина (TF) для слова "t" в документе "d";

$IDF(t)$ – обратная частота документа (IDF) для слова "t".

Вес слова зависит от частоты появления в документе, т. е. чем чаще встречается, тем меньше вес и значимость. TfidfVectorizer позволяет выделить ключевые слова (токены), которые часто встречаются в документе.

После загрузки данных необходимо инициализировать и обучить векторайзер с сохранением его в новый файл формата pkl. После обучения векторайзера можно использовать его для преобразования новых текстовых данных в числовые вектора.

Обучение модели основано на алгоритме линейного метода опорных векторов LinearSVC [8]. После обучения модели сохраняются в файлы pkl, чтобы можно было использовать в дальнейшем, без повторного обучения, для этого используется библиотека pickle. На основе тестовых обученных данных происходит классификация новых обращений.

Общая схема функционирования модели представлена на рисунке 3.

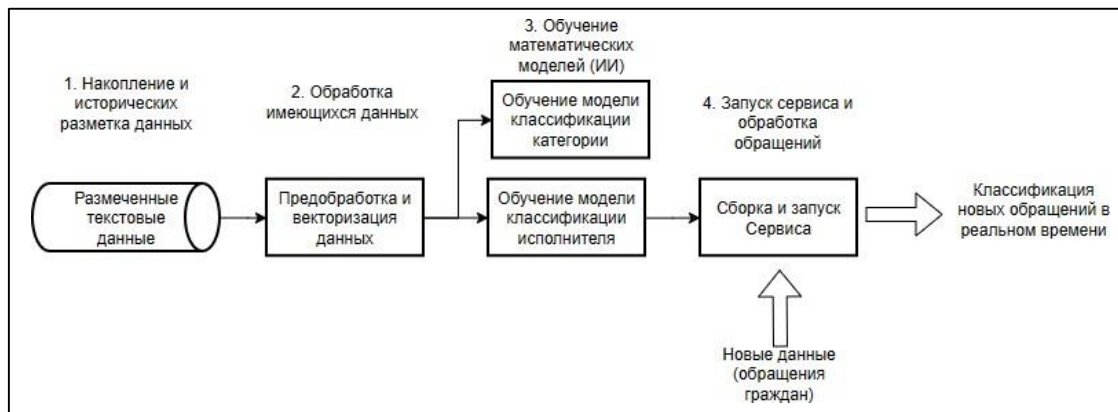


Рисунок 3 – Общая схема функционирования решения

Рассчитаем метрики для модели LinearSVC классификации по категориям.

Для этого разделим набор данных на обучающий и тестовый. Обучим модели LinearSVC и предскажем классы для тестового набора (рис. 4).

```
# Инициализация объектов
clf_category = LinearSVC()
clf_executor = LinearSVC()
clf_theme = LinearSVC()

# Обучение модели
clf_category.fit(X_train, Y_category_train)
clf_executor.fit(X_train, Y_executor_train)
clf_theme.fit(X_train, Y_theme_train)

y_pred = clf_category.predict(X_test)

precision = precision_score(Y_category_test, y_pred)
recall = recall_score(Y_category_test, y_pred)
f1 = f1_score(Y_category_test, y_pred)

print(f'Precision {precision}')
print(f'Recall {recall}')
print(f'f1 Score {f1}')
```

Рисунок 4 – Пример кода, демонстрирующий расчет метрик для модели LinearSVC

Были выбраны три метрики: precision (точность), recall (полнота), F-мера (среднее гармоническое precision и recall) (рис. 5).

```
Precision 0.7624309392265194
Recall 0.7624309392265194
f1 Score 0.7624309392265195
```

Рисунок 5 – Метрики

Для визуального представления предсказания классификатором вероятности категории построили кривую ROC для многоклассовой модели, для примера выбрали три класса.

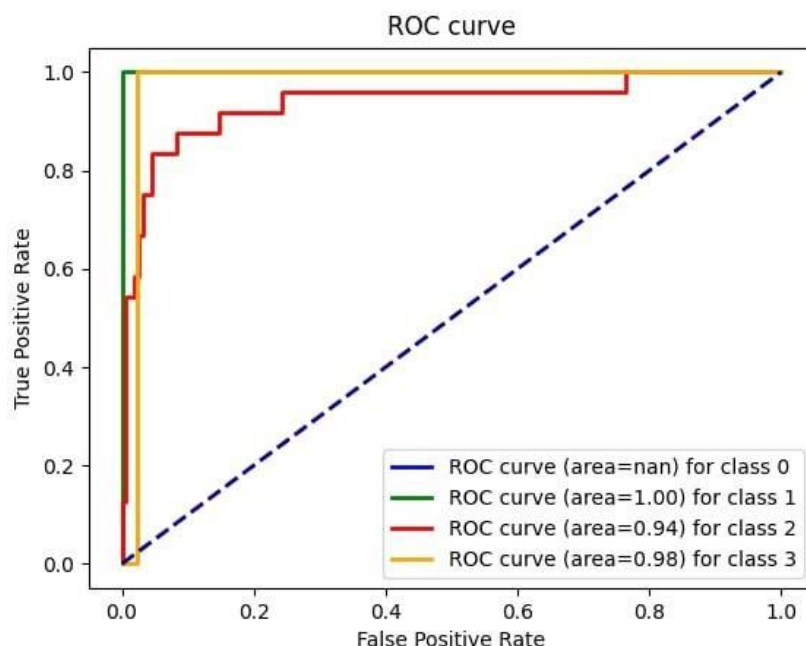


Рисунок 6 – Кривая ROC

Чем ближе кривая находится к верхнему углу графика, тем лучше производительность модели. Из рисунка 6 можно сделать вывод, что классификатор определяет категорию с высокой вероятностью верного результата и имеет низкую частоту ложных результатов.

Продемонстрируем работу автоматического классификатора. Набор параметров содержит следующие данные:

$O = \{\text{"На пересечении улицы Мира и Савушкина 23 не работает светофор"}\}$.

Текст обращения φ будет присвоен параметру *prediction_data* и отправлен на сервер GET-запросом: `127.0.0.1 "GET classify?prediction_data=["На пересечении улицы Мира и Савушкина 23 не работает светофор"]`. В результате сервер возвратил: *Категория: Дороги
 Исполнитель: Управление благоустройства города*. Результат соответствует действительности, следовательно, можно сделать выводы, что автоматический классификатор работает корректно. На рисунке 7 представлен фрагмент отображения результатов классификации в графическом виде.

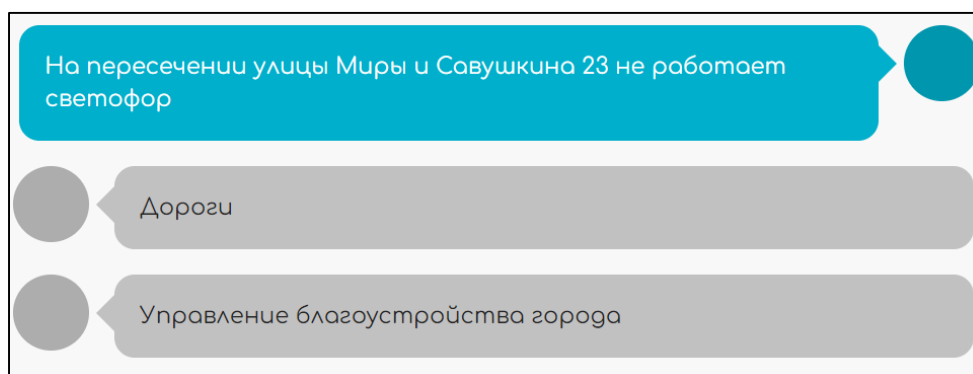


Рисунок 7 – Демонстрация работы классификатора

ЗАКЛЮЧЕНИЕ

В ходе исследования были рассмотрены существующие алгоритмы классификации данных. Разработана модель обработки обращений граждан по категориям и исполнителям. Модель может быть использована при разработке информационной системы, которая позволит ускорить и упростить обработку обращений от граждан, существенно повысить эффективность и оперативность реагирования на возможные аварийные ситуации. Модель может обрабатывать данные из датасетов, используемых в различных регионах.

Список источников

1. Евдошенко, О. И. Архитектура геоинформационной системы мониторинга аварийных ситуаций / О. И. Евдошенко, Н. И. Мавлюдова // Потенциал интеллектуально одаренной молодежи – развитию науки и образования : Международный научный форум молодых ученых, студентов и школьников. – Астрахань : Астраханский государственный архитектурно-строительный университет, 2022. – С. 101–104.
2. Низамитдинов, А. И. Алгоритмы машинного обучения для классификации текста / А. И. Низамитдинов, Б. Б. Иномов // Вестник ПИТТУ имени академика М.С. Осими. – 2020. – № 1 (14). – С. 27–35.
3. Камаева, А. А. Современное состояние искусственных нейронных сетей / А. А. Камаева // Инновации. Наука. Образование. – 2020. – № 16. – С. 377–387.
4. Быков, К. В. Особенности предобработки данных для применения машинного обучения / К. В. Быков // Молодой ученый. – 2021. – № 53 (395). – С. 1–4.
5. NLTK: Natural Language Toolkit // nltk.org. – URL: <https://www.nltk.org/>.
6. Шарутин, К. Н. Метод опорных векторов для категоризации текста / К. Н. Шарутин // Инновационные научные исследования. – 2021. – № 9–2 (11). – С. 49–61.
7. Machine Learning in Python // scikit-learn.org. – URL: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html> (дата обращения: 28.03.2024).
8. Machine Learning in Python // scikit-learn.org. – URL: https://scikit-learn.org/stable/modules/feature_extraction.html (дата обращения: 28.03.2024).
9. Machine Learning in Python // scikit-learn.org. – URL: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html> (дата обращения: 28.03.2024).

References

1. Evdoshenko, O. I., Mavlyudova, N. I. Architecture of the geoinformation system for emergency monitoring. *The potential of intellectually gifted youth for the development of science and education : International Scientific Forum of Young Scientists, Students and Schoolchildren*. Astrakhan, Astrakhan State University of Architecture and Civil Engineering, 2022, pp. 101–104 (In Russ.).
2. Nizametdinov, A. I., Inomov, B. B. Machine learning algorithms for text classification. *Bulletin of the PITTU named after academician M.S. Osimi*, 2020, no. 1 (14), pp. 27–35 (In Russ.).
3. Kamaeva, A. A. The current state of artificial neural networks. *Innovation. Science. Education*, 2020, no. 16, pp. 377–387 (In Russ.).
4. Bykov, K. V. Features of data preprocessing for the application of machine learning. *Young Scientist*, 2021, no. 53 (395), pp. 1–4 (In Russ.).
5. NLTK: Natural Language Toolkit. *nltk.org*. URL: <https://www.nltk.org/>.
6. Shrutin, K. N. The method of support vectors for text categorization. *Innovative scientific research*, 2021, no. 9–2 (11), pp. 49–61.
7. Machine Learning in Python. *scikit-learn.org*. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html> (accessed 28.03.2024).
8. Machine Learning in Python. *scikit-learn.org*. URL: https://scikit-learn.org/stable/modules/feature_extraction.html (accessed 28.03.2024).
9. Machine Learning in Python. *scikit-learn.org*. URL: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html> (accessed 28.03.2024).

Статья поступила в редакцию 03.05.2024; одобрена после рецензирования 20.05.2024; принята к публикации 21.05.2024.

The article was submitted 03.05.2024; approved after reviewing 20.05.2024; accepted for publication 21.05.2024.

УДК 004.4

**ИНФОРМАЦИОННАЯ ПОДДЕРЖКА ПРОЦЕССА ОКАЗАНИЯ
БЛАГОТВОРИТЕЛЬНОЙ ПОМОЩИ**

Лебедева Кристина Евгеньевна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
бакалавр, ORCID: 0009-0001-8678-3397, e-mail: klebedeva14052003@gmail.ru

Азмухамедов Искандар Маратович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
доктор технических наук, декан факультета цифровых технологий и кибербезопасности, профессор кафедры информационной безопасности, ORCID: 0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

В статье проанализированы проблемы, возникающие при осуществлении благотворительной деятельности. Для повышения эффективности взаимодействия между волонтерами и людьми, нуждающимися в помощи, предложено разработать информационную систему, позволяющую облегчить коммуникацию между указанными группами лиц. В основу процесса подбора наиболее подходящей пары «волонтер – человек, нуждающийся в помощи» положено нахождение хэммингово расстояния между параметрами, которые участники процесса указали в качестве значимых. Разработаны проектные диаграммы, и приведено описание структуры и предполагаемых интерфейсов соответствующего программного обеспечения. Программное обеспечение будет представлять собой клиент-серверное приложение. Клиентскую часть предложено реализовать в виде мобильного приложения, работающего на устройствах под управлением операционной системы Android. В качестве системы управления базами данных предложено использовать PostgreSQL.

Ключевые слова: благотворительная помощь, волонтер, человек нуждающийся в помощи (ЧНП), информационная система, расстояние Хэмминга

INFORMATION SUPPORT FOR THE PROCESS OF PROVIDING CHARITABLE ASSISTANCE

Lebedeva Kristina E., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

bachelor, ORCID: 0009-0001-8678-3397, e-mail: klebedeva14052003@gmail.ru

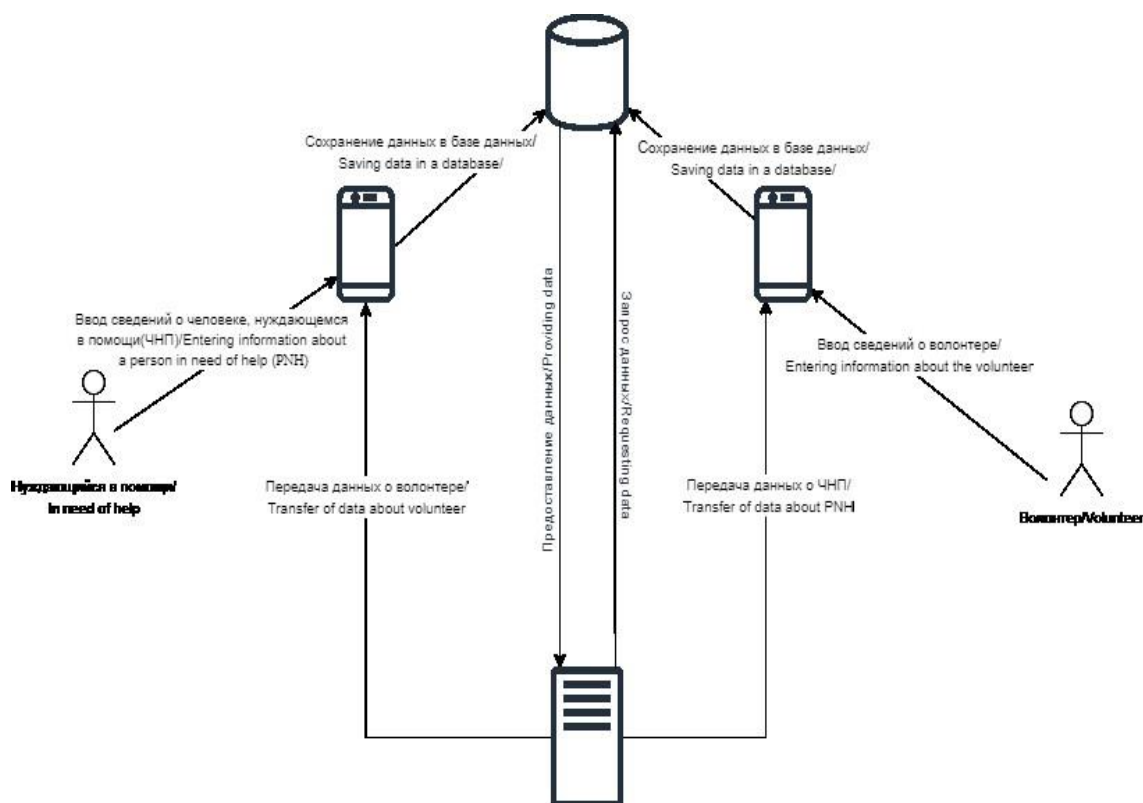
Azhmukhamedov Iskandar M., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

Doct. Sci. (Engineering), Dean of the Faculty of Digital Technologies and Cybersecurity, Professor of the Department of Information Security and Digital Technologies, ORCID: 0000-0001-9058-123X, e-mail: aim_agtu@mail.ru

The article analyzes the problems that arise in the implementation of charitable activities. To increase the effectiveness of interaction between volunteers and people in need of help, it is proposed to develop an information system that facilitates communication between these groups of people. The process of selecting the most suitable pair of "volunteer – person in need of help" is based on finding the Hamming distance between the parameters that the participants in the process indicated as significant. Design diagrams have been developed and a description of the structure and intended interfaces of the corresponding software is provided. The software will be a client-server application. It is proposed to implement the client part in the form of a mobile application running on devices running the Android operating system. It is proposed to use PostgreSQL as a database management system.

Keywords: charity aid, volunteer, person in need of help (PNH), information system, Hamming distance

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

Оказание благотворительной помощи имеет глубокие исторические корни [1, 2]. В настоящее время подобная деятельность в РФ юридически регламентируется Федеральным законом «О благотворительной деятельности и добровольчестве (волонтерстве)», в котором устанавливаются основы правового регулирования благотворительной и добровольческой (волонтерской) деятельности, определяются возможные формы ее поддержки органами государственной власти и органами местного самоуправления, особенности создания и деятельности благотворительных организаций в целях широкого распространения и развития благотворительной и добровольческой (волонтерской) деятельности в Российской Федерации [3].

Изменения, происходящие в социальных, экономических сферах, сказываются на увеличении проблем в обществе. В связи с этим количество нуждающихся в благотворительной помощи людей не уменьшается. Это могут быть люди, попавшие в сложные жизненные ситуации из-за потери работы, болезни или других непредвиденных обстоятельств. Также существуют категории людей, которые постоянно нуждаются в помощи из-за возраста, социального положения или нетрудоспособности.

Для таких категорий людей благотворительность может выражаться как в помощи приобретения вещей первой необходимости (например, лекарств, продуктов питания, одежды и т. п.), так и в виде оказания различных услуг (уборка квартиры, сопровождение человека в больницу, психологическая помощь и т. п.).

Необходимо отметить, что существует большое количество людей, готовых оказать такого рода помощь. Однако желательно, чтобы данная помощь оказывалась адресно, с учетом всех обстоятельств и личностных характеристик нуждающихся в такой помощи людей. Таким образом, с одной стороны, есть люди, нуждающиеся в помощи, с другой стороны, есть люди, готовые оказать эту помощь.

При этом большинство благотворительных фондов, занимающихся данной проблемой, работают на макроуровне: они собирают и аккумулируют средства, поступающие от людей на благотворительность, и направляют их для оказания помощи нуждающимся. При такой организации жертвующий на благотворительность человек не является активным участником процесса.

Для того чтобы привлечь больше заинтересованных граждан к оказанию помощи нуждающимся, целесообразно организовать этот процесс по принципу P2P (от человека к человеку), чтобы каждый мог непосредственно участвовать в помощи.

Это, в свою очередь, делает актуальной разработку информационной системы поддержки данного процесса.

ПОСТАНОВКА ЗАДАЧИ

Исходя из этого, целью данной работы стало описание процесса подбора наиболее подходящих пар «волонтер – человек, нуждающийся в помощи» с учетом их индивидуальных особенностей, а также описание структуры и предполагаемых интерфейсов соответствующего программного обеспечения.

МЕТОДИКА ПОДБОРА ВОЛОНТЕРА ДЛЯ ЧЕЛОВЕКА, НУЖДАЮЩЕГОСЯ В ПОМОЩИ

Каждый участник процесса оказания благотворительной помощи имеет набор характеризующих его параметров.

Так человек, готовый оказать помощь, – волонтер (V), характеризуется следующими параметрами:

$$V^i = \langle L^i; M^i \rangle, \quad (1)$$

где L^i – личные характеристики i -го волонтера;

M^i – характеристики нуждающегося в помощи человека (ЧНП), при которых i -й волонтер готов и может оказать ему помощь.

Личные характеристики волонтера, в свою очередь, включают в себя следующие параметры:

$$L^i = \langle l_1; l_2; l_3; l_4; l_5 \rangle, \quad (2)$$

где l_1 – адрес проживания волонтера;

l_2 – возраст волонтера;

l_3 – пол волонтера (женский, мужской);

l_4 – национальность волонтера (необходима для преодоления культурно-языкового барьера между волонтером и ЧНП);

l_5 – мобильный номер телефона волонтера.

Характеристики нуждающегося в помощи человека, при которых волонтер готов и может оказать ему помощь:

$$M^i = \langle m_1; m_2; m_3; m_4; m_5; m_6; m_7 \rangle, \quad (3)$$

где m_1 – район города, где волонтер готов оказать помощь нуждающемуся;

m_2 – диапазон возраста человека, которому необходима помощь (0–18, 18–30, 31–60, 60 и выше, любой возраст);

m_3 – пол человека, которому необходимо оказать услугу (женский, мужской, любой пол);

m_4 – национальность ЧНП;

m_5 – виды помощи (убраться в квартире, отвезти на медицинские процедуры, оказать психологическую помощь и т. п.);

m_6 – длительность (периодичность) оказания услуги (1 день, 7 дней в неделю, каждый месяц и т. п.);

m_7 – время оказания услуги (7:00–12:00, 12:00–18:00, 18:00–22:00, 24 часа в сутки).

Человека, нуждающегося в помощи (H), можно описать следующим кортежем:

$$H^j = \langle P^j; N^j \rangle, \quad (4)$$

где P^j – личные характеристики j -го ЧНП;

N^j – характеристики волонтера, при которых j -й нуждающийся в помощи человек готов принять помощь.

К личным характеристикам человека, которому необходима помощь относятся:

$$P^j = \langle p_1; p_2; p_3; p_4; p_5; p_6; p_7; p_8 \rangle, \quad (5)$$

где p_1 – адрес проживания ЧНП;

p_2 – возраст человека, нуждающегося в помощи;

p_3 – пол ЧНП (женский, мужской);

p_4 – национальность человека, которому необходима помощь (необходима для преодоления культурно-языкового барьера между ЧНП и волонтером);

p_5 – мобильный номер телефона ЧНП;

p_6 – помощь, в которой нуждается ЧНП (принести продукты питания, отвезти в больницу, проконсультировать по различным вопросам и т. п.);

p_7 – длительность (периодичность) оказания услуги (1 день, 7 дней в неделю, каждый месяц и т. п.);

p_8 – время, когда необходимо оказывать помощь (7:00–12:00, 12:00–18:00, 18:00–22:00, 24 часа в сутки).

Характеристики волонтера, при которых ЧНП готов принять помощь:

$$N^j = \langle n_1; n_2; n_3 \rangle, \quad (6)$$

где n_1 – диапазон возраста человека (14–18, 18–30, 31–60, 60 и выше, любой возраст);

n_2 – пол человека (женский, мужской);

n_3 – национальность волонтера.

Исходя из этого, основной задачей разработки программного обеспечения является нахождение оптимального соответствия между волонтером (V) и нуждающимся в помощи человеком (H):

$$V \leftrightarrow H$$

оптим

Для нахождения такого соответствия необходимо найти минимальное хэммингово расстояние между элементами потребностей человека, нуждающегося в помощи и возможностями волонтера:

$$d_H(V_i, H_j) = \text{количество несовпадающих пар соответствующих элементов во множествах } (V_i^*, H_j^*), \quad (7)$$

где $V_i^* \subset V_i, H_j^* \subset H_j$ – множество значимых параметров, влияющих на нахождение соответствия.

При этом можно учесть различную значимость несовпадающих параметров, приписав им соответствующие веса. Процедура нахождения указанных весов заключается в следующем:

1. Каждый из участников процесса (волонтер и ЧНП) ранжируют параметры по степени возрастания значимости. При этом на одном и том же месте могут оказаться несколько параметров, если их значимость для актора одинакова.

2. По результатам нестрогого ранжирования вычисляются так называемые веса Фишберна [4].

3. При нахождении оптимального соответствия в качестве веса выбирается максимальное значение из весов, полученных для волонтера и ЧНП.

Расчетный пример. Допустим, при выборе оптимальной пары необходимо учесть 5 значимых параметров – $k_1; k_2; k_3; k_4; k_5$. Пусть волонтер ранжировал их по степени возрастания в следующем порядке: $k_2; (k_4; k_5); k_3; k_1$. Веса Фишберна для волонтера будут определены на основе методики, представленной в [4], следующим образом:

k_2	k_4	k_5	k_3	k_1
1	2	2	3	4

Сумма мест = 12.

Исходя из этого, для волонтера веса параметров будут следующие:

k_1	k_2	k_3	k_4	k_5
4/12	1/12	3/12	2/12	2/12

При этом сумма весов равна единице:

$$\frac{4}{12} + \frac{1}{12} + \frac{3}{12} + \frac{2}{12} + \frac{2}{12} = 1.$$

Допустим, человек, нуждающийся в помощи, проделал то же самое и получил свой ранжированный список: $k_4; (k_2; k_3); (k_1; k_5)$.

Для ЧНП, проделав аналогичную описанную выше процедуру, получим таблицу:

k_1	k_2	k_3	k_4	k_5
3/11	2/11	2/11	1/11	3/11

При этом сумма весов также равна единице:

$$\frac{3}{11} + \frac{2}{11} + \frac{2}{11} + \frac{1}{11} + \frac{3}{11} = 1$$

Окончательно при определении оптимальной пары к несовпадающим параметрам будут применены следующие веса:

	p_1	p_2	p_3	p_4	p_5
V	4/12	1/12	3/12	2/12	2/12
H	3/11	2/11	2/11	1/11	3/11
max (V; H)	4/12	2/11	3/12	2/12	3/11

Общий алгоритм работы программы, реализующий методику подбора наиболее подходящей пары «Волонтер – ЧНП» будет выглядеть следующим образом:

1. Люди, которые хотят оказывать благотворительную помощь, регистрируются на платформе и указывают о себе соответствующие сведения, входящие в кортеж 2, 3.

2. Данные сведения сохраняются в базе данных в таблице «Волонтеры».

3. Люди, нуждающиеся в помощи, также регистрируются в приложении и указывают сведения о себе и своих потребностях, входящие в кортеж 5, 6. Данные сведения также сохраняются в базе данных.

4. При поступлении запроса от человека, нуждающегося в помощи, система генерирует ранжированный список наиболее подходящих волонтеров для удовлетворения той или иной потребности ЧНП.

5. После генерации списка предложение об оказании помощи конкретному ЧНП направляется первому, наиболее подходящему в списке волонтеру. Данные указываются на основе вычислений хэммингово расстояния с учетом найденных весов Фишберна.

6. Если выбранный волонтер согласен оказать данный вид благотворительной помощи, то система направляет уведомление ЧНП, в котором, в частности, указывается мобильный телефон волонтера. Если в процессе общения между волонтером и человеком, нуждающимся в помощи, выясняется, что по какой-либо причине данный волонтер не может выполнить просьбу ЧНП или, наоборот, человек, нуждающийся в помощи не согласен на выполнение его запроса предложенным волонтером, то человек, которому необходима помощь просит систему назначить другого волонтера. В этом случае запрос будет направлен следующему в списке наиболее подходящих волонтеров. В случае согласия все контактные данные волонтера передаются ЧНП. Данная процедура будет повторяться до тех пор, пока не будет выбран подходящий волонтер.

7. После выполнения запроса ЧНП ставит отметку о его выполнении. Данная отметка появляется на странице волонтера в списке указанных услуг. Также человек, которому оказали помощь, может оставить отзыв об оказанной услуге и оценить по пятибалльной шкале работу волонтера. В свою очередь, данная функция будет доступна и волонтеру. Он сможет оценить человека, нуждающегося в помощи, и оставить о нем отзыв.

РАЗРАБОТКА ОСНОВНЫХ ПРОЕКТНЫХ ДИАГРАММ

Для эффективного взаимодействия волонтеров и ЧНП было принято решение автоматизировать разработанную выше методику.

Для более детального понимания общего алгоритма работы программного обеспечения рассмотрим проектные диаграммы в нотации UML [5, 6]. Диаграмма, которая показывает, какие функции программного продукта доступны различным группам пользователей, представляет собой use-case диаграмму. Для разрабатываемого ПО она представлена на рисунке 1.

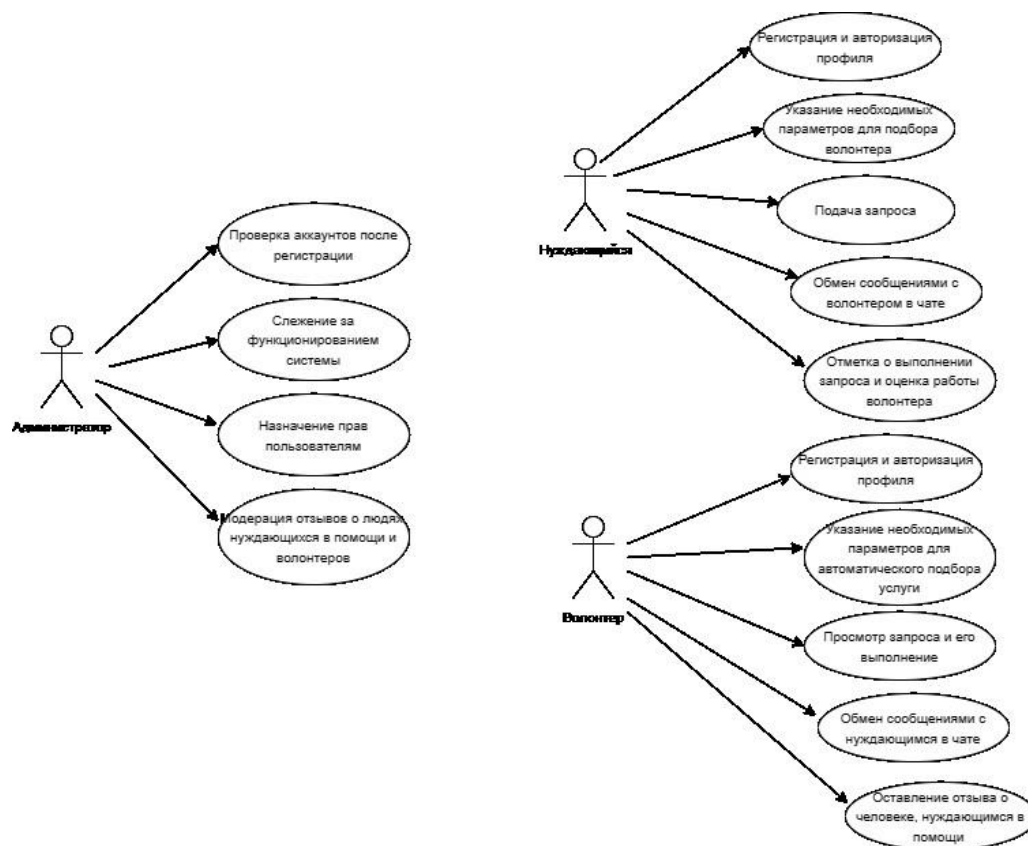


Рисунок 1 – Диаграмма вариантов использования

В разрабатываемой системе предусмотрено 3 вида пользователей:

1. Администратор.
2. Нуждающийся – человек, которому необходимо оказать помощь.
3. Волонтер – человек, который готов оказать помощь.

Администратор осуществляет контроль над системой: следит за функционированием системы, назначает определенные права пользователям, то есть то, к чему они имеют доступ в системе, проверяет аккаунты после регистрации на правильность заполнения полей и одобряет регистрацию аккаунта или отправляет на доработку. Также он проводит модерацию отзывов со стороны волонтеров и ЧНП.

Зарегистрированный администратором нуждающийся после того, как пройдет регистрацию или авторизацию профиля, должен указать необходимые параметры для последующего подбора волонтеров, после чего имеет право подавать запросы, общаться в чате с волонтерами, которые оказывают ему помощь, и после выполнения волонтером запроса ставить отметку о выполнении и оценивать выполненную работу.

Есть еще один вид пользователя – волонтер, который после прохождения регистрации или авторизации, заходит на страничку, заполняет в аккаунте информацию о необходимых параметрах для подбора услуги, также он может видеть актуальные запросы, которые ему предлагает выполнить система. После того как волонтер выбрал подходящий запрос, который он сможет выполнить исходя из своих предпочтений и возможностей, он сможет перейти в чат и пообщаться с человеком, который создал этот запрос и задать необходимые ему вопросы. Также волонтер имеет право оставить отзыв. Пользователь (нуждающийся или волонтер) может работать в системе только после того, как администратор подтвердит его учетную запись в системе.

Для понимания процесса создания запроса и выбора волонтера для конкретного запроса необходимо воспользоваться диаграммой состояний (рис. 2), которая показывает, как объект переходит из одного состояния в другое.

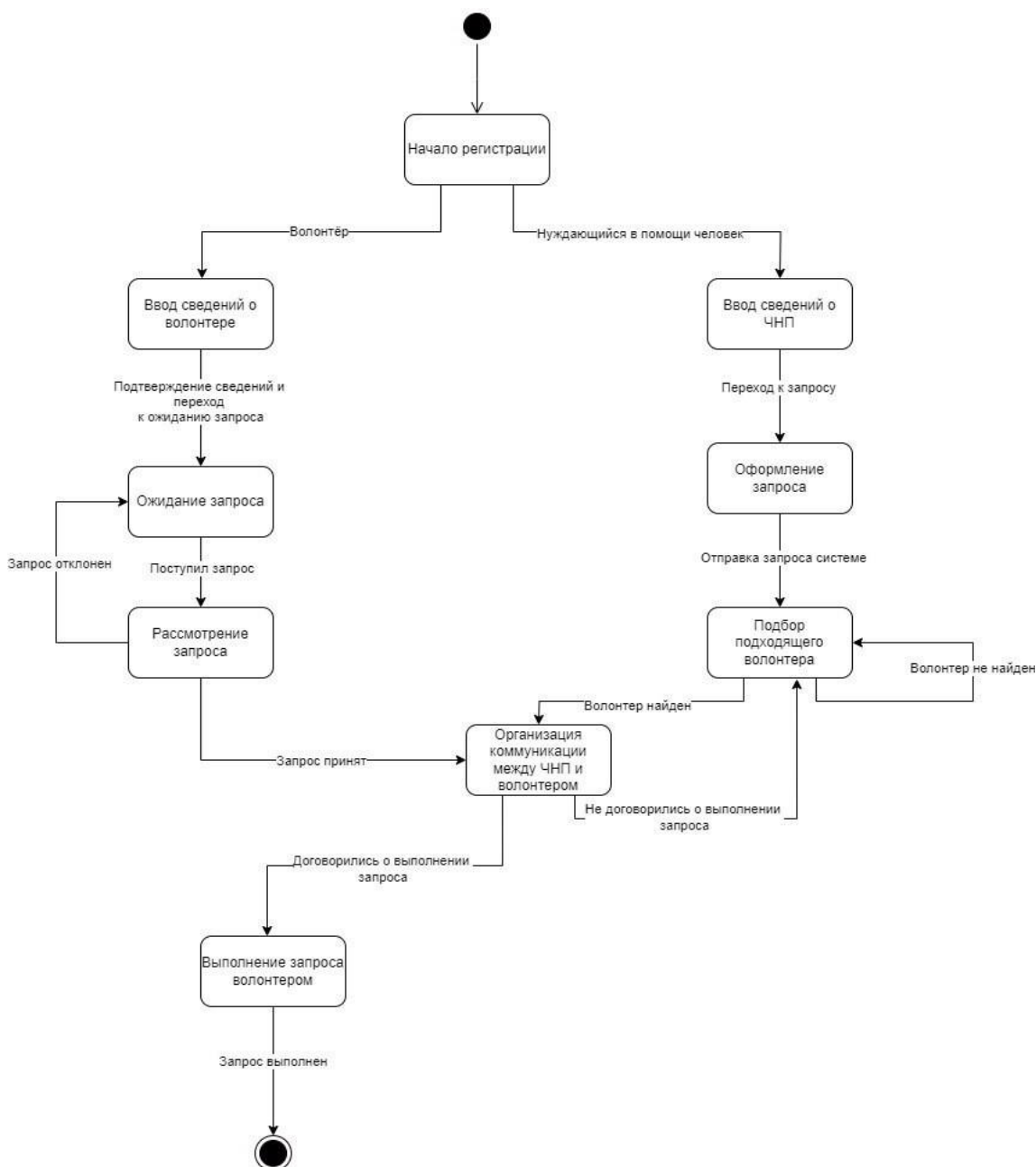


Рисунок 2 – Диаграмма состояний

ВЫБОР СРЕДЫ РАЗРАБОТКИ

Клиентская часть представлена в виде мобильного приложения, так как прирост мобильной аудитории превышает общий прирост интернет-аудитории [7]. Основной платформой мобильного приложения будет операционная система (ОС) Android. Выбор был сделан в пользу данной операционной системы, поскольку, согласно статистике [8], около 70 % людей используют мобильные телефоны на базе ОС Android, в связи с чем приложение будет доступно для большего числа пользователей. Также основными преимуществами при выборе ОС служили:

- наиболее долгий срок поддержки приложений на любую версию ОС Android [9];
- открытость платформы;
- гибкая настройка интерфейса и функциональности приложения;
- широкий спектр интеграции с другими устройствами и сервисами.

Исходный код программ будет написан на языке программирования Java. Преимущества данного языка программирования в том, что он имеет обширную библиотеку классов, которая предоставляет множество инструментов и функций для создания мобильного приложения [10].

В качестве хранилища данных была выбрана свободная объектно-реляционная система управления базами данных PostgreSQL. Основными преимуществами выбора СУБД служили [11]:

- поддержка базы данных неограниченного размера;
- богатый набор встроенных и пользовательских типов данных, включая географические массивы, JSON и XML;
- надёжные механизмы транзакций и репликации;
- эффективная система управления многозадачностью, что позволяет обслуживать большое количество одновременных запросов.

ПРИМЕРЫ ПРЕДПОЛАГАЕМЫХ ИНТЕРФЕЙСОВ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Для использования приложения пользователю необходимо войти в систему и авторизоваться (рис. 3), в случае если пользователь не зарегистрирован, то необходимо будет пройти регистрацию.

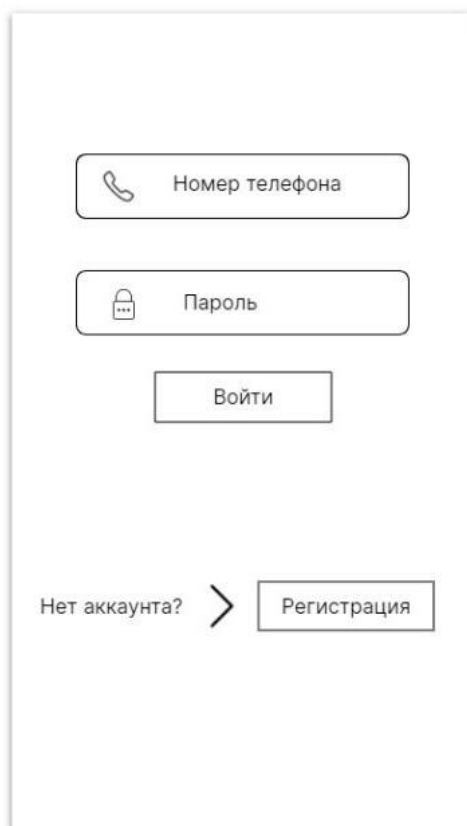


Рисунок 3 – Окно авторизации пользователя

При регистрации ему будет изначально предложено выбрать роль – нуждающийся или волонтер (рис. 4). Созданный аккаунт будет относиться к одной роли.

Выберите свою роль

Волонтер

Человек
нуждающийся в помощи

Продолжить

Рисунок 4 – Окно выбора роли

Если человек захочет выбрать другую роль, ему будет необходимо создать второй аккаунт.

Если человек выбрал роль волонтера, то ему необходимо будет заполнить соответствующие поля при регистрации (рис. 5). Все поля обязательны для заполнения.

Регистрация
На данном поле вы заполняете свои личные данные

* Укажите ваше ФИО:

* Место проживания (район, улица):

* Ваш пол:

* Национальность:

* Мобильный номер телефона:

Вы даете свое согласие на обработку персональных данных?

Если Вам меньше 18 лет, то необходимо заполнить согласие от ваших родителей/опекунов

Скачать

Загрузить файл

Далее

Регистрация
На данном поле вы заполняете характеристики нуждающегося в помощи человека (ЧНП), при которых вы готовы и можете оказать ему помощь. Рядом с характеристикой укажите от 0 до 3 степень важности критерия для Вас.

* Места проживания людей, где вам будет удобно оказывать им помощь:

* Возраст:

* Пол:

* Национальность:

* Виды помощи, которые вы готовы оказывать:

* Длительность (периодичность) оказания услуги:

* Время оказания услуги:

Завершить регистрацию

Рисунок 5 – Окна регистрации волонтера

При выборе роли «нуждающегося» человек заполняет другие поля (рис. 6). Все поля также обязательны для заполнения.

The image shows two side-by-side registration forms titled "Регистрация".

Left Form (Personal Data):

- Header: "Регистрация"
- Text: "На данном поле вы заполняете свои личные данные"
- Field: "* Укажите ваше ФИО:" (text input)
- Field: "* Место проживания (район, улица):" (text input)
- Field: "* Ваш пол:" (dropdown menu)
- Field: "* Национальность:" (text input)
- Field: "* Мобильный номер телефона:" (text input)
- Checkbox: "Вы даете свое согласие на обработку персональных данных?" (unchecked)
- Text: "Если Вам меньше 18 лет, то необходимо заполнить согласие от ваших родителей/опекунов"
- Buttons: "Скачать" (with cloud icon) and "Загрузить файл" (with file icon)
- Bottom Button: "Далее"

Right Form (Volunteer Characteristics):

- Header: "Регистрация"
- Text: "На данном поле вы заполняете характеристики волонтера, при которых вы готовы принять от него помощь. Рядом с характеристикой укажите от 0 до 3 степень важности критерия для Вас."
- Field: "* Возраст:" (dropdown menu with radio button)
- Field: "* Пол:" (dropdown menu with radio button)
- Field: "* Национальность:" (dropdown menu with radio button)
- Bottom Button: "Завершить регистрацию"

Рисунок 6 – Окна регистрации человека, нуждающегося в помощи

Как только пользователь заполнил все поля и нажал кнопку «завершить регистрацию», аккаунт отправляется администратору на проверку правильности заполнения полей.

Для заполнения заявки и правильного подбора волонтера ЧНП заполняет сведения о своем запросе. Окно создания заявки представлено на рисунке 7.

The image shows a form titled "Заявка на оказание услуги".

- Header: "Заявка на оказание услуги"
- Text: "Помощь, которую вам необходимо оказать:"
- Text: "* пишите кратко (например: отвезти в больницу, принести продукты и т.п.)"
- Field: (text input for help description)
- Text: "Длительность (периодичность) предоставления услуги:"
- Field: (dropdown menu for duration)
- Text: "Время, когда необходимо оказывать Вам помощь:"
- Field: (dropdown menu for time)
- Text: "После заполнения полей, нажмите кнопку "создать заявку", а после кнопку "сгенерировать подходящего волонтера"
- Buttons: "Создать заявку" and "Сгенерировать подходящего волонтера" (with a downward arrow between them)

Рисунок 7 – Окно создания заявки

ЗАКЛЮЧЕНИЕ

Проведенный анализ процесса взаимодействия людей, нуждающихся в помощи, и волонтеров позволил выявить проблему, заключающуюся в отсутствии информационной системы, которая могла бы эффективно взаимодействовать указанным группам лиц.

Решением данной проблемы является разработка мобильного приложения для оказания благотворительной помощи на безвозмездной основе, охватывающего различные слои населения. Данная система предоставит возможность подачи заявок на оказание помощи и будет обеспечивать автоматический подбор волонтеров для конкретной заявки. Это позволит оптимизировать процесс взаимодействия волонтеров и нуждающихся.

В итоге разработанная информационная система, доступная через мобильное приложение для операционной системы Android, будет содействовать улучшению процесса благотворительной помощи, обеспечивая своевременную и адекватную поддержку нуждающихся, а также оптимизируя работу волонтеров, что, в свою очередь, повысит эффективность всей благотворительной деятельности.

Список источников

1. Гривастова, А. Н. Благотворительность в дореволюционной России: основные проблемы и итоги изучения / А. Н. Гривастова // Христианское чтение. – 2022. – № 2. – С. 335–348.
2. Кусов, А. С. О некоторых тенденциях трансформации института благотворительности в современной России / А. С. Кусов // Общество: социология, психология, педагогика. – 2021. – № 9 (89). – С. 86–89.
3. Федеральный закон «О благотворительной деятельности и добровольчестве (волонтерстве)» от 11.08.1995 № 135-ФЗ (последняя редакция). – URL: https://www.consultant.ru/document/cons_doc_LAW_7495/ (дата обращения: 01.04.2024).
4. Ажмухамедов, И. М. Анализ и управление комплексной безопасностью на основе когнитивного моделирования / И. М. Ажмухамедов // Управление большими системами : сборник трудов. – 2010. – № 29. – С. 5–15.
5. Леоненков, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose : учебное пособие / А. В. Леоненков. – 3-е изд. – Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. – 317 с. – ISBN 978-5-4497-0667-6.
6. Fowler, M. UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition / M. Fowler ; Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison Wesley. – 2004. – 179 p.
7. Статистика интернета и соцсетей на 2023 год – цифры и тренды в мире и в России // WebCanape. – URL: <https://www.web-canape.ru/business/statistika-interneta-i-socsetej-na-2023-god-cifry-i-trendy-v-mire-i-v-rossii/?ysclid=lv0oejfphc717351959> (дата обращения: 02.04.2024).
8. Каких пользователей больше: Android или iOS? // Uchet-jkh.ru. – URL: <https://uchet-jkh.ru/i/kakix-polzovatelei-bolse-android-ili-ios> (дата обращения: 02.04.2024).
9. Погорелов, Д. В. Сравнение мобильных операционных систем Android и iOS / Д. В. Погорелов, Е. А. Колоколов, В. В. Ермолаева // Вестник науки. – 2022. – № 12 (57). – С. 118–127.
10. Martinez, D. D. A Review on Java Programming Language / D. D. Martinez, A. H. Remegio, D. R. Lincopinis // ResearchGate. – URL: https://www.researchgate.net/publication/371166744_A_Review_on_Java_Programming_Language (дата обращения: 02.04.2024).
11. Wolff, D. A. MySQL, PostgreSQL, and PHP: open source technologies for a database management course / D. A. Wolff // ResearchGate. – URL: https://www.researchgate.net/publication/255673735_MySQL_PostgreSQL_and_PHP_open_source_technologies_for_a_database_management_course (дата обращения: 02.04.2024).

References

1. Grivastova, A. N. Charity in pre-revolutionary Russia: main problems and results of the study. *Christian Reading*, 2022, no. 2, pp. 335–348 (In Russ.).
2. Kusov, A. S. On some trends in the transformation of the institution of charity in modern Russia. *Society: Sociology, Psychology, Pedagogy*, 2021, no. 9 (89), pp. 86–89 (In Russ.).
3. Federal Law «On Charitable Activities and Volunteering (Volunteering)» dated August 11, 1995 No. 135-FZ (latest edition). URL: https://www.consultant.ru/document/cons_doc_LAW_7495/ (accessed 04.01.2024) (In Russ.).
4. Azhmukhamedov, I. M. Analysis and management of integrated security based on cognitive modeling. *Management of large systems : collection of works*, 2010, no. 29, pp. 5–15 (In Russ.).
5. Leonenkov, A.V. *Object-oriented analysis and design using UML and IBM Rational Rose : textbook*. 3rd ed. Moscow, Internet University of Information Technologies (INTUIT), IP Ar Media, 2020. 317 p. ISBN 978-5-4497-0667-6 (In Russ.).
6. Fowler, M. *UML Distilled: A Brief Guide to the Standard Object Modeling Language, Third Edition*. Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison Wesley, 2004. 179 p.
7. Internet and social network statistics for 2023 – numbers and trends in the world and in Russia. *WebCanape*. URL: <https://www.web-canape.ru/business/statistika-interneta-i-socsetej-na-2023-god-cifry-i-trendy-v-mire-i-v-rossii/?ysclid=lv0oejfphc717351959> (accessed 04.02.2024) (In Russ.).
8. Which users are there more: Android or iOS? *Uchet-jkh.ru*. URL: <https://uchet-jkh.ru/i/kakix-polzovatelei-bolse-android-ili-ios> (accessed 04.02.2024) (In Russ.).

9. Pogorelov, D. V., Kolokolov, E. A., Ermolaeva, V. V. Comparison of Android and iOS mobile operating systems. *Bulletin of Science*, 2022, no. 12 (57), pp. 118–127 (In Russ.).

10. Martinez, D. D., Remegio, A. H., Lincopinis, D. R. A Review on Java Programming Language. *ResearchGate*. URL: https://www.researchgate.net/publication/371166744_A_Review_on_Java_Programming_Language (accessed 04.02.2024).

11. Wolff, D. A. MySQL, PostgreSQL, and PHP: open source technologies for a database management course. *ResearchGate*. URL: https://www.researchgate.net/publication/255673735_MySQL_PostgreSQL_and_PHP_open_source_technologies_for_a_database_management_course (accessed 04.02.2024).

Статья поступила в редакцию 27.04.2024; одобрена после рецензирования 17.05.2024; принята к публикации 17.05.2024.

The article was submitted 27.04.2024; approved after reviewing 17.05.2024; accepted for publication 17.05.2024.

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ, ЧИСЛЕННЫЕ МЕТОДЫ И КОМПЛЕКСЫ ПРОГРАММ

УДК 004.001

ДИАГНОСТИКА ТУБЕРКУЛЕЗА БЕЗ БАКТЕРИОВЫДЕЛЕНИЯ МЕТОДАМИ СЛУЧАЙНОГО ЛЕСА И КОМИТЕТОВ С МИНИМИЗАЦИЕЙ ЧИСЛА ВХОДНЫХ ПРИЗНАКОВ

Тюлькова Татьяна Евгеньевна, Федеральное государственное бюджетное учреждение «Национальный медицинский исследовательский центр фтизиопульмонологии и инфекционных заболеваний» Министерства здравоохранения Российской Федерации, 127473, Российская Федерация, г. Москва, ул. Достоевского, 4,

доктор медицинских наук, руководитель отдела координации научных исследований, ORCID: 0000-0002-2292-1228, e-mail: tulkova2006@rambler.ru

Чернавин Павел Федорович, Уральский федеральный университет, 620002, Российская Федерация, г. Екатеринбург, ул. Мира, 19,

кандидат экономических наук, доцент, ORCID: 0000-0003-3214-3906, e-mail: chernavin.p.f@gmail.com

Чернавин Николай Павлович, Уральский федеральный университет, 620002, Российская Федерация, г. Екатеринбург, ул. Мира, 19,

ассистент кафедры, ORCID: 0000-0002-2093-9715, e-mail: ch_k@mail.ru

Статья исследует применение методов машинного обучения для диагностики туберкулеза у взрослых при отсутствии бактериовыделения с целью улучшить точность диагностирования и предотвратить ненужное применение антибиотиков. Диагностика туберкулеза представляет собой сложную задачу, особенно при отсутствии бактериовыделения. Иммунологические тесты тоже имеют определенную погрешность и зависят от человеческого фактора. Исследование проводилось на основе данных 740 пациентов в возрасте от 18 до 70 лет. Изначально рассматривались 139 входных признаков. После обработки с помощью классических методов определения информативности признаков и человеко-машинных процедур анализа данных, количество информативных признаков сокращалось в следующей последовательности 62, 29, 19. Отдельно исследовался вопрос выбора наиболее информативных признаков на основе авторского подхода к минимизации признаков на основе использования моделей линейного программирования с частично-булевыми переменными. В результате число признаков было снижено до 8. Статья представляет интерес как для профессиональных медицинских работников, особенно фтизиатров, так и специалистов по машинному обучению.

Ключевые слова: туберкулез без бактериовыделения, машинное обучение, метод комитетов, информативность признаков

DIAGNOSTICS BACTERIOLOGICALLY NON-CONFIRMED TUBERCULOSIS BY THE FORESTT METHOD AND THE COMMITTEE MACHINE METHOD WITH A DECREASE IN THE NUMBER OF INJECTED SIGNS

Tyulkova Tatiana Y., Federal StateBudgetary Institution “National Medical Research Center for Phthiisopulmology and Infectious Diseases” of the Ministry of Health of Russian Federation, 4 Dostoevsky St., Moscow, 127473, Russian Federation,

Doct. Sci. (Medicine), Head of Research Coordination Department, ORCID: 0000-0002-2292-1228, e-mail: tulkova2006@rambler.ru

Chernavin Pavel F., Ural Federal University, 19 Mira St., Yekaterinburg, 620002, Russian Federation,

Cand. Sci. (Economics), Associate Professor, ORCID: 0000-0003-3214-3906, e-mail: chernavin.p.f@gmail.com

Chernavin Nikolai P., Ural Federal University, 19 Mira St., Yekaterinburg, 620002, Russian Federation,

Assistant, ORCID: 0000-0002-2093-9715, e-mail: ch_k@mail.ru

The article explores the application of machine learning methods for diagnosing bacteriologically non-confirmed tuberculosis in adults, with the aim of to improve the accuracy of the diagnosis of exclusion of drug. Diagnosis of tuberculosis is a difficult task, especially with unconfirmed bacterial excretion. Immunological tests also have a certain error and depend on the human factor. The study was conducted on the basis of data from 740 patients aged 18 to 70 years. Initially, 139 input features were considered. After processing using classical methods for determining the information content of features and human-machine data analysis procedures, the number of informative features was reduced in the following sequence 62, 29, 19. The issue of choosing the most informative features based on the author's approach to minimizing features based on the use of linear models programming with partially boolean variables. As a result, the number of features was reduced to 8. Based on the results of the study, a patent was issued. The article is of interest both to professional medical workers, especially TB specialists, and machine learning specialists.

Keywords: tuberculosis without bacterioexcretion, machine learning, committee method, feature informativeness

ВВЕДЕНИЕ

По данным федеральной статистической отчетности, в Российской Федерации в 2018 г. у пациентов с впервые выявленным туберкулезом бактериовыделение зафиксировано только в 52,3 % случаев и не зафиксировано в 47,7 %. В 2019–2020 гг. [3] туберкулез без доказанного бактериовыделения регистрировался в 43,3–43,9 % случаев. То есть данная ситуация встречается достаточно часто.

Постановка диагноза по туберкулезу при отсутствии бактериовыделения является сложной задачей даже для профессиональных опытных фтизиатров. При обращении в противотуберкулезную службу пациенту проводят сбор анамнеза, осмотр, комплекс исследований. В общей сложности при постановке диагноза на основе клинических рекомендаций и нормативно-правовой документации [1–6, 8, 11–13] учитывается 139 различных показателей. Проблема состоит в том, что многие симптомы, которые есть у пациента, поступившего с подозрением на туберкулез, могут быть вызваны другими заболеваниями. Рентгенологические анализы в данном случае могут не дать результата, так как очаги заражения могут иметь столь малые размеры, что заметить их на рентгеновских снимках невозможно. Существует большое количество различных иммунологических тестов на туберкулез (туберкулинодиагностика, Диаскин тест, T-spot, QuantiFERON и т. д.). Однако все они имеют определенную погрешность и не исключают влияния человеческого фактора при трактовке их результатов. В результате примерно в 25 % случаев ставится неправильный диагноз и назначается неправильное лечение. Лечение больного туберкулезом обычно осуществляется на основе применения антибиотиков. Причем их количество назначается от 1 до 5. Такое лечение обычно плохо переносится пациентом и часто приводит к сильному дисбактериозу. В случае неправильно поставленного диагноза такое лечение пациенту наносит только вред.

Существенную помощь в постановке диагноза при отсутствии бактериовыделения могут оказать методы машинного обучения (МО). Результаты применения классических методов МО приведены нами в статье [8]. Хорошие результаты, в частности, показали методы линейного разделения множеств, логистической регрессии и опорных векторов. Поэтому мы продолжили исследования в этом направлении и оценили перспективы построения различных ансамблей (комитетных конструкций) на основе линейных разделителей. У нас имеется опыт построения комитетных конструкций применительно к туберкулезу [10], но объектом исследования у нас были дети и подростки в возрасте до 18 лет. Развитие туберкулезной инфекции и иммунный ответ организма на ее развитие у детей и взрослых имеет существенные отличия. Данное исследование проводилось на выборке из 740 пациентов, находившихся под наблюдением фтизиатров в 85 субъектах Российской Федерации в 2020–2021 гг. по поводу дифференциальной диагностики туберкулеза. Среди них пациентов с установленным туберкулезом (630) и пациентов с верифицированным диагнозом другой этиологии (110). Возраст пациентов от 18 до 70 лет. На начальном этапе учитывались 139 входных параметров. В результате человеко-машинных процедур анализа данных с применением классических методов МО количество информативных признаков было снижено до 62, которые были практическими специалистами (экспертами) разбиты на 6 групп. Далее каждая группа исследовалась классическими методами МО, на основе которой была выбрана группа из 29 показателей с наилучшими качественными характеристиками решающих правил (Ассигасу и AUC ROC). Однако даже при таком существенном снижении числа рассматриваемых признаков достаточно сложно построить целостную картину принятия решения при постановке диагноза, без которой обычно возникают проблемы с интерпретацией решающего правила (РП). Кроме того, была задача понять, какие признаки в качестве наиболее информативных выберет математическая модель и смогут ли этот выбор проинтерпретировать профессиональные фтизиатры. Мы не являемся сторонниками замены профессиональных медицинских специалистов системами на основе искусственного интеллекта, но нам было необходимо понять, насколько интерпретируем будет машинный выбор и какое минимальное количество признаков без существенного ухудшения качественных характеристик РП останется в результате этого выбора.

АНАЛИЗ ПРИЗНАКОВ НА ОСНОВЕ МЕТОДА СЛУЧАЙНОГО ЛЕСА

Кроме методов на основе линейных разделителей хорошие результаты показал метод случайного леса. Первоначально мы начали с него и исследовали группу из 29 признаков, на которой все перечисленные методы показали хорошие результаты. Методом случайного леса была сделана оценка информативности признаков. Данные приведены в таблице 1. Для сравнения там же приведены результаты оценки информативности методами f-classif, взаимной информации и хи-квадрат. С целью более наглядного представления значимость признаков была отнормирована относительно наиболее информативного признака. Высший бал равен 29.

Таблица 1 – Оценка информативности признаков методами МО

Наименование признака	Случайный Лес	F-classif	Взаимной информации	Хи-квадрат
Факт трудоустройства	29	29	29	29
Употребление алкоголя	12	13	10	2
Возраст	12	2	0	0
Фтизиатрическая группа риска	9	8	8	0
Выявление при обращении	6	0	0	0
Выявление при профосмотре	6	3	6	0
СОЭ (скорость оседания эритроцитов)	3	0	3	0
Уровень гемоглобина	3	2	1	4
Уровень эритроцитов	3	3	7	0
Кровохаркание	2	2	1	0
Уровень лейкоцитов	2	1	5	0
Факт установленного контакта	2	0	0	0
Пол	2	0	0	0
Курение	2	0	0	0
Сегментоядерные нейтрофилы	2	1	0	1
Лимфоциты	2	1	0	1
Поликлиническая группа риска	2	0	1	0
Слабость при выполнении обычной нагрузки	2	0	1	0
Кашель	2	1	3	0
Моноциты	1	0	1	0
Отсутствие флюорографии 2 и более лет	1	0	0	0
Палочкоядерные нейтрофилы	1	0	2	0
Факт нахождения в местах лишения свободы	1	0	0	0
Снижение массы тела	1	0	0	0
Температура 37 градусов и более	1	0	0	0
Потливость	1	0	0	0
Боли в груди	1	0	1	0
Факт наличия соответствующей патологии	0	0	0	0
Употребление наркотиков	0	0	3	0

Из данной таблицы следует, что наиболее информативным признаком является факт трудоустройства и это вполне объяснимо. Из таблицы также следует, что после девятнадцатого признака информативность признаков резко падает. Этот факт был проверен нами следующим образом. Мы последовательно, начиная с конца, исключали по одному признаку и оценивали, как это влияет на качественные показатели.

Влияние исключения признаков на качество решения мы оценивали по показателям чувствительности (True Positive Rate, TPR) и специфичности (True Negative Rate, TNR). По TPR мы оценивали процент правильного прогноза для больных туберкулезом, а по TNR – процент правильного прогноза для тех, у кого не туберкулез.

Обращаем внимание, что диагноз «не туберкулез» означает, что пациент имеет другое заболевание со схожими с туберкулезом симптомами и поэтому его надо направлять на дополнительное исследование при первом обращении. Исходя из этого, у нас были и требования к показателям TPR больше или равно 95 %, а TNR больше или равно 80 %

При использовании метода случайного леса, исключая признаки от № 29 до № 20 мы укладывались в указанные границы, а начиная с признака № 19 – нет. Таким образом, если для диагноза использовать метод случайного леса, то надо использовать первые девятнадцать признаков.

СОДЕРЖАТЕЛЬНАЯ ИНТЕРПРЕТАЦИЯ ПРИЗНАКОВ

В данном параграфе мы приведем мнение специалистов, почему именно признаки, приведенные в таблице 1, являются информативными:

Таблица 2 – Причины включения признаков в информативные

Признак	Причина для включения в информативные
Возраст	При туберкулезе чаще это лица молодого возраста 30–39 лет
Пол	Чаще страдают мужчины
Социальный статус	Туберкулезу чаще подвержены неработающие лица, лица без определенного места жительства, проживающие в приютах и других социальных учреждениях
Фтизиатрическая группа риска	Перенесенный туберкулез, контакт с больным туберкулезом, положительные иммунологические тесты
Отсутствие флюорографии более 2 лет	Риск развития туберкулеза
Выявление с жалобами	Менее характерно для выявления ограниченных процессов
Выявление на профосмотре	Активное выявление, чаще выявляет ограниченные поражения легких
Выявление из контакта	Этот факт увеличивает риск выявления туберкулезной этиологии
Наличие факта контакта	Подтверждение контакта с больным туберкулезом
Наличие факта снижения массы тела на 2 и более кг за последние 2–6 месяцев	Жалоба, подозрительная на туберкулез
Наличие факта повышения температуры более 37 в течение 2 недель и более	Жалоба, подозрительная на туберкулез
Уровень эритроцитов и гемоглобина отражают состояние красной крови	Микобактерии туберкулеза поглощают железо, что приводит к истощению ее содержания в эритроцитах и оказывает влияние на их уровень и уровень гемоглобина как переносчика железа и кислорода
Количество лейкоцитов	Характеризуют ответную реакцию организма на внедрение патогенного агента, способного к формированию хронического воспалительного процесса
Относительное содержание лимфоцитов и моноцитов	Определяет активацию/снижение Th1 пути иммунного ответа, определяющего ограничение туберкулезной инфекции
Количество палочкоядерных и сегментоядерных нейтрофилов	Определяют фагоцитоз возбудителя на первоначальном этапе, что должно сопровождать ранний период развития туберкулезного воспаления

Содержательная интерпретация информативности выбранных признаков является важным этапом в анализе данных, так как повышает доверие специалиста к тому, что выбрала машина.

Построение итерационного комитета старшинства на основе признаков, выбранных методом случайного леса. После выбора девятнадцати признаков методом случайного леса на основе этих признаков был построен итерационный комитет старшинства. Алгоритм построения данного комитета приведен нами в [6] и состоит в последовательном применении линейных разделителей с максимизацией правильных классификаций на каждом этапе и исключением их из дальнейшего рассмотрения. Естественно, что такая процедура всегда сходится, но важно за сколько шагов. Если число шагов большое, то найденное решение, скорее всего, будет переобученным и не пригодным для последующего применения. В результате было построено классифицирующее правило из пяти иерархически упорядоченных линейных разделителей, позволяющее определять наличие туберкулеза с TPR 98 %, а его отсутствие – с TNR 81%. То есть данный комитет не переобучен и соответствует требуемым качественным характеристикам. К числу достоинств итерационного метода следует отнести возможность быстрого решения задач большой размерности, но данный метод не гарантирует минимальный состав членов комитета (разделяющих гиперплоскостей) и не позволяет дополнительно минимизировать количество признаков.

МИНИМИЗАЦИЯ ЧИСЛА ВХОДНЫХ ПРИЗНАКОВ НА ОСНОВЕ МЕТОДА КОМИТЕТОВ

Существуют различные способы оценки информативности входных признаков в задачах МО. Эти способы можно разделить на две большие группы: основанные на анализе свойств входных данных и встроенные в метод МО. Методы F-classif, взаимной информации и хи-квадрат относятся к первой категории, а оценка информативности на основе случайного леса ко второй. К числу достоинств метода случайного леса следует отнести быстроту получения РП с высокими метриками качества решения, но, к сожалению, решение абсолютно не интерпретируемо (черный ящик). Более того, управлять качеством получаемого решения в процессе его получения практически невозможно, и оценка информативности и исключение неинформативных признаков делается постфактум с участием человека. В данном исследовании нас интересовало, сколько информативных признаков останется, если влияние человека исключить полностью и полагаться только на математическую модель и насколько будет интерпретируем этот выбор.

В случае если метод линейного разделения множеств показывает хорошие результаты, то за счет использования нескольких линейных разделителей эти результаты обычно можно улучшить. Ансамбль из линейных разделителей мы предпочитаем называть методом комитетов, и способы построения различного рода комитетных конструкций приведены нами в [9] и апробированы на практике [7, 10]. К числу достоинств такого подхода следует отнести тот факт, что минимизация числа признаков может быть напрямую включена в математическую модель за счет использования булевых переменных. Такого рода модели приведены нами в [8]. При использовании данных моделей мы начинали расчеты со всех 139 признаков. Расчеты проводились на основе комитетов единогласия, большинства и старшинства.

Таблица 3 – Метрики комитетов единогласия из трех членов и состав признаков

Число признаков	10	9	8	8	8	7	7
TPR	99	93	93,5	93,8	91,2	91,7	94,3
TNR	80	78	78,2	63,6	78,1	72	0,6
ПРИЗНАКИ							
Гистограмма		v		v	v		
Возраст	v	v	v	v	v	v	v
Пол	v						
Социальный статус	v						
Фтизиатрическая группа	v	v	v	v	v	v	v
Алкоголь		v	v	v	v	v	v
Выявление при обращении	v						
Выявление при профосмотре	v						
Выявление при обследовании при тубконтакте		v	v	v	v		v
Снижение массы тела	v						
Температура более 37 градусов	v						
Онкологические заболевания		v	v			v	
Несколько очагов		v	v	v	v	v	v
Эритроциты	v						
Лейкоциты	v						
Гемоглобин		v	v	v	v	v	v
Палочкоядерные нейтрофилы		v	v	v	v	v	v

Компромиссное решение для практического применения. После анализа результатов метода комитетов единогласия практическими специалистами было высказано пожелание на основе восьми признаков найти один линейный разделитель с максимально возможными показателями TPR и FRP. Поэтому в модели было зафиксировано количество признаков и использована многократная последовательная оптимизация функций цели. Сформулированным требованиям соответствует линейный разделитель с TPR 93 % и FPR 73 % со следующим составом и кодировкой входных признаков.

Формула решающего правила:

$$R = 0,055 * X1 + 2,456 * X2 + 0,67 * X3 - 1,294 * X4 + 2,589 * X5 - 1 * X6 - 0,416 * X7 - 0,025 * X8 - 2,645$$

где X1 – возраст (полных лет);

X2 – факт принадлежности к фтизиатрической группе (1 – есть, 0 – нет);

X3 – факт употребление алкоголя (0 – не употребляет, 1 – употребляет редко, 2 – нет данных, 3 – закодирован);

X4 – факт наличия контакта с больным туберкулезом (1 – был, 0 – нет);

X5 – факт наличия онкологического заболевания сейчас или в анамнезе (1 – есть, 0 – нет);

X6 – очаговое образование нескольких очагов, выявленных при лучевом методе обследования (1 – есть, 0 – нет);

X7 – палочкоядерные нейтрофилы (%);

X8 – лимфоциты (%);

2,645 – коэффициент.

Если $R \leq 0$, то у пациента туберкулез. Если $R > 0$, то у пациента другое заболевание.

Содержательная интерпретация признаков, кроме наличия факта онкологического заболевания, приведена нами в таблице 2. Наличие факта онкологического заболевания делает диагноз туберкулеза разной степени вероятности. Таким образом, математическая модель совершенно обоснованно включила данный признак в состав наиболее информативных. Более того, среди бинарных признаков этому признаку присвоен самый высокий коэффициент.

Проанализируем знаки в формуле перед признаками и свободным членом. Так как в нашем случае все пациенты поступили с подозрением на туберкулез, то вполне логично, что свободный член со знаком минус. Следовательно, признаки со знаком минус должны повышать вероятность выявить у пациента туберкулез, а со знаком плюс – снижать, обуславливая необходимость консультации других специалистов (онкологов, пульмонологов, торакальных хирургов). Естественно, что наличие контакта с больным туберкулезом, очаговые тени в легком, отклонение уровня нейтрофилов и лимфоцитов увеличивают шансы выявить у пациента туберкулезное воспаление. Несколько сложнее объяснить положительные знаки, но при тщательном анализе все выглядит тоже вполне логично. Положительный знак перед возрастом констатирует факт, что туберкулезом чаще болеют люди в молодом трудоспособном возрасте, с увеличением возраста увеличивается риск онкологических заболеваний (табл. 2). Наличие онкологического заболевания в анамнезе увеличивает риск развития у пациента онкопатологии легких. Употребление алкоголя не влияет на выявление этиологического фактора (возбудителя туберкулеза) и может встречаться как у пациентов с туберкулезом, так и другой патологии. Таким образом, приведенная формула достаточно хорошо интерпретируема, а ее простота может только способствовать ее практическому применению.

ЗАКЛЮЧЕНИЕ

В статье [10] авторами был сформулирован теоретический подход к управлению качеством решающего правила и минимизации числа признаков в задачах классификации на основе моделей математического программирования. Данный подход был проверен на реальной, сложной задаче с достаточно большим числом признаков и доказал свою работоспособность и практическую ценность. На его основе создано простое, но содержательно обоснованное решающее правило.

Преимущества способа состоят в возможности своевременно выявить туберкулезную инфекцию (до момента выделения микобактерий во внешнюю среду), что позволит регистрировать случай выявления ограниченного поражения легочной ткани как заболевание туберкулезом и начинать основной курс химиотерапии без ожиданий, пробных курсов химиотерапии, которые негативно сказываются не только на самочувствии пациента(ов), но и сопровождаются ростом недоверия врачу, снижением приверженности химиотерапии, отрывом от лечения, снижением эффективности химиотерапии и ростом лекарственно-резистентных форм возбудителей туберкулеза.

Способ определения активности туберкулезного процесса не влияет на здоровье пациентов и позволяет ускорить и автоматизировать процесс получения окончательного результата диагностики, исключив человеческий фактор. Предлагаемый способ не требует дополнительных инвазивных вмешательств и экономических затрат, так как сбор анамнеза, лучевые и лабораторные методы обследования являются обязательными и входят в клинический минимум при обследовании на туберкулез на первичном приеме у врача любой специальности. Однако может послужить обоснованием для назначения инвазивных диагностических манипуляций для верификации нетуберкулезного заболевания. Одномоментное внесение изучаемых показателей в формулу позволяет в течение нескольких минут получать информацию о наличии/отсутствии туберкулеза.

Список источников

1. Алексеева, Г. И. Бактерионосительство или латентный туберкулез? / Г. И. Алексеева, А. Ф. Кравченко // *Acta Biomedica Scientifica*. – 2012. – № 5–1 (87).
2. Клинические рекомендации «Туберкулез у взрослых». – 2022. – URL: https://cr.minzdrav.gov.ru/schema/16_2.
3. Отраслевые и экономические показатели противотуберкулезной работы в 2019–2020 гг. Аналитический обзор основных показателей и статистические материалы / под ред. С. А. Стерликова. – Москва : РИО ЦНИИОИЗ, 2021. – 63 с.
4. Приказ Министерства здравоохранения РФ от 15 ноября 2012 г. № 932н «Об утверждении Порядка оказания медицинской помощи больным туберкулезом». – URL: <https://cr.minzdrav.gov.ru/documents/9119>.
5. Сеницына, А. В. Оценка эффективности различных методов лучевой диагностики в выявлении туберкулеза у детей / А. В. Сеницына, П. В. Гаврилов, А. В. Сеницын, С. В. Михайлова, К. В. Прибыток, Е. В. Синельникова // *Педиатр*. – 2017. – Т. 8, вып. 3. – С. 94–100.
6. Старшинова, А. А. Иммунодиагностика туберкулеза сегодня: современные иммунологические тесты и дифференцированный подход к их применению в практике / А. А. Старшинова, И. Ф. Довгалюк, Ю. С. Зинченко, Д. А. Кудлай, П. К. Яблонский // *Практическая пульмонология*. – 2019. – № 2. – С. 28–32.
7. Тюлькова, Т. Е. Практическое применение методов машинного обучения на примере определения активности туберкулезного процесса у лиц с минимальными туберкулезными изменениями, выявленными на рентгенограмме органов грудной клетки / Т. Е. Тюлькова, П. Ф. Чернавин, Н. П. Чернавин // *Клинический вестник ФМБЦ им. А.И. Бурназяна*. – 2022. – № 2. – С. 64–73.
8. Фтизиатрия: национальное руководство / главный редактор М. И. Перельман. – Москва : ГЭОТАР-Медиа, 2007. – 505 с.
9. Чернавин, П. Ф. Машинное обучение на основе задач математического программирования / П. Ф. Чернавин, Д. Н. Гайнанов, В. Н. Панкращенко, Ф. П. Чернавин, Н. П. Чернавин. – Москва : Наука, 2021. – 128 с.

10. Чернавин, П. Ф. Управление качеством решающего правила и минимизация числа признаков в задачах классификации на основе моделей математического программирования / П. Ф. Чернавин, Н. П. Чернавин, Ф. П. Чернавин // Прикаспийский журнал: управление и высокие технологии. – 2023. – № 1. – С. 112–119.
11. Datta, S. Comparison of sputum collection methods for tuberculosis diagnosis: a systematic review and pairwise and network meta-analysis / S. Datta, L. Shah, R. H. Gilman, C. A. Evans // *Lancet Glob Health*. – 2017 Aug. – Vol. 5 (8). – e760-e771.
12. Divala, T. H. Accuracy and consequences of using trialof-antibiotics for TB diagnosis (ACT-TB study): protocol for a randomised controlled clinical trial / T. H. Divala, K. L. Fielding, D. J. Sloan et al. // *BMJ Open*. – 2020. – Vol. 10. – e033999. – doi: 10.1136/ bmjopen-2019-033999.
13. Zuberi, F. F. Role of Bronchial Washing Gene Xpert in Sputum-Scarce Cases of Suspected Pulmonary Tuberculosis / F. F. Zuberi, S. Hussain, S. Hameed, B. F. Zuberi // *Pak. J. Med. Sci.* – 2019. – Vol. 35 (1). – P. 211–214.

References

1. Alekseyeva, G. I., Kravchenko, A. F. Bacteriocarrier or latent tuberculosis? *Acta Biomedica Scientifica*, 2012, no. 5–1 (87) (In Russ.).
2. *Clinical guidelines "Tuberculosis in adults"*, 2022. URL: https://cr.minzdrav.gov.ru/schema/16_2 (In Russ.).
3. *Industry and economic indicators of TB work in 2019–2020. Analytical review of key indicators and statistical materials*. Ed. S. A. Sterlikov. Moscow, RIO TsNIOIZ, 2021. 63 p. (In Russ.).
4. Order of the Ministry of Health of the Russian Federation of November 15, 2012 No. 932n "On Approval of the Procedure for Providing Medical Care to Patients with Tuberculosis". (In Russ.).
5. Sinitsyna, A. V., Gavrilov, P. V., Sinitsyn, A. V., Mikhaylova, S. V., Pribytok K. V., Sinelnikova, Ye. V. Evaluation of efficiency of different methods of radiation diagnosis in the detection of tuberculosis in children. *Pediatrician*, 2017, Vol. 8, issue 3, pp. 94–100.
6. Starshinova, A. A., Dovgalyuk, I. F., Zinchenko, Yu. S., Kudlay, D. A., Yablonskiy, P. K. Immunodiagnosis of tuberculosis: modern immunological tests and differentiated approach to their use in practice. *Practical pulmonology*, 2019, no. 2, pp. 28–32 (In Russ.).
7. Tyulkova, T. Ye. Chernavin, P. F., Chernavin, N. P. Practical application of machine learning methods on the example of determining the activity of the tuberculosis process in individuals with minimal tuberculous changes detected on chest X-ray. *Clinical Bulletin of the A. I. Burnazyan State Medical Research Center*, 2022, no. 2, pp. 64–73 (In Russ.).
8. *Phthiology: national leadership*. Editor-in-chief M. I. Perelman. Moscow, GEOTAR-Media Publ., 2007. (In Russ.).
9. Chernavin, P. F., Gaynanov, D. N., Pankrashchenko, V. N., Chernavin, F. P., Chernavin, N. P. *Machine learning based on mathematical programming problems*. Moscow, Nauka Publ., 2021. 128 p. (In Russ.).
10. Chernavin, P. F., Chernavin, N. P., Chernavin, F. P. Control of the quality of the decision rule and minimization of the number of features in classification problems based on mathematical programming models. *Caspian Journal: Control and High Technologies*, 2023, no. 1, pp. 112–119 (In Russ.).
11. Datta S, Shah L, Gilman RH, Evans CA. Comparison of sputum collection methods for tuberculosis diagnosis: a systematic review and pairwise and network meta-analysis. *Lancet Glob Health*, 2017 Aug, Vol. 5 (8), e760-e771.
12. Divala, T. H., Fielding, K. L., Sloan, D. J. et al. Accuracy and consequences of using trialof-antibiotics for TB diagnosis (ACT-TB study): protocol for a randomised controlled clinical trial. *BMJ Open*, 2020, vol. 10, e033999. doi: 10.1136/ bmjopen-2019-033999.
13. Zuberi, F. F, Hussain, S., Hameed, S., Zuberi, B. F. Role of Bronchial Washing Gene Xpert in Sputum-Scarce Cases of Suspected Pulmonary Tuberculosis. *Pak. J. Med. Sci.*, 2019, vol. 35 (1), pp. 211–214.

Статья поступила в редакцию 21.09.2023; одобрена после рецензирования 09.10.2023; принята к публикации 05.03.2024.

The article was submitted 21.09.2023; approved after reviewing 09.10.2023; accepted for publication 05.03.2024.

МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ, ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

УДК 004.942

МЕТОД ПОСТРОЕНИЯ ТРАЕКТОРИИ ПОЛЕТА БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ В УСЛОВИЯХ КИБЕРАТАК GPS/ГЛОНАСС СПУФИНГА

Петренко Вячеслав Иванович, Северо-Кавказский федеральный университет, 355017, Российская Федерация, г. Ставрополь, ул. Пушкина, 1,
кандидат технических наук, заведующий кафедрой, ORCID: 0000-0003-4293-7013, e-mail: vipetrenko@ncfu.ru

Тебужева Фариза Биляловна, Северо-Кавказский федеральный университет, 355017, Российская Федерация, г. Ставрополь, ул. Пушкина, 1,
доктор физико-математических наук, заведующий кафедрой, ORCID: 0000-0002-7373-4692, e-mail: fbtebueva@ncfu.ru

Волошин Денис Геннадиевич, Северо-Кавказский федеральный университет, 355017, Российская Федерация, г. Ставрополь, ул. Пушкина, 1,
аспирант ORCID: 0009-0007-4841-0147, e-mail: ultrageron@gmail.com

Целью исследований настоящей работы является повышение защищенности беспилотных летательных аппаратов (БПЛА) в условиях воздействия на аппарат активных помех. Показано, что развитие БПЛА и средств противодействия им приводит к необходимости создания более безопасных методов управления БПЛА. Такие методы охватывают широкий спектр научных и технических задач, в том числе и задачу построения безопасного маршрута БПЛА. Рассмотрены уже имеющиеся решения этой задачи, которые подразумевают построение маршрута без учета применения активных помех на пути или построение маршрута лишь для возврата БПЛА на место старта. В статье проведен анализ основных подходов методов планирования пути в пространстве. Сделан вывод о недостатках двумерного представления и визуализации. В имитационной модели представлена трехмерная визуализация расчетной траектории выполнения полетного задания беспилотным летательным аппаратом. Имитационная модель объединяет решения подзадач: моделирование трехмерной среды с препятствиями; построение пути беспилотного летательного аппарата в такой среде с обходом препятствий. Моделирование миссий с помощью программного комплекса проводится последовательно шаг за шагом. Программный комплекс построен на языке C# по модульному принципу.

Ключевые слова: БПЛА, траектория полета, кибератака, спуфинг, GPS, ГЛОНАСС, бортовая система, внешние воздействия

METHOD FOR PLANING AN UNMANNED AERIAL VEHICLE FLIGHT PATH IN CONDITIONS OF GPS/GLONASS SPOOFING CYBERATTACKS

Petrenko Vyacheslav I., North Caucasian Federal University, 1 Pushkin St., Stavropol, 355017, Russian Federation,

Cand. Sci. (Engineering), Head of Department, ORCID: 0000-0003-4293-7013, e-mail: vipetrenko@ncfu.ru

Tebueva Fariza B., North Caucasian Federal University, 1 Pushkin St., Stavropol, 355017, Russian Federation,

Doct. Sci. (Physics and Mathematics), Head of Department, ORCID: 0000-0002-7373-4692, e-mail: fbtebueva@ncfu.ru

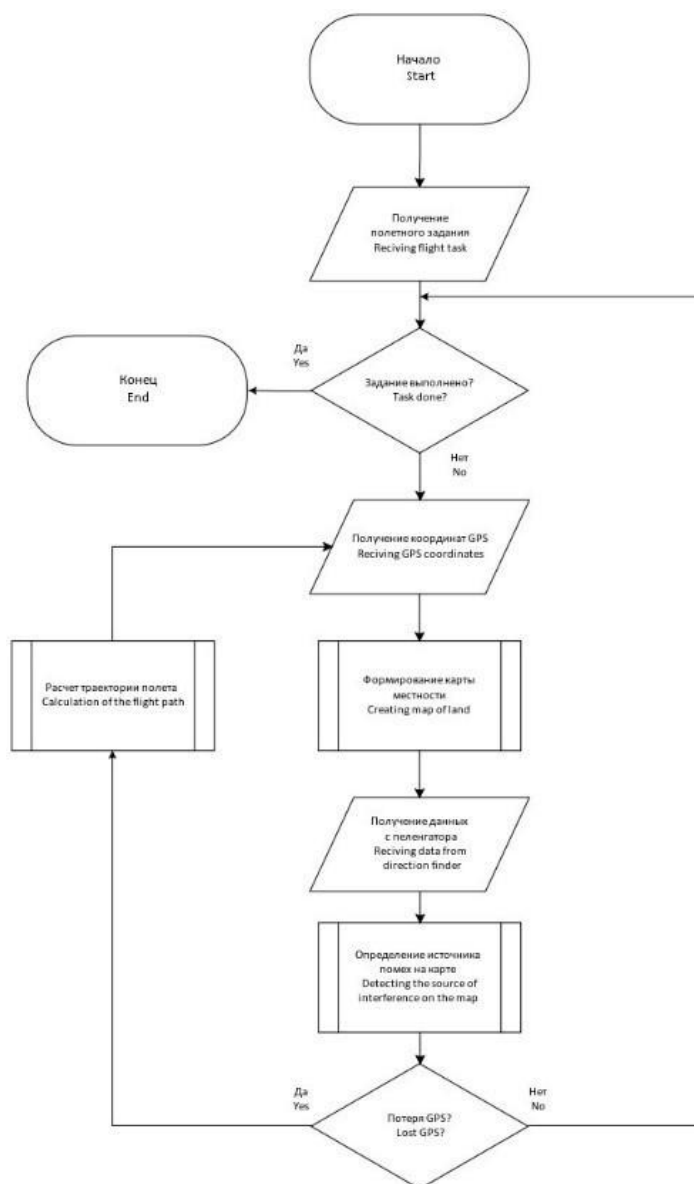
Voloshin Denis G., North Caucasus Federal University, 1 Pushkin St., Stavropol, 355017, Russian Federation,

postgraduate student, ORCID: 0009-0007-4841-0147, e-mail: ultrageron@gmail.com

The purpose of the research of this work is to increase the security of unmanned aerial vehicles (UAVs) under the conditions of exposure to active interference on the device. It is shown that the development of UAVs and means of countering them leads to the need to create safer methods of UAV control. Such methods cover a wide range of scientific and technical tasks, including the task of constructing a safe UAV flight path. The considered already available solutions to this problem, which imply the construction of a flight path without taking into account the use of active interference on the way or the construction of a flight path only for the return of the UAV to the start point. The article analyzes the main approaches of path planning methods in space. The conclusion is made about the disadvantages of two-dimensional representation and visualization. The simulation model presents a three-dimensional visualization of the calculated trajectory of the flight task by an unmanned aerial vehicle. The simulation model combines solutions to subtasks: modeling a three-dimensional environment with obstacles; building the path of an unmanned aerial vehicle in such an environment with obstacle avoidance. Simulation of missions using the software package is carried out sequentially step by step. The software package is built in the C# language according to the modular principle.

Keywords: UAV, flight path, cyber-attack, spoofing, GPS, GLONASS, on-board system, external influences

Graphical annotation (Графическая аннотация)

**ВВЕДЕНИЕ**

В настоящее время для выполнения различных миссий беспилотных летательных аппаратов (БПЛА) используется дистанционное управление БПЛА, качество сигнала в канале управления между БПЛА и станцией управления сильно зависит от внешних факторов и воздействий. Преодоление внешних факторов и воздействий приводит к увеличению автономности управления БПЛА. Это требует разработки таких методов и алгоритмов управления БПЛА, которые позволят выполнять полетные задания БПЛА по заданной траектории без участия человека [1–2].

Планирование траекторий полета БПЛА зависит от того, с какой целью используется летательный аппарат. Типичной здесь является задача, в которой БПЛА следует из одной заданной точки в другую. Основным элементом для определения точных координат БПЛА в пространстве в предложенном методе является навигационная система, которая использует данные с пеленгатора, фазовые антенны которого установлены на БПЛА. В работе [3] описан способ повышения дальности обнаружения таких пеленгаторов, что позволит в будущем увеличить площадь видимости пеленгатора при его использовании.

Траекторию полета БПЛА можно рассматривать как геометрический путь с профилем скорости вдоль него. Задачу планирования траектории полета БПЛА можно рассматривать как прокладку

геометрического пути с обходом препятствий и дальнейшую разработку временного графика движения вдоль геометрического пути.

Проведенные авторами статьи исследования показывают необходимость выполнения полетного задания с высокой степенью автономности. При этом целесообразным является корректировка полетного задания во время полета бортовой системой БПЛА, что позволяет БПЛА реагировать на внешние факторы и успешно нивелировать любые погрешности, возникающие во время полета. Такая адаптация процесса полета существенно повысит вероятность возврата БПЛА на место старта.

Однако в настоящее время уровень современных информационных технологий на легких БПЛА усложняет реализацию данной задачи в полном объеме. В ряде существующих аппаратов [4] используется контроллер Atmega 2560 с тактовой частотой 16 МГц, а также дополняется сопутствующим вычислительным блоком на базе процессоров Cortex-A53 1,2 ГГц. Несмотря на наличие более мощных вариантов конфигурации БПЛА, использование дополнительных вычислительных ресурсов ведет к непропорционально высокому росту цены такого аппарата. Таким образом, можно подчеркнуть актуальность адаптации траектории полета БПЛА и метода автоматического построения траектории БПЛА с учетом воздействия внешних помех.

АНАЛИЗ ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ И ПОСТАНОВКА ПРОБЛЕМЫ

Анализ литературы показал, что имитационному моделированию процесса прокладки пути геометрического пути в окружающей среде посвящен ряд работ [1–2, 5].

Среди ряда подходов к решению задачи в процессе моделирования можно выделить методы теории графов с применением процедуры сглаживания [6]. Построение сложных пространственных траекторий и полетов вдоль траекторий различной степени гладкости с использованием методов построения траектории с наименьшей ошибкой отклонения и задаваемые контрольными точками рассмотрены в [3, 7].

Из приведенных работ можно сделать вывод о детальной проработке методов планирования траектории БПЛА, однако существенным недостатком этих методов является отсутствие учета источников внешних помех в процессе построения траектории полета. Предложенные методы успешно применяются в таких практических задачах, как автоматическое вождение, складская обработка и т. п. Эти методы строятся на основе графов и деревьев [8–10], которые довольно успешно работают в условиях отсутствия внешних помех, имеют большую востребованность и описаны во многих работах. Метод построения траектории полета с использованием потенциальных полей [11] работает быстрее, но если рассматривается область пространства, которая не является односвязной, то скалярный потенциал может быть многозадачной функцией, решение которой требует значительных вычислительных ресурсов. Однако в данной работе планирование траектории полета БПЛА выполняется с учетом источников активных помех в пространстве. Рассматриваемая задача более сложная, чем планирование траектории полета без помех и проработана не так тщательно.

При этом корректное выполнение поставленных задач с оценкой их безопасности невозможно без предварительного имитационного моделирования. Противоречие на практике заключается в отсутствии методов, использование которых позволяет в автоматическом режиме с высокой вероятностью осуществлять построение маршрута БПЛА в режиме реального времени с учетом воздействия активных помех. Таким образом, разработка метода построения траектории БПЛА в условиях применения активных помех является актуальной научной задачей.

СУЩЕСТВУЮЩИЕ МЕТОДЫ НАВИГАЦИИ БПЛА В УСЛОВИЯХ КИБЕРАТАК

При использовании спутниковой и инерциальной системы навигации БПЛА имеется возможность сравнить данные спутниковой навигации с данными инерциальной системы навигации. Учитывая, что курс БПЛА для инерциальной системы навигации является одним из основных показателей, ложная смена курса БПЛА при GPS/Глонасс спуфинге будет определена с высокой степенью вероятности. Авторы работы [12] используют фильтр Кальмана для установления навигации по динамической траектории цели. Для уменьшения сложности метода используется алгоритм М-лучшего позиционирования. Преимущество метода заключается в его дешевизне и простоте использования на уже существующих БПЛА, которые уже имеют спутниковую и инерциальную систему навигации. Однако недостатком метода является его уязвимость перед умышленными спуфинг-атаками, которые подменяют истинные координаты спутниковой навигации с заданной скоростью и, как правило, не превышают пороговых значений, по которым их можно идентифицировать как ложные. В работе авторы также описывают метод на основе машинного обучения для выявления ложных сигналов спутниковой навигации.

МЕТОД ПОСТРОЕНИЯ ТРАЕКТОРИИ ПОЛЕТА БПЛА В УСЛОВИЯХ КИБЕРАТАК GPS/ГЛОНАСС СПУФИНГА

Рассмотрим автономный режим управления беспилотного летательного аппарата. Наиболее уязвимым местом для перехвата управлением БПЛА является автономный режим, когда БПЛА сам принимает решения о своем перемещении. При включении автономного режима управления

для возврата на базу в стандартной бортовой системе используется система спутниковой навигации. Из-за общеизвестности этого факта злоумышленники пытаются имитировать с ложными параметрами или поставить преднамеренную помеху сигналу спутниковой навигации. С технической точки зрения такая возможность очень сложна, но широко известны факты захвата БПЛА таким способом [8]. Следовательно, имеет смысл отказаться от обычных методов позиционирования и использовать устойчивые к помехам методы управления БПЛА. Схема стандартного навигационного блока показана на рисунке 1. На схеме изображены элементы бортовой системы БПЛА, которые отвечают за его положение в пространстве. Схема предлагаемого навигационного блока показана на рисунке 2. В момент подавления сигнала бортовая система БПЛА запоминает последние координаты с помощью GPS, строит обратный маршрут в безопасную зону или строит безопасный маршрут по указанному заданию, имея на борту карту местности. Определение момента перехвата производится с помощью параметра схожести координат спутниковой навигации и координат, полученных от навигационной системы по данным пеленгатора.

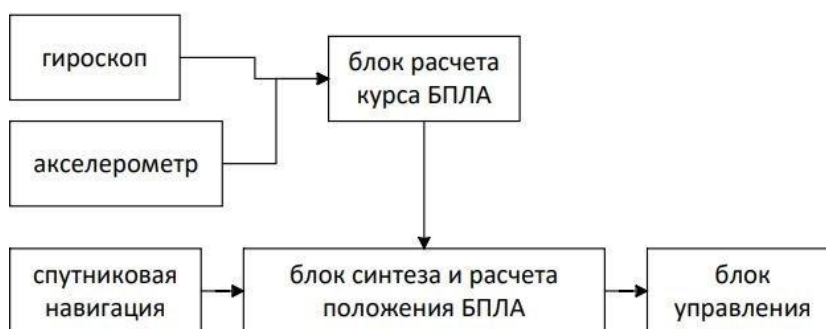


Рисунок 1 – Стандартная схема навигационного блока БПЛА



Рисунок 2 – Предлагаемая схема навигационного блока БПЛА

Инерциальная система позиционирования включает в себя данные с гироскопа, акселерометра. На рисунках 1–2 она изображена в виде блоков «гироскоп» и «акселерометр».

Для компенсации ошибки инерциальной системы позиционирования, которая накапливается с течением времени, предлагается использовать источник сигнала активных помех как точку для вычисления своего текущего местоположения, при этом сам источник помех должен быть стационарным. Бортовая система получает данные от двухфазных пеленгационных антенн и вычисляет угловое положение между антеннами (рис. 3). В случае если вероятный злоумышленник будет использовать несколько источников подавления сигнала, бортовая система определяет максимальный угол отклонения от предыдущего угла к текущему.



Рисунок 3 – Схема нахождения БПЛА в зоне работы активных помех

Выполним математическое описание предлагаемого метода. Зная расстояние между двумя фазовыми антеннами D , углы, указывающие на источник сигнала α и β , можно рассчитать расстояние от БПЛА до источника помех (рис. 4). Для этого нужно найти две стороны треугольника с известной стороной D углы α и β .

$$d_2 = D \frac{\sin \alpha}{\sin \psi}, \quad (1)$$

$$d_1 = D \frac{\sin \beta}{\sin \psi}, \quad (2)$$

где d_1 – расстояние от первой антенны БПЛА до антенны постановки помех,
 d_2 – расстояние от второй антенны БПЛА до антенны постановки помех,
 D – расстояние между антеннами,

α – угол между осью двух антенн и источником активных помех с точкой антенны 1,
 β – угол между осью двух антенн и источником сигнала глушения с точкой антенны 2.
 ψ – угол между двумя пеленгаторами с вершиной в источнике активных помех.

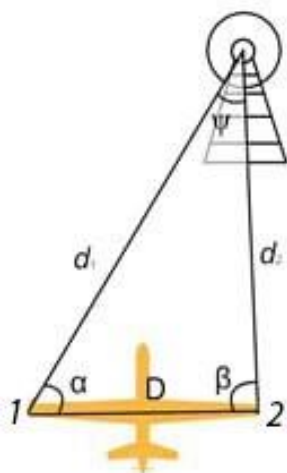


Рисунок 4 – Схема «БПЛА – источник помех»

Далее, чтобы определить точное расстояние между БПЛА и источником помех, найдем длину медианы M треугольника (рис. 5).

$$M = \frac{1}{2} \sqrt{2d_1^2 + 2d_2^2 - D^2}. \quad (3)$$

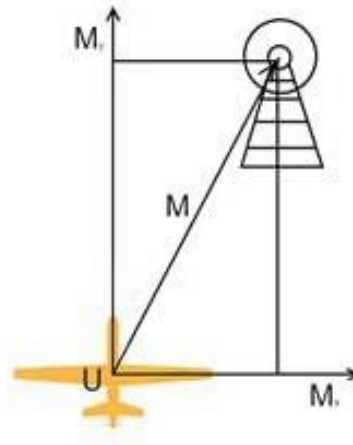


Рисунок 5 – Схема поиска расстояния до источника сигнала глушения

$$\left\{ \begin{array}{l} M_x = M \sin(MUM_y) \\ M_y = M \sin(MUM_x) \\ M_z \geq H_{\max} \end{array} \right\}, \quad (4)$$

где M_x – перемещение БПЛА по X согласно данным вычислений контроллера фазовой антенны;

M_y – перемещение БПЛА по Y согласно данным вычислений контроллера фазовой антенны;

M_z – перемещение БПЛА по Z ;

H_{\max} – максимальная высота, на которой работает источник помех, задается константой.

Далее представим среднее арифметическое смещения БПЛА в плоскости XU в виде формулы:

$$\left\{ \begin{array}{l} x_{offset} = \frac{M_x + x_{acs}}{2} \\ M + y \\ y_{offset} = \frac{y_{acs}}{2} \end{array} \right\}, \quad (5)$$

где x_{offset} – среднеарифметическое смещение по оси x ;

y_{offset} – среднеарифметическое смещение по оси y ;

x_{acs} – отклонение согласно данным акселерометра по оси x ;

y_{acs} – отклонение согласно данным акселерометра по оси y .

Получив смещения в плоскости XU , появляется возможность использовать эти данные в системе координат GPS. Координаты БПЛА при последнем достоверно верном положении были сохранены в момент времени t_{save} .

Рассмотрим погрешность метода позиционирования инерциальной системы E_i , которая зависит от скорости ветра и продолжительности полета. Скорость БПЛА не учитывается, так как сравниваемые методы выполняют изменения дискретно, а сама скорость полета БПЛА на порядки ниже скорости измерений, что могло бы дать значительные показатели лишь в случае накопительного эффекта.

$$E_i = \sqrt{(v_x t)^2 + (v_y t)^2}, \quad (6)$$

где v_x – скорость ветра по оси x ;

v_y – скорость ветра по оси y ;

t – время полета.

Погрешность предложенного метода в случае использования только одной фазовой антенны будет зависеть также и от направления ветра. В случае, если направление ветра будет перпендикулярно прямой БПЛА – помеха, погрешность будет равна нулю, так как любое смещение по оси направления ветра приведет к изменению угла между БПЛА и источником. Соответственно, чем ближе направление ветра к прямой пеленгации, тем ближе погрешность метода к погрешности инерциальной системы позиционирования (рис. 6). Зависимость погрешности от скорости ветра и времени полета можно выразить как:

$$E_p = \sqrt{(v_x t)^2 + (v_y t)^2} \cos \rho, \quad (7)$$

где v_x – скорость ветра по оси x ;

v_y – скорость ветра по оси y ;

t – общее время полета;

ρ – угол между помехой и направлением ветра.



Рисунок 6 – Угол между вектором пеленгации и направлением ветра

Для наглядности построим график, который будет отображать зависимость между углом ρ и погрешностью метода с пеленгацией только одной. Скорость ветра взята за константу.

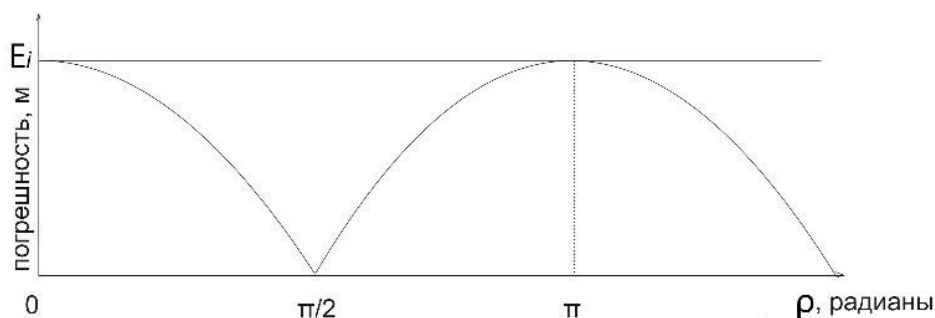


Рисунок 7 – Погрешность метода пеленгации

Исходя из графика (рис. 7), можно сделать вывод, что погрешность метода с использованием пеленгации одной фазовой антенны достигает уровня погрешности стандартного метода только в углах 0 и π , т. е., соответственно, когда ветер попутный или встречный, а во всех остальных точках имеет более низкое значение. Однако при использовании двух антенн с последующим расчетом расстояния до источника сигнала активных помех, погрешность метода будет зависеть от точности пеленгации угла антенны к источнику активных помех, т. е. от технических характеристик самого пеленгатора.

ОПИСАНИЕ РАБОТЫ МОДЕЛИ

Предложенный метод в отличие от существующих предоставляет возможность выполнить полетную задачу путем следования до точки ее выполнения по просчитанному безопасному маршруту. В отличие от прямого следования БПЛА к точке старта, при котором аппарат пересекает

источник активных помех, предложенный метод позволяет облетать источники активных помех. Предложенный метод, в отличие от метода нахождения пути Дейкстры, использует маневрирование по высоте и сближение к условной линии «свой/чужой», что уменьшает вероятность перехвата. В модели линия разграничения «свой/чужой» представлена нижним краем матрицы. Сама матрица 15x15 полностью находится на условной «чужой» территории.

При моделировании процесса расчета и построении траектории полета БПЛА было решено несколько задач в определенном порядке.

1. Разработан специальный графический интерфейс ввода исходных данных и вывода результатов.

2. Разработан блок подпрограмм моделирования пространства выполнения полетного задания с учетом места расположения и сигналов средств внешнего воздействия (например, станций радиоэлектронной борьбы (РЭБ)).

3. Разработан блок подпрограмм моделирования расчета и визуализации траектории полета.

Для удобства и упрощения использования функционала программного комплекса имитационной модели разработан специальный графический интерфейс. Схема комплекса представлена на рисунке 8.



Рисунок 8 – Схема комплекса программных модулей моделирования безопасной полетной траектории БПЛА

Программный комплекс выполнен по объектно-ориентированной методологии. Все блоки разбиты на классы.

ЗАКЛЮЧЕНИЕ

Данные были получены при построении маршрута от края до центра прямоугольной матрицы 15x15, количество источников активных помех – 15 единиц, количество опытов – 1000. В случае использования стандартного метода возврата по прямой, БПЛА пересекает источники активных помех в 44 % случаев, в то время как модифицированный метод огибает их во всех проведенных опытах (рис. 9–10).

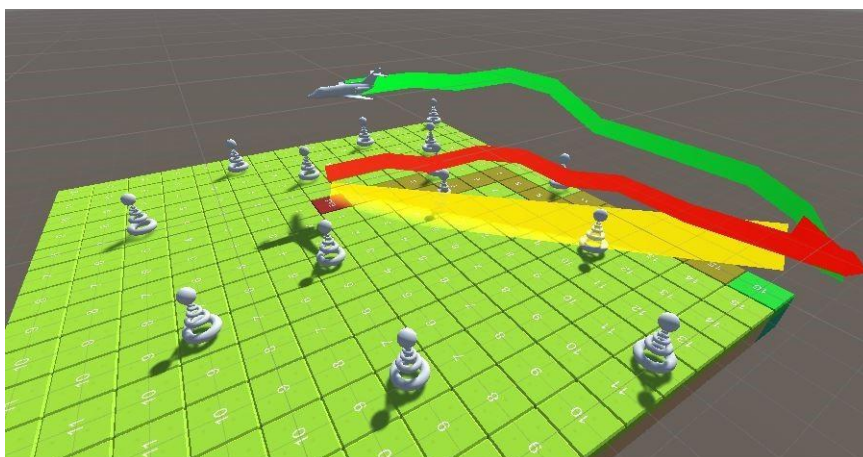


Рисунок 9 – Результаты моделирования 1



Рисунок 10 – Результаты моделирования 2

На рисунках 9–10 показаны траектории полета БПЛА. Желтым – стандартный метод, при котором БПЛА возвращается на место старта при воздействии внешних источников. Красным – метод нахождения кратчайшего пути в двухмерном пространстве. Зеленым – предложенный метод. Созданная при этом имитационная модель решает задачу построения траектории полета беспилотных летательных аппаратов в окружающей среде с обходом источников активных помех. Моделирование задач с помощью программного комплекса проводится последовательно, шаг за шагом. Программный комплекс построен в среде Unity на языке C# по модульному принципу. В статье описана математическая основа работы модели, функциональная структура комплекса, отдельно обсуждается структура графического интерфейса комплекса. Проиллюстрирован пример расчета и корректировки безопасной полетной траектории БПЛА. Планируются дальнейшие исследования в области оценки рисков информационной безопасности БПЛА в зонах излучения активных источников помех.

Список источников

1. Васильченко, А. С. Формирование маршрутов полета беспилотных летательных аппаратов с учетом местоположения средств противовоздушной обороны и радиоэлектронного подавления / А. С. Васильченко, М. С. Иванов, Г. Н. Колмыков // Системы управления, связи и безопасности. – 2019. – № 4. – С. 404–407.
2. Колчинский, И. М. Построение оптимального маршрута облета территории с препятствиями для сбора данных беспилотными летательными аппаратами / И. М. Колчинский, В. М. Хачумов // Глобус. – 2020. – С. 58–62.
3. Быстров, Р. П. Микро- и нанoeлектроника применительно к системам радиолокации и радиосвязи / Р. П. Быстров и др. // Успехи современной радиоэлектроники. – 2010. – № 9. – С. 19.
4. Shuo, Wang. Design of Autonomous Flight Control System for Small-Scale UAV / Shuo Wang, Ziyang Zhen, Fengying Zheng, Xinhua Wang // Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference. – 2014. – P. 1885–1888.
5. Sujit, B. P. Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicle / B. P. Sujit, S. Saripalli, Sousa J. Borges // Control Systems. – 2014. – Vol. 34, № 1. – P. 42–59.
6. Кочакаров, А. А. Сравнительный анализ полетов квадрокоптера вдоль траекторий различной степени гладкости / А. А. Кочакаров, Р. Т. Агишев // Современная наука. – 2016. – № 3. – С. 17–22.
7. Ткачев, С. Б. Автоматическая генерация сложных пространственных траекторий БПЛА и синтез управлений / С. Б. Ткачев, А. П. Крищенко, А. Н. Канатников // Математика и математическое моделирование. – 2015. – Т. 201.
8. Руденко, Э. М. Маршрутизация беспилотных летательных аппаратов, трансцендентные целевые функции графа и генетический алгоритм / Э. М. Руденко, Е. В. Семикина // Научно-технические исследования в космических исследованиях Земли. – 2021. – Т. 13, № 1. – С. 6–15.
9. Khatib, O. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots / O. Khatib // The International Journal of Robotics Research. – 1985. – Vol. 5, № 1. – P. 90–98.
10. Xuming, He. Monotone B-Spline Smoothing / He Xuming, Shi Peide // Journal of the American Statistical Association. – October 1997. – DOI: 10.2307/2670115.
11. Park, S. A new nonlinear guidance logic for trajectory tracking / S. Park, J. Deyst, P. J. How // AIAA Guidance, Navigation, and Control Conference and Exhibit. – Providence, Rhode Island, 16–19 August 2004. – Art. No. AIAA 2004-4900.
12. Pardhasaradhi, B. Navigation in GPS Spoofed Environment Using M-Best Positioning Algorithm and Data Association / B. Pardhasaradhi, P. Srihari, P. Aparna // IEEE Access. – 2021. – Vol. 9. – P. 51536–51549.

13. Dijkstra, E. W. A note on two problems in connexion with graphs / E. W. Dijkstra, F. Brezzi // *A Numer. Math.* – 1959. – Vol. 1, iss. 1. – P. 269–271.
14. Fan, X. Dynamic obstacle avoiding path plan for robots based on a new artificial potential field function / X. Fan, S. Li, Tefang T. Chen // *Control Theory & Applications.* – 2005. – Vol. 22, № 5. – P. 703–707.

References

1. Vasilchenko, A. S., Ivanov, M. S., Kolmykov, G. N. Formation of flight routes for unmanned aerial vehicles taking into account the location of air defense and electronic suppression systems. *Control Systems, Communications and Security*, 2019, no. 4, pp. 404–407 (In Russ.).
2. Kolchinsky, I. M., Khachumov, V. M. Construction of an optimal route to fly around an area with obstacles for collecting data by unmanned aerial vehicles. *Globus*, 2020, pp. 58–62 (In Russ.).
3. Bystrov, R. P. et al. Micro- and nanoelectronics as applied to radar and radio communication systems. *Advances in Modern Radio Electronics*, 2010, no. 9, p. 19 (In Russ.).
4. Shuo, Wang, Ziyang, Zhen, Fengying, Zheng, Xinhua, Wang. Design of Autonomous Flight Control System for Small-Scale UAV. *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, 2014, pp. 1885–1888.
5. Sujit, B. P., Saripalli, S., Borges, Sousa J. Unmanned Aerial Vehicle Path Following: A Survey and Analysis of Algorithms for Fixed-Wing Unmanned Aerial Vehicles. *Control Systems*, 2014, vol. 34, no. 1, pp. 42–59.
6. Kochakarov, A. A., Agishev, R. T. Comparative analysis of quadcopter flights along trajectories of varying degrees of smoothness. *Modern Science*, 2016, no. 3, pp. 17–22 (In Russ.).
7. Tkachev, S. B. Krischenko, A. P., Kanatnikov, A. N. Automatic generation of complex spatial trajectories of UAVs and control synthesis. *Mathematics and Mathematical Modeling*, 2015, vol. 201 (In Russ.).
8. Rudenko, E. M., Semikina, E. V. Routing of unmanned aerial vehicles, transcendental objective graph functions and genetic algorithm. *Science-Intensive Technologies in Space Research of the Earth*, 2021, vol. 13, no. 1, pp. 6–15 (In Russ.).
9. Khatib, O. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. *The International Journal of Robotics Research*, 1985, vol. 5, no. 1, pp. 90–98.
10. Xuming, He, Peide, Shi. Monotone B-Spline Smoothing. *Journal of the American Statistical Association*, October 1997. DOI: 10.2307/2670115.
11. Park, S., Deyst, J., How, P. J. A new nonlinear guidance logic for trajectory tracking. *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Providence, Rhode Island, 16–19 August 2004, Art. No. AIAA 2004-4900.
12. Pardhasaradhi, B., Srihari, P. Aparna, P. Navigation in GPS Spoofed Environment Using M-Best Positioning Algorithm and Data Association. *IEEE Access*, 2021, vol.9, pp. 51536–51549
13. Dijkstra, E. W., Brezzi, F. A note on two problems in connexion with graphs. *Numer. Math.*, 1959, vol. 1, iss. 1, pp. 269–271.
14. Fan, X., Li, S., Chen, Tefang T. Dynamic obstacle avoiding path plan for robots based on a new artificial potential field function. *Control Theory & Applications*, 2005, vol. 22, no. 5, pp. 703–707.

Статья поступила в редакцию 19.02.2024; одобрена после рецензирования 22.02.2024; принята к публикации 13.03.2024.

The article was submitted 19.02.2024; approved after reviewing 22.04.2024; accepted for publication 13.03.2024.

УДК 004.056.5

**МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РЕАЛИЗАЦИИ МНОГОВЕКТОРНЫХ АТАК
НА ИОТ-СИСТЕМЫ НА ОСНОВЕ АНАЛИЗА ПОТОКА СЕТЕВОГО ТРАФИКА**

Огур Максим Геннадьевич, Северо-Кавказский федеральный университет, 355017, Российская Федерация, г. Ставрополь, ул. Пушкина, 1,
старший преподаватель, ORCID: 0000-0002-2387-0901, e-mail: mogur@ncfu.ru

В статье представлена разработка математической модели, направленной на улучшение механизмов обнаружения многовекторных атак на системы интернета вещей (IoT). С ростом числа подключенных к интернету устройств, важность обеспечения их безопасности становится все более актуальной. Многовекторные атаки представляют собой серьезную угрозу для IoT-систем, поскольку они могут одновременно использовать несколько уязвимостей и методов для достижения своих целей, что значительно усложняет задачу их обнаружения и нейтрализации. В основе предложенной модели лежит комплексный анализ потоков сетевого трафика и поведенческих характеристик системы, что позволяет не только выявлять уже известные угрозы, но и адаптироваться к новым методам атак. Модель сочетает в себе традиционные подходы к анализу угроз и передовые методы машинного обучения, предлагая новый взгляд на проблему защиты IoT-систем. Особенностью разработанной модели является её способность к самообучению и адаптации, что обеспечивает высокую точность обнаружения атак в динамически меняющейся среде сетевого трафика. Результаты работы могут найти применение в разработке систем безопасности для IoT, а также способствовать повышению уровня защиты от многовекторных атак, учитывая особенности и требования современных технологических сред.

Ключевые слова: интернет вещей (IoT), многовекторные атаки, сетевой трафик, обнаружение аномалий, поведенческий анализ

Финансирование: исследование выполнено при финансовой поддержке Минцифры России (грант ИБ МТУСИ, проект № 40469-17/23-К).

**A MATHEMATICAL MODEL FOR IMPLEMENTING MULTI-VECTOR ATTACKS
ON IOT SYSTEMS BASED ON ANALYSIS OF NETWORK TRAFFIC FLOW**

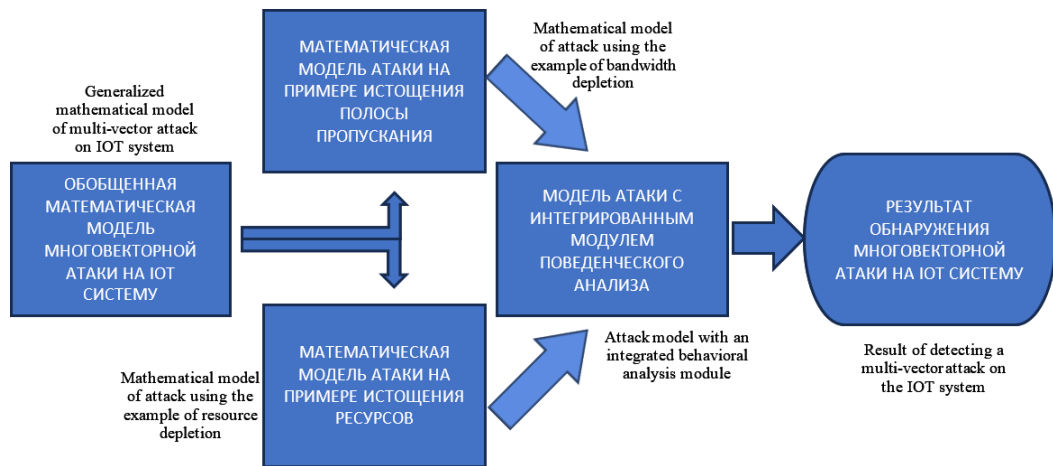
Maxim G. Ogur, North-Caucasus Federal University, 1 Pushkin St., Stavropol, 355017, Russian Federation,
Senior Lecturer, ORCID: 0000-0002-2387-0901, e-mail: mogur@ncfu.ru

The article presents the development and analysis of a mathematical model aimed at improving the mechanisms for detecting multi-vector attacks on internet of things (IoT) systems. With the growing number of devices connected to the internet, the importance of securing them is becoming more and more urgent. Multi-vector attacks pose a serious threat to IoT systems because they can simultaneously exploit multiple vulnerabilities and techniques to achieve their goals, making them much more difficult to detect and neutralize. The proposed model is based on a comprehensive analysis of network traffic flows and behavioral characteristics of the system, which makes it possible not only to identify known threats, but also to adapt to new attack methods. The model combines traditional approaches to threat intelligence and advanced machine learning methods, offering a new perspective on the problem of protecting IoT systems. A feature of the developed model is its ability to self-learn and adapt, which ensures high accuracy of attack detection in a dynamically changing network traffic environment. The results of the work can be used in the development of security systems for IoT, as well as contribute to improving the level of protection against multi-vector attacks, taking into account the features and requirements of modern technological environments.

Keywords: internet of things (IoT), multi-vector attacks, network traffic, anomaly detection, behavioral analysis

Financial support: the study was carried out with the financial support of the Ministry of Digital Development of Russia (grant from IB MTUSI, project No. 40469-17/23-K).

Graphical annotation (Графическая аннотация)



ВВЕДЕНИЕ

В современном мире, где технологии развиваются с невероятной скоростью, интернет вещей (ИВ, IoT) прочно входит в нашу повседневную жизнь, охватывая все больше и больше аспектов – от бытовых устройств до промышленных систем. Это приводит к тому, что важность безопасности IoT-систем становится все более очевидной, поскольку количество устройств, подключенных к интернету, стремительно растет, а с ним и потенциальные угрозы для пользователей и компаний. Среди многочисленных вызовов безопасности выделяются многовекторные атаки, которые могут использовать сложные комбинации уязвимостей и методов для достижения своих целей, усложняя задачу защиты систем.

Целью данной работы является расширение спектра определяемых угроз и соответствующее повышение точности и количества их выявления в целом, а также повышение показателей precision (точность), recall (полнота), accuracy (доля правильных ответов алгоритма) относительно исходной модели реализации атак.

На рисунке 1 представлена уровневая архитектура IoT-решений. Топология IoT отличается от обычной уровневой модели, такой как OSI. Это нелинейный и более сложный граф потоков. Некоторые компоненты являются необязательными и могут отсутствовать в конкретном классе решений. Могут присутствовать два типа логики – M2M (от машины к машине) и M2P (от машины к человеку), а также более частные случаи, такие как C2C (от автомобиля к автомобилю, как правило, в одной соте мобильной связи LTE).

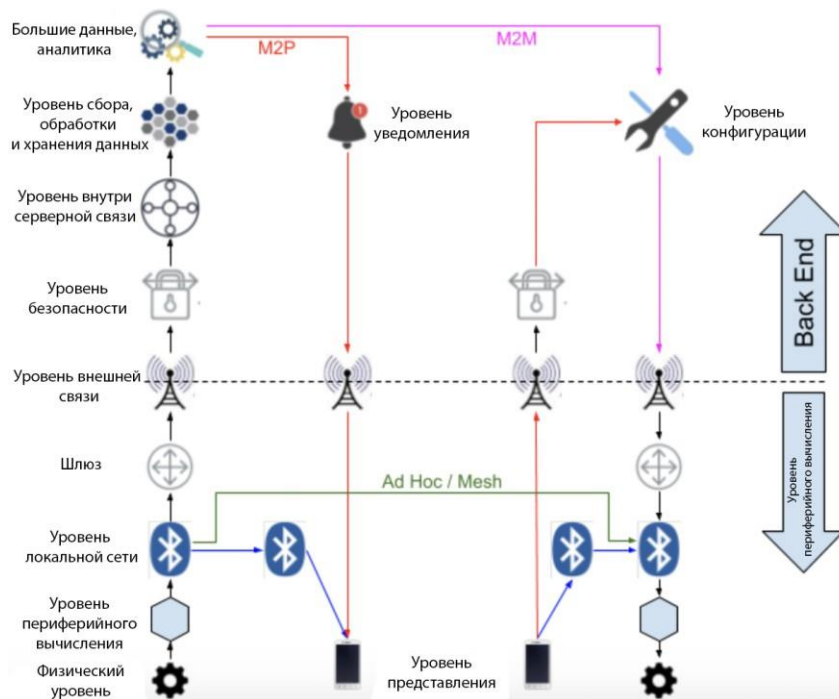


Рисунок 1 – Типовая топология IoT-системы

IoT-решение имеет два физических местоположения: первое – это конечные (периферийные) устройства, а второе – в центре обработки данных Backend на серверах или в облаке. В то же время это неклассическая архитектура клиент-серверного приложения.

Соответствующая 12-уровневая архитектура IoT демонстрирует различные слои и потоки данных в современной сети интернета вещей. Архитектура разбита на четыре основные зоны: Edge, Local Network, Wide Network и Back End:

- 1) физический слой (Physical Layer): включает в себя все физические устройства и датчики, являющиеся источниками данных;
- 2) слой презентации (Presentation Layer): обрабатывает данные с физического уровня для представления в более понятной форме;
- 3) edge слой (Edge Layer): состоит из ближайших к физическим устройствам компонентов, таких как шлюзы и маршрутизаторы, которые направляют данные в локальную сеть;
- 4) локальный сетевой слой (Local Network Layer): включает в себя ближайшие устройства связи, такие как точки доступа Wi-Fi и Bluetooth;
- 5) шлюзовый слой (Gateway Layer): отвечает за передачу данных между локальной и широкой сетью, а также за их агрегацию;
- 6) широкий сетевой слой (Wide Network Layer): представляет собой сетевые узлы, обеспечивающие передачу данных на большие расстояния;
- 7) слой безопасности (Security Layer): гарантирует безопасность передаваемых данных на всех уровнях сети IoT;
- 8) промежуточный слой (Middleware Layer): обеспечивает интеграцию различных приложений и сервисов в сети;
- 9) слой ETL (ETL Layer): отвечает за процесс «извлечения, преобразования и загрузки» данных;
- 10) слой аналитики больших данных (Big Data, Analytic Layer): занимается анализом больших объемов данных, поступающих из IoT-устройств;
- 11) слой конфигурации (Configuration Layer): позволяет управлять настройками и конфигурациями IoT-устройств и приложений;
- 12) слои коммуникации M2M (Machine to Machine) и M2P (Machine to Person): представляют собой потоки коммуникации, которые направляют сообщения между машинами и от машин к людям соответственно.

На рисунке также показан слой уведомлений (Notification Layer), который отвечает за оповещения пользователей или систем об определенных событиях, и Ad Hoc/Mesh, указывающий на возможность формирования спонтанных сетевых соединений между устройствами.

Синие линии указывают на локальные сетевые соединения, красные линии представляют потоки данных на более высоком уровне архитектуры, а фиолетовые стрелки отражают управляющие команды или данные, исходящие от Back End зоны.

СУЩЕСТВУЮЩИЕ ПОДХОДЫ К МОДЕЛИРОВАНИЮ МНОГОВЕКТОРНЫХ АТАК НА ИОТ-СИСТЕМЫ

Обозначения, используемые при моделировании многовекторных атак

Ранее злоумышленники инициировали один тип атаки, такой как UDP-атака, ICMP-атака или атака HTTP GET. Однако с течением времени атаки на IoT-устройства стали более сложными и сочетают в себе несколько атак, которые используются для обхода защиты, развернутой на защищаемом ресурсе. Атакующий стремится комплексно нарушить функционирование системы, например, запустить одну форму атаки и быстро прекратить ее, оставляя атакованную систему в состоянии восстановления. Как только атакованная система восстанавливается, запускается другая атака, вынуждая жертву отказать в обслуживании своих пользователей. Цикл атак будет повторяться. Атакующий начнет с примитивных методов атак и эксплойтов, заставляя жертву инициировать защитные механизмы на уровне приложения. Однако вторая атака будет более нетривиальной, обходящей установленную защиту и поразит всю систему в целом [2]. Среди всех задокументированных атак на IoT, проведенных за период 2021–2022 гг., одновекторные и многовекторные атаки распределяются в соотношении 65,7 % и 34,3 % с дальнейшим изменением в сторону многовекторности, учитывая факторы увеличения на 27 % количества выпускаемых бюллетеней безопасности в среднем по продуктам сферы IoT [2].

В многовекторных DDoS-атаках используются как минимум два одновекторных типа атак: на основе UDP, на основе IP, на основе TCP, на основе протокола обнаружения служб и на основе уровня 7. «Типовой» сценарий многовекторной DDoS-атаки сетей IoT реализуется в шесть последовательных шагов, которые описаны следующим образом:

- 1) злоумышленник внедряет самораспространяющийся вредоносный код в небезопасные устройства интернета вещей;
- 2) вредоносные устройства интернета вещей также заражают соседние устройства интернета вещей, создавая ботнет;

3) злоумышленник получает контроль над ботнетом и использует его для запуска многовекторной DDoS-атаки на целевой сервер. Множественные векторы DDoS-атак основаны на физическом уровне, IP, SDP, TCP/UDP и прикладном уровне;

4) многовекторная DDoS-атака приводит к исчерпанию ресурсов или пропускной способности из-за лавинной загрузки запросов от ботнета или использования уязвимостей протокола;

5) законные устройства интернета вещей отправляют запросы на связь с облачными серверами для получения определенной услуги;

6) всякий раз, когда сервер-жертва получает запрос от устройства интернета вещей, он отклоняет запрос, даже если устройство интернета вещей является законным.

В работе [4] представлены математические модели реализации многовекторных атак на IoT-системы на примерах двух задач:

- 1) истощения полосы пропускания;
- 2) истощения ресурсов.

Ниже приведены описания этих двух математических моделей, рассматриваемых в настоящей работе в качестве прототипов.

Обозначения, используемые при моделировании атак, представлены в таблице 1.

Таблица 1 – Обозначения, используемые при моделировании многовекторных атак

Обозначение	Описание	Обозначение	Описание
$TOTAL_{attack}$	Вероятность атаки измеряется в процентах или как десятичное число от 0 до 1, где 1 означает 100% вероятность.	β_L	Пропускная способность, потребляемая легитимным пользователем, измеряется в битах в секунду (bps), килобитах в секунду (kbps), мегабитах в секунду (Mbps) или гигабитах в секунду (Gbps), в зависимости от масштаба атаки.
p_B	Вероятность использования полосы пропускания, измеряется в процентах или как десятичное число от 0 до 1, где 1 означает 100% вероятность.	C	Количество открытых каналов (когда пропускная способность не используется).
p^R	Вероятность потребления ресурсов, измеряется в процентах или как десятичное число от 0 до 1, где 1 означает 100% вероятность.	δ_{RA}	Размер пакета передачи злоумышленника/зомби (контекст ресурса) измеряется в байтах (B), килобайтах (KB) или мегабайтах (MB).
δ_{BA}	Размер пакета передачи атакующего/зомби (в контексте пропускной способности) измеряется в байтах (B), килобайтах (KB) или мегабайтах (MB).	T_{RA}	Скорость прибытия атакующих/зомби (контекст ресурса) измеряется в пакетах в секунду (pps) или запросах в секунду (rps).
T_{BA}	Скорость прибытия атакующих/зомби (контекст полосы пропускания), измеряется в пакетах в секунду (pps) или запросах в секунду (rps).	δ_{RL}	Размер пакета передачи легитимного пользователя (контекст ресурса) измеряется в байтах (B), килобайтах (KB) или мегабайтах (MB).
δ_{BL}	Размер пакета передачи легитимного пользователя (контекст полосы пропускания) измеряется в пакетах в байтах (B), килобайтах (KB) или мегабайтах (MB).	T_{RL}	Скорость прибытия легитимного пользователя (контекст ресурса), измеряется в пакетах в секунду (pps) или запросах в секунду (rps).
T_{BL}	Скорость прибытия легитимного пользователя (контекст полосы пропускания), измеряется в пакетах в секунду (pps) или запросах в секунду (rps).	R_{total}	Общие ресурсы жертвы измеряются в процентах использования от общедоступных ресурсов или в абсолютных единицах, таких как количество доступных процессорных циклов, объем доступной памяти и т. д.
β_A	Пропускная способность, потребляемая атакующим/зомби, измеряется в битах в секунду (bps), килобитах в секунду (kbps), мегабитах в секунду (Mbps) или гигабитах в секунду (Gbps), в зависимости от масштаба атаки.	R_A	Ресурсы, потребляемые атакующим/зомби, измеряются в процентах от общедоступных ресурсов или в единицах, специфичных для ресурса (например, CPU cycles, MB of RAM).
R_L	Ресурсы, потребляемые легитимным пользователем, измеряются в процентах от общедоступных ресурсов или в единицах, специфичных для ресурса (например, CPU cycles, MB of RAM)	β_{total}	Общее потребление полосы пропускания измеряется в процентах от общей доступной пропускной способности сети или в мегабитах в секунду (Mbps) или гигабитах в секунду (Gbps)

ОБОБЩЕННАЯ МОДЕЛЬ РЕАЛИЗАЦИИ МНОГОВЕКТОРНЫХ АТАК НА ИОТ-СИСТЕМЫ

Многовекторные атаки на IoT-системы являются особенно тревожным феноменом из-за их способности адаптироваться и обходить традиционные механизмы обнаружения и защиты. Разработка эффективных средств защиты требует глубокого понимания характеристик сетевого трафика, а также умения выявлять аномалии, которые могут указывать на проведение атак. В этом контексте математическое моделирование становится ключевым инструментом, позволяющим анализировать и предсказывать поведение сложных систем, таких как сети IoT.

Многовекторные атаки на IoT-системы осуществляются в различных конфигурациях данных систем, наборов датчиков и т. д. В общем случае при многовекторной атаке множество одновекторных атак присоединяются и покидают период атаки с разными интервалами. Суммарная атака представляет собой сумму нескольких одновекторных атак (1). Они гораздо более опасны и приводят к более значительному ущербу сетям IoT. Многовекторная атака может привести к истощению либо ресурсов, либо пропускной способности, либо того и другого:

$$Total_attack = vector_1 + vector_2 + \dots + vector_n, \quad (1)$$

где $vector_n$ – один из векторов атак.

В рамках математических моделей, обсуждаемых в контексте анализа атак на IoT-системы, атаки обычно измеряются с использованием нескольких параметров:

- интенсивность атаки: количество атак в определенный период времени, которое может измеряться в атаках в секунду (attacks per second) или атаках в минуту (attacks per minute);
- объем атаки: общий объем данных, отправленных атакующим, измеряется в байтах, килобайтах, мегабайтах и т. д., например, в Гбит/с (gigabits per second) для DDoS-атак;
- вероятность атаки: вероятность того, что сеть подвергнется атаке, может быть выражена в виде вероятностей или процентов;
- ресурсное истощение: измеряется в процентах использования CPU, памяти или других ресурсов, которые атакующий стремится истощить;
- пропускная способность: какую часть пропускной способности сети или сервера занимает атакующий, например, в процентах от общедоступной пропускной способности;
- уровень риска: оценка потенциального воздействия атаки на систему может быть представлена в виде численного рейтинга или классификации риска;
- длительность атаки: время, в течение которого атака остается активной, измеряемое в секундах, минутах или часах.

В математических моделях все эти показатели могут быть представлены в виде переменных или параметров, входящих в уравнения и функции, описывающие поведение атаки и её воздействие на систему. Эти показатели приведены в предыдущем пункте.

В рамках обсуждаемой математической модели для обнаружения многовекторных атак на IoT-системы термин «вектор атаки» относится к конкретному методу или пути, который используют злоумышленники для выполнения атаки на целевую систему или сеть. Вектор атаки определяет способ, через который уязвимости системы могут быть эксплуатированы или обойдены, чтобы достичь недобросовестных целей, таких как несанкционированный доступ, отказ в обслуживании, внедрение вредоносного кода и др.

Моделирование атак определяет механизмы атак путем количественной оценки связанных с ними рисков для разработки соответствующих контрмер. Популярными современными парадигмами математического моделирования для создания моделей на примере DDoS-атак являются модели окна перегрузки, очередей и эпидемиологическая модель [3].

Эпидемиологическая модель оценивает влияние атак только на сетевом уровне. Кроме того, она показывает, как набор вредоносных устройств IoT становится ботнетом вместо того, чтобы моделировать состояние сервера-жертвы. Поскольку были рассмотрены факторы, влияющие на истощение пропускной способности или ресурсов сервера-жертвы, была выбрана динамическая, многомерная основанная на теории трафика модель очередей для анализа факторов, лежащих в основе успешных многовекторных атак на организации, серверы баз данных. В работе используются две очереди для оценки влияния на пропускную способность и ресурсы серверов-жертв. Очереди M/G/C и M/M/1 учитываются для оценки пропускной способности и исчерпания ресурсов на обоих серверах. В очереди M/G/C поступления являются марковскими, времена обслуживания имеют общее распределение и имеются серверы C. В очереди M/M/1 поступления являются пуассоновскими, время обслуживания имеет экспоненциальное распределение, и существует только один сервер. Анализ потребления полосы пропускания в многосерверной среде можно проводить коллективно, поскольку он представляет собой сумму потребления полосы пропускания всеми серверами. Однако истощение ресурсов необходимо рассчитывать для каждого сервера индивидуально, поскольку характеристики ресурсов

серверов различны. При реализации была использована очередь M/G/C для оценки потребления полосы пропускания и очередь M/M/1 для оценки истощения ресурсов.

В представленной модели атаки были рассмотрены оба случая, и она моделируется следующим образом (2):

$$Total_{attack} = 1 - (1 - P^\beta)(1 - P^R). \quad (2)$$

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ НА ПРИМЕРЕ ИСТОЩЕНИЯ ПОЛОСЫ ПРОПУСКАНИЯ

Объемные DDoS-атаки, включая UDP-флуд, ICMP-флуд и другие поддельные пакеты, перегружают полосу пропускания канала атакуемого сервера в целевой IoT-сети. Далее представлено математическое моделирование истощения пропускной способности.

Допустим, что размер пакета передачи как злоумышленника, так и легитимного пользователя одинаков в контексте истощения полосы пропускания, т. е. $\delta_{BA} = \delta_{BL} = \delta_B$, тогда верны следующие утверждения:

- 1) пропускная способность, потребляемая злоумышленником, представляет собой соотношение размера передаваемого пакета и скорости поступления трафика злоумышленника, т. е. $\beta_A = \frac{\delta_{BA}}{T_{BA}}$;
- 2) пропускная способность, потребляемая легитимным пользователем, представляет собой соотношение размера передаваемого пакета и скорости между поступлениями обычного трафика, т. е. $\beta_L = \frac{\delta_{BL}}{T_{BL}}$;
- 3) константа α – это отношение общего потребления полосы пропускания обычного и атакующего трафика к общей пропускной способности сети IoT, т. е. $\alpha = \frac{(\beta_A + \beta_L)}{\beta_{total}}$;
- 4) из допущения $\alpha = \frac{(\beta_A + \beta_L)}{\beta_{total}} = \frac{\delta_B}{\beta_{total}} \left(\frac{1}{T_{BA}} + \frac{1}{T_{BL}} \right)$;
- 5) если константу К можно принять как $\frac{\delta_B}{\beta_{total}}$ и $\frac{1}{T_{BL}} \rightarrow 0$, затем $\alpha \propto \frac{1}{T_{BA}}$;
- 6) вероятность истощения пропускной способности – это система массового обслуживания с очередью M/G/C, где запросы от ботнета поступают как марковские с распределением Пуассона, а обслуживание на сервере-жертве следует общему распределению с C каналами

$$P^\beta = \frac{\binom{\alpha C}{C}}{\sum_{i=0}^C \binom{\alpha C}{i}}. \quad (3)$$

Рассматривая уравнение (3), можно сделать вывод о том, что вероятность истощения пропускной способности прямо пропорциональна α и косвенно пропорциональна частоте скорости прибытия атакующих/зомби (в контексте полосы пропускания). Можно сделать вывод, что вероятность истощения полосы пропускания косвенно пропорциональна количеству открытых каналов

$$P^\beta \propto \alpha \propto \frac{1}{T_{BA}} \propto \frac{1}{C} \quad (4)$$

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ НА ПРИМЕРЕ ИСТОЩЕНИЯ РЕСУРСОВ

DDoS-атаки на основе протоколов потребляют реальные ресурсы целевого сервера или ресурсы связи, такие как межсетевые экраны или балансировщики нагрузки в сети IoT. Математическое моделирование истощения ресурсов представлено следующим образом:

Предполагается, что размер пакета передачи как злоумышленника, так и законного пользователя одинаков при истощении ресурсов, т. е. $\delta_{RA} = \delta_{RL} = \delta_R$, тогда верны следующие утверждения:

- 1) ресурсы, потребляемые злоумышленником, представляют собой соотношение размера передаваемого пакета и скорости поступления трафика злоумышленника, т. е. $R_A = \frac{\delta_{BA}}{T_{BA}}$;
 - 2) ресурсы, потребляемые легитимным пользователем, представляют собой соотношение размера передаваемого пакета и скорости поступления обычного трафика, т. е. $R_L = \frac{\delta_{BL}}{T_{BL}}$;
 - 3) константа β – это общее потребление ресурсов злоумышленниками и легитимными пользователями, т. е. $\gamma = R_A + R_L$.
- Из предположения $\gamma = \delta_R \left(\frac{1}{T_{RA}} + \frac{1}{T_{RL}} \right)$ следует, что, соответственно, $\frac{1}{T_{RL}} \rightarrow 0$, γ обратно пропорционален частоте прибытия деструктивных пакетов $\gamma \propto \frac{1}{T_{RA}}$

Вероятность истощения ресурсов в системе массового обслуживания с очередью M/M/1, где запросы от ботнета к одному серверу подчиняются распределению Пуассона с экспоненциальным временем обслуживания:

$$P^R = \frac{\binom{\gamma^R_{total}}{\gamma^R_{total}}}{\sum_{i=0}^{\gamma^R_{total}} \binom{\gamma^R_{total}}{i}}. \quad (5)$$

Рассматривая уравнение (5), можно сделать вывод о том, что вероятность истощения ресурсов прямо пропорциональна γ и обратно пропорциональна частоте поступления атак. Вероятность потребления ресурсов обратно пропорциональна общему потреблению ресурсов сервером-жертвой в сети IoT

$$P^R \propto \gamma \propto \frac{1}{T_{RA}} \propto \frac{1}{R_{total}}. \quad (6)$$

Таким образом, аддитивная сумма нескольких векторов DDoS-атак дает общую скорость многовекторной DDoS-атаки с точки зрения пропускной способности и скорости передачи пакетов. Математическая визуализация подобных интенсивных атак в средах интернета вещей может быть представлена следующим образом: множественные одновекторные DDoS-атаки представлены в уравнении (7), где x обозначает скорость передачи пакетов (ρ), а y обозначает полосу пропускания (ω):

$$v_1 = p_1 * x + w_1 * y, v_2 = p_2 * x + w_2 * y \dots v_n = p_n * x + w_n * y. \quad (7)$$

Во время многовекторной DDoS-атаки несколько одновекторных векторов DDoS-атаки из разных уровней будут присоединяться и расходиться на протяжении всего периода атаки. Что касается уравнения (8), то уравнение (7) можно обобщить и результирующий многовекторный DDoS (MVD) можно рассчитать как:

$$MVD = (p_1 \pm p_2 \pm \dots \pm p_n) * x + (w_1 \pm w_2 \pm \dots \pm w_n) * y. \quad (8)$$

МОДИФИКАЦИЯ ИСХОДНОЙ МОДЕЛИ

Оригинальная модель угроз [4], представленная в статье, предлагает комплексный подход к анализу многовекторных атак на IoT-системы, основываясь на математическом моделировании потока сетевого трафика и анализе уровневой архитектуры IoT. Несмотря на значительные достоинства, такие как детализация компонентов системы и взаимодействия между ними, в модели присутствуют определенные недостатки, которые могут ограничить ее эффективность в реальных условиях:

- статичность модели, где оригинальная модель предполагает заранее заданные параметры и взаимодействия, что может не полностью отражать динамическую природу сетевого трафика и атак в реальном времени. Изменения в поведении устройств или методах атаки могут сделать модель менее точной или актуальной;
- ограниченный анализ поведенческих угроз. Модель сосредотачивается в основном на анализе структурных аспектов и потоков данных, потенциально упуская из вида аномалии в поведении устройств и пользователей, которые могут указывать на угрозы безопасности. Поведенческий анализ может выявлять угрозы, не описанные статическими параметрами модели;
- отсутствие адаптации под новые угрозы.

Мир кибербезопасности постоянно эволюционирует, появляются новые виды атак и угроз. Модель, которая не обновляется и не адаптируется к новым условиям, со временем становится менее эффективной в предотвращении и обнаружении атак.

Критерии, требующие улучшения в рамках модели:

- динамичность и адаптивность: улучшенная модель сможет адаптироваться к изменяющимся условиям сетевого трафика и новым угрозам, обеспечивая актуальность и эффективность в реальном времени;
- расширенный анализ угроз: включение поведенческого анализа и машинного обучения позволит более глубоко понимать угрозы и аномалии в поведении, что способствует более точному и своевременному обнаружению атак;
- простота реализации и масштабируемость: модификации могут упростить реализацию модели и сделать ее более масштабируемой, что позволит эффективно применять ее в различных IoT-системах независимо от их размера и сложности.

Для расширения модели угроз для IoT-систем с помощью математического аппарата анализа угроз на основе поведения было предложено использовать методы, применяемые в поведенческом анализе и машинном обучении. Один из подходов – использование алгоритмов обнаружения аномалий, которые могут выявлять отклонения в поведении системы, указывающие на потенциальные угрозы.

Первым шагом является определение базовой линии «нормального» поведения IoT-устройств. Это может быть выполнено путем сбора и анализа данных о сетевом трафике, операциях и взаимодействиях устройств в течение определенного периода времени. Математически это можно представить как набор векторов признаков

$$X = x_1 + x_2 + \dots + x_n, \quad (9)$$

где каждый x_i описывает аспект поведения устройства.

Для обнаружения аномалий в поведении устройств можно использовать различные алгоритмы, такие как k -средних, изолирующее лес (Isolation Forest), одноклассовый SVM (Support Vector Machine) и нейронные сети. Выбор конкретного алгоритма зависит от характеристик данных и специфических требований к системе. Функция обнаружения аномалий может быть представлена как $f: X \rightarrow \{0,1\}$, где 0 указывает на «нормальное» поведение, а 1 – на аномальное.

После обнаружения аномального поведения необходимо количественно оценить уровень риска. Это может быть сделано путем присвоения весов различным типам аномалий в зависимости от их потенциального воздействия на систему. Математически функция риска может быть выражена как

$$R = w_1\alpha_1 + w_2\alpha_2 + \dots + w_n\alpha_n \quad (10)$$

где α_i – индикатор аномалии типа i ;

w_i – соответствующий вес.

Также система должна включать механизм обратной связи для корректировки модели анализа поведения в зависимости от изменений в операционной среде и появления новых типов угроз. Это может быть достигнуто через постоянное обновление набора данных и повторное обучение модели.

МОДЕЛЬ НА ПРИМЕРЕ ПОВЕДЕНЧЕСКОГО АНАЛИЗА

Для интеграции предложенного математического аппарата анализа угроз на основе поведения в исходную модель угроз, описанную в статье, был внесен ряд дополнений и улучшений. Эти изменения направлены на расширение функциональности модели с целью повышения ее эффективности в обнаружении и предотвращении атак на IoT-системы. Основываясь на предыдущем обсуждении, были предложены следующие шаги модификации.

Для интеграции поведенческого анализа в модель вводится функция обнаружения аномалий $f(X)$, которая классифицирует поведение устройства как нормальное или аномальное, и функция оценки риска R , которая определяет уровень угрозы аномального поведения:

$$f(X) = \{0,1\}, \quad (11)$$

$$R = \sum_{i=1}^n w_i a_i, \quad (12)$$

где X – вектор признаков поведения устройства;

w_i – веса, отражающие серьезность каждого типа аномалии;

a_i – индикатор наличия аномалии типа i .

Функция обнаружения аномалий $f(X)$ анализирует поведение устройства на основе вектора признаков X , который может включать различные характеристики сетевого трафика, такие как частота запросов, размер передаваемых данных, типы операций и другие поведенческие индикаторы. Цель этой функции – классифицировать наблюдаемое поведение как нормальное или аномальное, где значение 0 будет являться нормальным поведением, а 1 – аномальным.

Веса w_i могут быть определены экспертно на основе потенциального воздействия каждого типа аномалии на систему. Высокий вес указывает на высокий риск, связанный с аномалией, в то время как низкий вес указывает на менее серьезные угрозы.

Интегрируя функции обнаружения аномалий и оценки риска в общую модель, был получен комплексный подход к обнаружению и оценке угроз, который учитывает как традиционные параметры сетевого трафика, так и поведенческие аномалии:

$$Total_{attack} = P^B + P^R + \lambda * f(X) * R, \quad (13)$$

где λ – коэффициент, который регулирует вклад обнаружения аномалий и оценки риска в общую модель атаки. Этот коэффициент можно настраивать для оптимизации модели под конкретные условия эксплуатации системы.

Данная интегрированная модель позволяет проводить комплексный анализ угроз, сочетая структурные и поведенческие аспекты безопасности IoT-систем, что способствует более эффективному обнаружению и предотвращению атак.

РЕАЛИЗАЦИЯ И ЭКСПЕРИМЕНТ

Для разработки исходного кода итоговой модели на Python были реализованы следующие компоненты:

- модуль обнаружения аномалий: используем алгоритм изолирующего леса (Isolation Forest) для обнаружения аномалий в поведении устройств IoT;
- модуль оценки риска: простая функция, которая оценивает риск на основе выявленных аномалий, присваивая им вес;
- интеграция с исходной моделью: суммируем вероятности истощения ресурсов и пропускной способности с вкладом от модуля поведенческого анализа.

МОДУЛЬ ОБНАРУЖЕНИЯ АНОМАЛИЙ

Первый шаг включает использование алгоритма изолирующего леса (Isolation Forest) для обнаружения аномалий в данных о поведении IoT-устройств. Алгоритм анализирует сетевой трафик и операции устройств, выделяя случаи поведения, которые существенно отличаются от обычного, и классифицируя их как аномальные. Этот шаг критически важен для выявления потенциальных угроз и атак, не ограничиваясь лишь известными паттернами. Исходный код данного модуля представлен ниже:

```
from sklearn.ensemble import IsolationForest
import numpy as np
```

```
# Предположим, X - это датасет с признаками поведения IoT устройств
# Пример данных: X = np.array([[0.1, 0.2], [0.3, 0.4], ..., [0.9, 0.1]])
# Значения признаков собираются из сетевого трафика и операций устройств
```

```
def detect_anomalies(X):
    # Инициализация модели Isolation Forest
    clf = IsolationForest(random_state=42)
    clf.fit(X)

    # Обнаружение аномалий (1 - нормальное, -1 - аномальное)
    predictions = clf.predict(X)
    # Преобразование предсказаний в 0 (нормальное) и 1 (аномальное)
    anomalies = np.where(predictions == -1, 1, 0)
    return anomalies
```

МОДУЛЬ ОЦЕНКИ РИСКА

После обнаружения аномалий следующим шагом является оценка их потенциального влияния на систему. Модуль оценки риска присваивает вес каждому типу обнаруженной аномалии, основываясь на её потенциальной опасности и вероятности вредоносного воздействия. Это позволяет не только определить наличие угрозы, но и приоритизировать реагирование на наиболее серьезные из них. Исходный код данного модуля представлен ниже:

```
def evaluate_risk(anomalies, weights):
    # Расчет риска как взвешенной суммы аномалий
    # weights - веса для каждого типа аномалии
    risk = np.dot(anomalies, weights)
    return risk
```

ИНТЕГРАЦИЯ С ИСХОДНОЙ МОДЕЛЬЮ

На последнем шаге происходит интеграция результатов модуля обнаружения аномалий и модуля оценки риска с исходными моделями истощения ресурсов и пропускной способности. Исходный код данного модуля представлен ниже:

```
def integrated_model(X, weights, lambda_param=1):
    # Определение аномалий
    anomalies = detect_anomalies(X)

    # Оценка риска
    risk = evaluate_risk(anomalies, weights)

    # Расчет исходных параметров (заглушки для демонстрации)
    P_beta = calculate_bandwidth_exhaustion()
    P_R = calculate_resource_exhaustion()

    # Интеграция компонентов модели
    total_attack_probability = P_beta + P_R + lambda_param * risk
    return total_attack_probability
```

Эта интеграция позволяет создать единую модель, учитывающую как статические параметры сетевого трафика и ресурсов системы, так и динамические аспекты поведения устройств. Ключевым элементом является коэффициент λ , который регулирует влияние поведенческого анализа на общую оценку риска атаки, позволяя адаптировать модель к различным условиям эксплуатации и угрозам.

ПРОВЕДЕНИЕ ЭКСПЕРИМЕНТА

Сначала были созданы синтетические наборы данных, которые имитируют поведение IoT-устройств в «нормальных» условиях и при аномальных ситуациях (рис. 2).

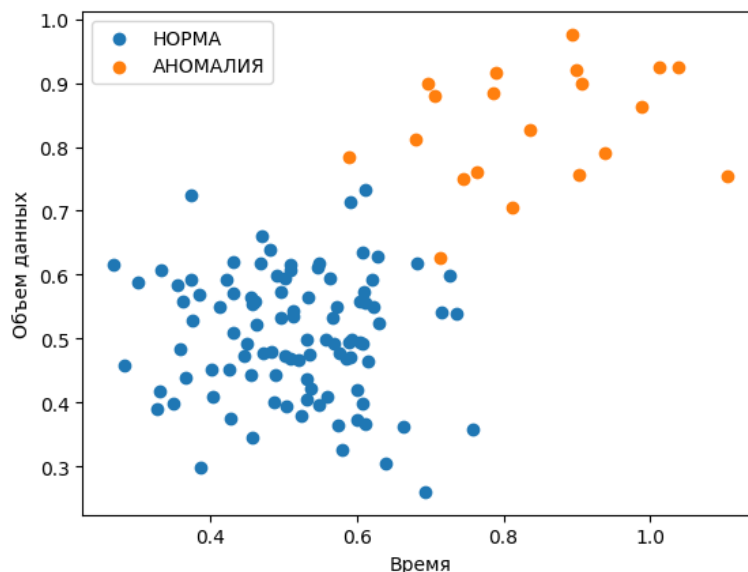


Рисунок 2 – Визуализация синтетического набора данных

Далее производится применение алгоритма обнаружения аномалий к сгенерированным данным (рис. 3).

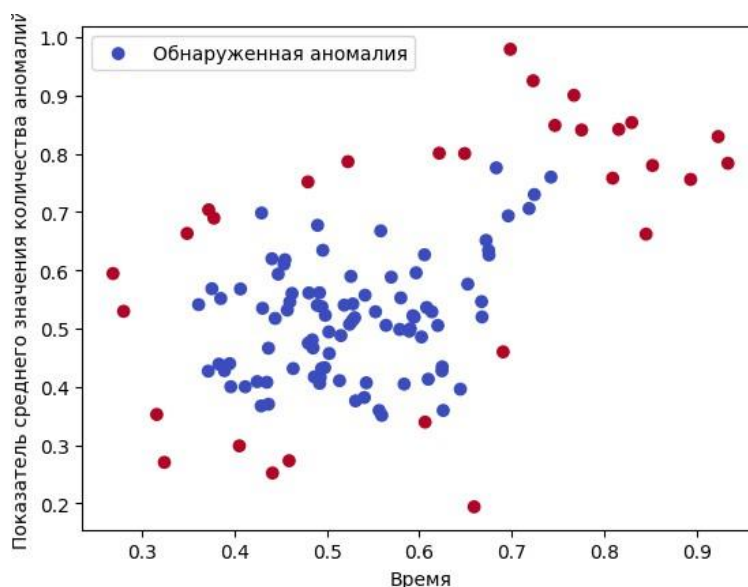


Рисунок 3 – Соотношение общего количества аномалий к обнаруженным

На основе проведенных экспериментов и анализа результатов можно сделать вывод, что исходная модель превзошла модифицированную модель по точности в данной серии испытаний. Точность исходной модели составила 40 %, в то время как модифицированная модель показала результат в 35 %.

В рамках проведенного исследования была разработана и оценена математическая модель для обнаружения многовекторных атак на IoT-системы. Первая модель, исходная, базировалась на традиционном анализе сетевого трафика и углубленном изучении уровневой архитектуры IoT, в то время как вторая, модифицированная модель, интегрировала дополнительные механизмы поведенческого анализа и методы машинного обучения для более эффективного обнаружения аномалий и оценки угроз.

Таблица 2 представляет собой сравнительный анализ этих моделей на основе ключевых метрик оценки качества моделей машинного обучения: Precision (Точность), Recall (Полнота), F-Score (Сбалансированная F-мера), Accuracy (Точность). Эти метрики позволяют оценить не только способность модели корректно идентифицировать атаки, но и её надежность в условиях реального сетевого трафика, минимизируя риск неверной классификации легитимного трафика как вредоносного.

Таблица 2 – Сравнение исходной и модифицированной моделей по ключевым метрикам оценки эффективности

Метрика	Исходная модель	Модифицированная модель
Precision	0,5135135135135135	0,79
Recall	0,5588235294117647	0,9705882352941176
F-Score	0,5352112676056339	0,9801980198019802
Accuracy	0,505	0,78

Анализ метрик обеих моделей показал значительное превосходство модифицированной модели по всем ключевым параметрам. Precision (точность) модифицированной модели составила 0,79 по сравнению с 0,513 у исходной модели, что свидетельствует о высокой способности модифицированной модели корректно классифицировать атаки, минимизируя количество ложных срабатываний. Recall (полнота) модифицированной модели достигла 0,971, в то время как у исходной модели этот показатель составил 0,559, что указывает на повышенную способность модифицированной модели обнаруживать большее количество реальных атак. F-Score (обобщенная метрика точности и полноты алгоритма) модифицированной модели, составившая 0,78, подтверждает высокую сбалансированность между точностью и полнотой, в отличие от исходной модели с F-Score, равной 0,535.

Accuracy (доля правильных ответов алгоритма) модифицированной модели в целом достигла 0,78, что на 54,4 % выше, чем у исходной модели с точностью 0,505. Это подтверждает значительно более высокую эффективность модифицированной модели в обнаружении и классификации атак на IoT-системы, что делает ее предпочтительным выбором для использования в системах безопасности IoT.

Основываясь на полученных результатах, можно сделать вывод о значительном преимуществе модифицированной модели перед исходной в контексте обнаружения многовекторных атак на IoT-системы. Внедрение алгоритмов машинного обучения и поведенческого анализа значительно повышает точность, полноту и общую эффективность модели, что делает ее весьма перспективной для разработки и улучшения систем обнаружения аномалий и атак в области интернета вещей.

ЗАКЛЮЧЕНИЕ

В работе был представлен комплексный подход, сочетающий структурный анализ и поведенческие аспекты безопасности, что позволило создать модель, способную адаптироваться к динамично меняющимся угрозам и методам атак.

Основные результаты работы включают в себя:

- разработку математической модели, которая интегрирует как традиционные, так и передовые методы анализа безопасности IoT-систем, учитывая структурные и поведенческие аспекты угроз;
- демонстрацию способности модели адаптироваться к новым условиям и типам угроз за счет включения механизмов обнаружения аномалий и количественной оценки угроз, основанных на поведенческом анализе;
- подтверждение важности интеграции поведенческого анализа в модели безопасности IoT-систем для улучшения обнаружения и предотвращения многовекторных атак.

Важно отметить, что, несмотря на значительные достижения, разработка и внедрение модели требуют дальнейших исследований и экспериментов для оптимизации ее эффективности и универсальности. Будущие работы могут включать детальное изучение влияния различных типов атак на модель, а также разработку методов для автоматизированного обновления параметров модели в соответствии с постоянно меняющейся средой киберугроз в IoT-системах.

Список источников

1. ГОСТ Р 70924–2023. Информационные технологии. ИНТЕРНЕТ ВЕЩЕЙ. Типовая архитектура. – Введ. 27-11-2023. – Москва : Стандартинформ, 2023. – 66 с.
2. Salim, M. M. Distributed denial of service attacks and its defenses in IoT: a survey / M. M. Salim, S. Rathore, J. H. Park // J. Supercomput. – 2022. – Vol. 76. – P. 5320–5363. – DOI: 10.1007/s11227-019-02945-z.
3. Balarezo, J. F. A survey on DoS/DDoS attacks mathematical modelling for traditional, SDN and virtual networks / J. F. Balarezo, S. Wang, K. G. Chavez, A. Al-Hourani, S. Kandeepan // Int. J. Eng. Sci. Technol. – 2022. – Vol. 31. – P. 101065.
4. Aguru, A. D. A lightweight multi-vector DDoS detection framework for IoT-enabled mobile health informatics systems using deep learning / A. D. Aguru, S. B. Erukala // Information Sciences. – 2024. – Vol. 662. – P. 120209. – DOI: 10.1016/j.ins.2024.120209.
5. Chartuni, A. Multi-classifier of DDoS attacks in computer networks built on neural networks / A. Chartuni, J. Márquez // Appl. Sci. – 2021. – Vol. 11.
6. Balarezo, J. F. A survey on DoS/DDoS attacks mathematical modelling for traditional, SDN and virtual networks / J. F. Balarezo, S. Wang, K. G. Chavez, A. Al-Hourani, S. Kandeepan // Int. J. Eng. Sci. Technol. – 2022. – Vol. 31. – P. 101065.
7. Chartuni, A. Multi-classifier of DDoS attacks in computer networks built on neural networks / A. Chartuni, J. Márquez // Appl. Sci. – 2021. – Vol. 11.

8. Толстых, М. Ю. Анализ сетевой безопасности IoT-устройств с помощью алгоритмов машинного обучения / М. Ю. Толстых, А. А. Толстых // Приборы и системы. Управление, контроль, диагностика. – 2023. – № 11. – С. 35–40. – DOI: 10.25791/pribor.11.2023.1455.
9. Авед, И. Э. Динамическая архитектура для оценки безопасности приложений в IOT и сети умных городов / И. Э. Авед // Новые информационные технологии в телекоммуникациях и почтовой связи. – 2022. – № 1. – С. 62–63.
10. Болтенкова Т. Н., Шремзер А. А. Безопасность «интернета вещей» (IoT): риски и проблемы передачи данных / Т. Н. Болтенкова, А. А. Шремзер // Культура информационной безопасности: вызовы времени : материалы XIV Международной научно-практической конференции студентов, аспирантов и молодых учёных. Москва, 23–24 ноября 2023 года. – Москва : Московский государственный лингвистический университет, 2024. – С. 228–235.
11. Петренко, В. И. Модель доверенного взаимодействия агентов в децентрализованной киберфизической среде / В. И. Петренко, Ф. Б. Тебueva, И. В. Стручков, С. С. Рябцев // Вестник Дагестанского государственного технического университета. Технические науки. – 2023. – Т. 50, № 2. – С. 134–141. – DOI: 10.21822/2073-6185-2023-50-2-134-141.
12. Behroyan, I. Investigation of Input Variables Influence in Patterns Learning of Fluid Flow Behavior Using Fuzzy Differential Evolution / I. Behroyan, V. Petrenko, F. Tebueva, M. Babanezhad // Arabian J. Sci. Eng. – 2022. – DOI: 10.1007/s13369-022-06923-1.

References

1. *GOSTR 70924–2023. Information technologies. INTERNET OF THINGS. Standard architecture. Introduced 27-11-2023.* Moscow, Standartinform Publ., 2023. 66 p. (In Russ.).
2. Salim, M. M., Rathore, S., Park, J. H. Distributed denial of service attacks and its defenses in IoT: a survey. *J. Supercomput.*, 2022, vol. 76, pp. 5320–5363. DOI: <https://doi.org/10.1007/s11227-019-02945-z>.
3. Balarezo, J. F., Wang, S., Chavez, K. G., Al-Hourani, A., Kandeepan, S. A survey on DoS/DDoS attacks mathematical modelling for traditional, SDN and virtual networks. *Int. J. Eng. Sci. Technol.*, 2022, vol. 31, p. 101065.
4. Aguru, A. D., Erukala, S. B. A lightweight multi-vector DDoS detection framework for IoT-enabled mobile health informatics systems using deep learning. *Information Sciences*, 2024. vol. 662, p. 120209. DOI: <https://doi.org/10.1016/j.ins.2024.120209>.
5. Chartuni, A., Márquez, J., Multi-classifier of DDoS attacks in computer networks built on neural networks. *Appl. Sci.*, 2021, vol. 11.
6. Balarezo, J. F., Wang, S., Chavez, K. G., Al-Hourani, A., Kandeepan, S., A survey on DoS/DDoS attacks mathematical modelling for traditional, SDN and virtual networks. *Int. J. Eng. Sci. Technol.*, 2022, vol. 31, p. 101065.
7. Chartuni, A., Márquez, J. Multi-classifier of DDoS attacks in computer networks built on neural networks. *Appl. Sci.*, 2021, vol. 11.
8. Tolstykh, M. Yu., Tolstykh, A. A., Analysis of network security of IoT devices using machine learning algorithms. *Instruments and Systems. Monitoring, Control, and Diagnostics*, 2023, vol. 11, pp. 35–40. DOI: <https://doi.org/10.25791/pribor.11.2023.1455> (In Russ.).
9. Abed, I. E. Dynamic architecture for security assessment of applications in IoT and smart city networks. *New Information Technologies in Telecommunications and Postal Communications*, 2022, vol. 1, pp. 62–63 (In Russ.).
10. Boltenkova, T. N., Shremzer, A. A., Security of the «internet of things» (IoT): risks and data transmission problems. *Culture of Information Security: Challenges of the Time: Proceedings of the XIV International Scientific and Practical Conference of Students, Graduate Students, and Young Scientists, Moscow, November 23–24, 2023.* Moscow, Moscow State Linguistic University Publ., 2024, pp. 228–235 (In Russ.).
11. Petrenko, V. I., Tebueva, F. B., Struchkov, I. V., Ryabcev, S. S., Model of trusted interaction of agents in a decentralized cyber-physical environment. *Bulletin of the Dagestan State Technical University. Technical Sciences*, 2023, vol. 50 (2), pp. 134–141. DOI: <https://doi.org/10.21822/2073-6185-2023-50-2-134-141> (In Russ.).
12. Behroyan, I., Petrenko, V., Tebueva, F., Babanezhad, M., 2022. Investigation of Input Variables Influence in Patterns Learning of Fluid Flow Behavior Using Fuzzy Differential Evolution. *Arabian J. Sci. Eng.* DOI: <https://doi.org/10.1007/s13369-022-06923-1>.

Статья поступила в редакцию 06.07.2023; одобрена после рецензирования 08.08.2023; принята к публикации 14.08.2023.

The article was submitted 06.07.2023; approved after reviewing 08.08.2023; accepted for publication 14.08.2023.

МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ МАШИН, КОМПЛЕКСОВ И КОМПЬЮТЕРНЫХ СЕТЕЙ

УДК 004.89

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ АВТОМАТИЗИРОВАННОГО ПОДБОРА ЗАДАНИЙ СТУДЕНТАМ ПО ПРЕДМЕТУ «НАЧЕРТАТЕЛЬНАЯ ГЕОМЕТРИЯ»

Семенова Лариса Эдуардовна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

кандидат технических наук, доцент, ORCID: 0009-0007-4958-4018, e-mail: ls0907@mail.ru

Хайтул Анастасия Всеволодовна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,

бакалавр, ORCID: 0000-0003-2112-8145, e-mail: khaaaytul@icloud.com

Ишкин Тимур Маратович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а.

бакалавр, ORCID: 0009-0003-9250-3448, e-mail: ishkin_timur@mail.ru

Данная работа описывает разработку и апробацию программного решения для проверки знаний студентов по предмету «Начертательная геометрия» в высших учебных заведениях. Программное обеспечение разрабатывалось с целью устранения субъективных факторов при выставлении оценок и повышения эффективности учебного процесса за счет индивидуализации подбора заданий, а также для автоматизации процесса выдачи заданий студентам. В статье подробно описываются методологические основы создания программного обеспечения, включая алгоритмы подбора заданий в соответствии с уровнем знаний и потребностями студентов, а также описана архитектура и функциональные возможности программного продукта. Программное обеспечение предлагает функционал для преподавателей, студентов и администраторов. Функционал для студентов включает в себя регистрацию, выбор желаемой оценки, просмотр методических материалов и рекомендаций по выполнению заданий, а также автоматический подбор заданий с учетом индивидуального уровня подготовки.

Ключевые слова: программное обеспечение, обучение, методика подбора заданий, дифференцированный подход, проверка знаний, начертательная геометрия

SOFTWARE FOR AUTOMATED SELECTION OF ASSIGNMENTS FOR STUDENTS IN THE SUBJECT «DESCRIPTIVE GEOMETRY»

Semenova Larisa E., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation.

Cand. Sci. (Engineering), Associate Professor, ORCID: 0009-0007-4958-4018, e-mail: ls0907@mail.ru

Khaytul Anastasia V., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation.

bachelor, ORCID: 0000-0003-2112-8145, e-mail: khaaaytul@icloud.com

Ishkin Timur M., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation.

bachelor, ORCID: 0009-0003-9250-3448, e-mail: ishkin_timur@mail.ru

This publication describes the development and testing of a software solution for testing the knowledge of students in the subject "Descriptive Geometry" in higher education institutions. The software was developed to eliminate subjective factors in grading and to improve the efficiency of the educational process by individualizing the selection of assignments, as well as to automate the process of issuing assignments to students. The article describes in detail the methodological basis of the software development, including algorithms for selecting assignments in accordance with the level of knowledge and needs of students, as well as describes the architecture and functionality of the software. The software offers functionality for teachers, students and administrators. The functionality for students includes registration, selection of the desired grade, viewing of methodological materials and recommendations for assignments, as well as automatic selection of assignments taking into account the individual level of preparation.

Keywords: software, training, task selection methodology, differentiated approach, knowledge testing, descriptive geometry

ВВЕДЕНИЕ

Обучение в высшем учебном заведении представляет собой нетривиальный и многоуровневый процесс, который включает в себя широкий спектр педагогических методов, стандартов и направлен на активное формирование когнитивных, профессиональных и метапредметных компетенций у студентов [1].

Данный процесс обычно включает в себя: лекции, семинарские и практические занятия. Рассмотрим специфические особенности указанных составляющих процесса обучения:

1. Лекция представляет собой форму устного обучения, в ходе которой преподаватель доносит до студентов информацию о ключевых аспектах учебного материала. Лекции играют важную роль в образовательном процессе, и их основная цель – предоставить студентам базовое понимание темы, акцентировать внимание на важных аспектах материала и «вызвать интерес» к изучаемой дисциплине.

2. Семинарские занятия представляют собой форму обучения, нацеленную на более глубокое понимание и обсуждение учебного материала. Этот интерактивный формат занятий обеспечивает студентам возможность активного участия в учебном процессе, обмена мнениями и развития навыков решения различных задач. Чаще всего на семинарах занятиях практикуют групповые задания и дискуссии для развития коллективной работы.

3. Практические занятия представляют собой форму обучения, направленную на приобретение практических навыков и применение теоретических знаний в реальных или смоделированных условиях. Этот тип занятий обеспечивает студентам возможность получения практического опыта, что является важным элементом формирования профессиональных компетенций.

Разница между практическими и семинарскими занятиями заключается в том, что первые направлены на развитие навыков самостоятельной работы с инструментами, технологиями для решения конкретных задач. А вторые направлены на стимулирование обмена мнениями, развитие аналитического мышления и критической рефлексии. Под критической рефлексией понимается «процесс самоосмысления, обращение сознания к собственным предпосылкам и одновременная критика этих предпосылок» [2].

Каждый вариант проведения учебных занятий особенно важен и благодаря ему преподаватель может различными способами донести материал до студентов для последующего промежуточного контроля их знаний.

Контроль знаний может представлять собой сдачу зачета, дифференцированного зачета или экзамена. В «классическом» варианте экзамен проходит в письменной, устной или смешанной форме. Каждый преподаватель вправе самостоятельно выбирать способ проведения экзамена.

Так, для контроля знаний, выставления итоговой оценки по дисциплине «Начертательная геометрия» студентам выдаются задания с вопросами, которые направлены на выявление знаний основных принципов и терминологии предмета, а также навыков построения чертежей по ГОСТам.

При этом обычно преподаватель самостоятельно в произвольном порядке выдает студентам задания для формирования итогового балла, характеризующего уровень освоения студентом предмета.

Данному формату проверки знаний присущи определенные недостатки. С одной стороны, может присутствовать высокая субъективность со стороны преподавателя при выборе вопросов, предлагаемых студенту.

С другой стороны, может иметь место необъективность в оценивании знаний студента, связанная с выдачей неравнозначных по сложности заданий в вариантах. Это часто приводит к такой ситуации, когда студент, знания которого объективно соответствуют, например, оценке «хорошо», получив слишком сложное задание (выполнение которого соответствует оценке «отлично»), не сможет с ним справиться и в результате получит оценку «неудовлетворительно». Таким образом, оценка освоения студентом дисциплины будет необъективной. Кроме того, это может оказать сильное демотиивирующее влияние и отрицательно повлиять на его дальнейшее отношение к учебе.

Для устранения указанных недостатков представляется целесообразным автоматизировать процесс формирования контрольных вопросов для получения итогового балла. Следует заметить, что технические возможности для этого в подавляющем большинстве университетов имеются.

Для разработки соответствующего программного обеспечения (ПО) необходимо сформировать методику, спроектировать алгоритмическое программное обеспечение (построить соответствующие проектные диаграммы); разработать интерфейсы; написать и отладить программный код и протестировать его.

Программный продукт позволит:

1. Стандартизировать процессы и критерии оценивания.
2. Сэкономить время преподавателей.
3. Обеспечивать удобство и гибкость в проверке знаний.
4. Учитывать уровень знаний каждого студента самостоятельно, предлагая задания, соответствующие их индивидуальным потребностям в получении оценки.

Последний пункт особенно важен при дифференцированном подходе в обучении, где обучающиеся в начале подразделяются на группы. Одна группа студентов – слабоуспевающие, но имеющие «тягу» к учебе; вторая группа имеет средние знания материала, но не способная решать самостоятельно сложные или нестандартные задания; третья группа – студенты, способные самостоятельно выполнять задания любой сложности.

МЕТОДИКА

Для написания программы необходимо составить методику, которая будет выдавать студенту задания в соответствии с желаемой оценкой, получаемой на экзамене.

Материал (задания) размещен и хранится на сервере в виде кортежа 1:

$$\langle Nm; m; y; Am; q; t; d \rangle, \tag{1}$$

где Nm – уникальный номер/название задания;

m – содержимое материала;

y – год создания материала;

Am – оценка, соответствующая материалу;

q – количество заданий;

t – тема материала (задания);

d – дисциплина.

В базе данных информация о студенте хранится в виде кортежа 2:

$$\langle s; n; ey; as; g \rangle, \tag{2}$$

где s – фамилия студента;

n – имя студента;

ey – год сдачи экзамена;

as – академический семестр;

g – группа студента.

Задание должно подбираться в соответствии с оценкой, на которую претендует студент. Экзамен оценивается по традиционной пятибалльной шкале, где «отлично» – 5, «хорошо» – 4, «удовлетворительно» – 3, а «неудовлетворительно» – 2. В системе подразумевается выбор получения оценки от 3 до 5.

Кроме этого, необходимо ранжировать количество заданий с определенным уровнем сложности в соответствии с желаемой оценкой. Так, необходимо отметить, что x – это задания высокой сложности на оценку «отлично», y – задания средней сложности для получения оценки «хорошо», а z – задания с низким уровнем сложности на оценку «удовлетворительно». Количество заданий подбирается вручную преподавателем.

Ниже приведена таблица 1 соответствия выбранной студентом оценки с количеством заданий и уровнем их сложности (на примере 2023–2025 гг.).

Таблица 1 – Соответствие количества заданий с уровнем сложности, оценкой и годами

Оценка	Учебный год		
	2023	2024	2025
Отлично	x_3	x_4	x_5
Хорошо	y_3	y_4	y_5
Удовлетворительно	z_3	z_4	z_5

На основе вышеизложенного методику формирования проверочного материала в соответствии с выбранной студентом оценкой можно представить следующим образом:

1. Преподаватель заполняет информацию на сервере:
 - банк заданий с указанием элементов, входящих в кортеж 1;
 - методические пособия;
 - рекомендации с указаниями по выполнению заданий.
2. Студент регистрируется в системе для получения заданий, заполняя элементы, входящие в кортеж 2, и выбирает оценку, которую хочет получить, от «удовлетворительно» до «отлично».
3. В соответствии с выбранной оценкой студенту предлагаются задания с соответствующим уровнем сложности.

ПОСТРОЕНИЕ ПРОЕКТНЫХ ДИАГРАММ И ОПИСАНИЕ ПО

Исходя их предложенной методики, взаимодействия с программой осуществляют три актора: преподаватель, студент и администратор. Диаграмма вариантов использования (USE CASE в нотации UML) представлена на рисунке 1.

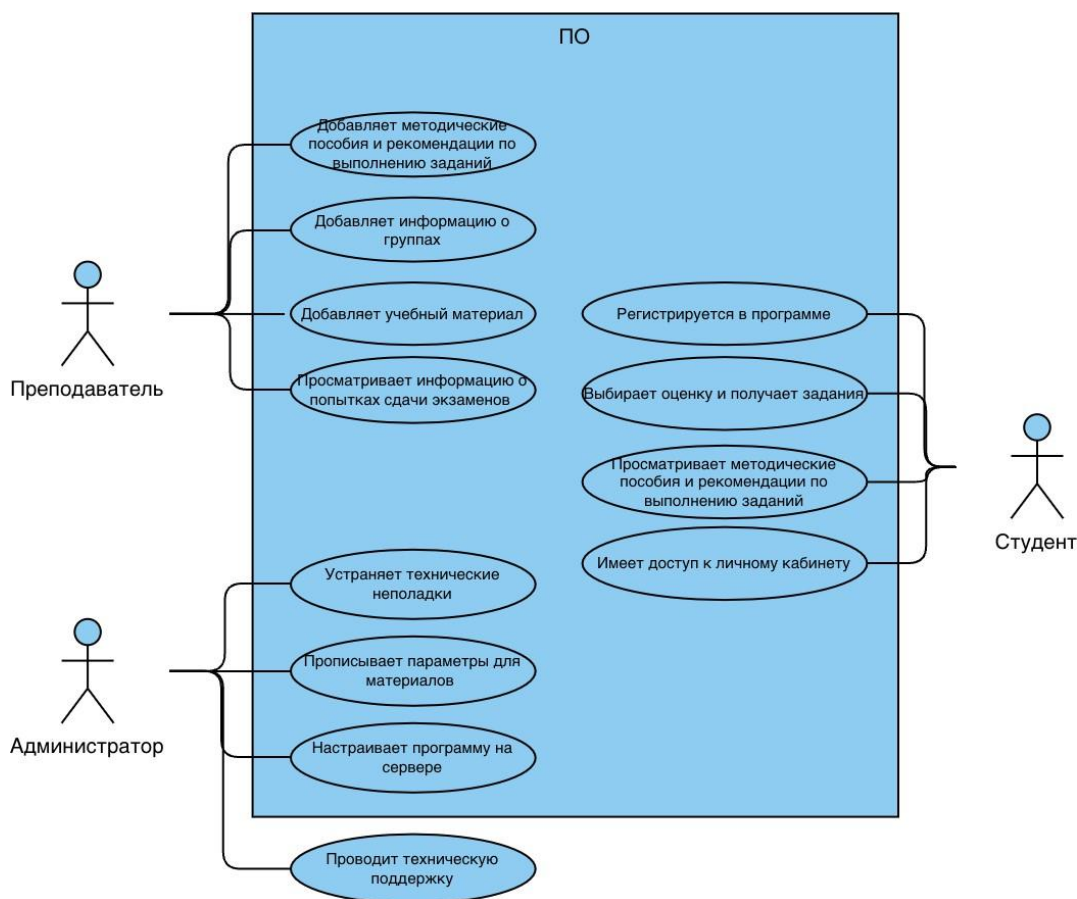


Рисунок 1 – Диаграмма вариантов использования

Основной функционал акторов представлен в таблице 2.

Таблица 2 – Функционал акторов

Актор	Функционал
Преподаватель (актор 1)	Просмотр информации о студентах и выданных им программой заданий Возможность добавления названий групп для унификации формы названий Возможность добавления и редактирования материала Возможность добавления дисциплин и тем
Студент (актор 2)	Возможность регистрации в программе Возможность выбора получаемой оценки Просмотр методических пособий и рекомендаций по выполнению заданий Использование личного кабинета Возможность изменения выбранной оценки
Администратор (актор 3)	Установка программы на серверы Устранение возможных неполадок Предоставление технической поддержки

Структура базы данных, учитывающая набор параметров, входящих в кортеж 1, 2, а также дополнительных данных, представлена на рисунке 2.

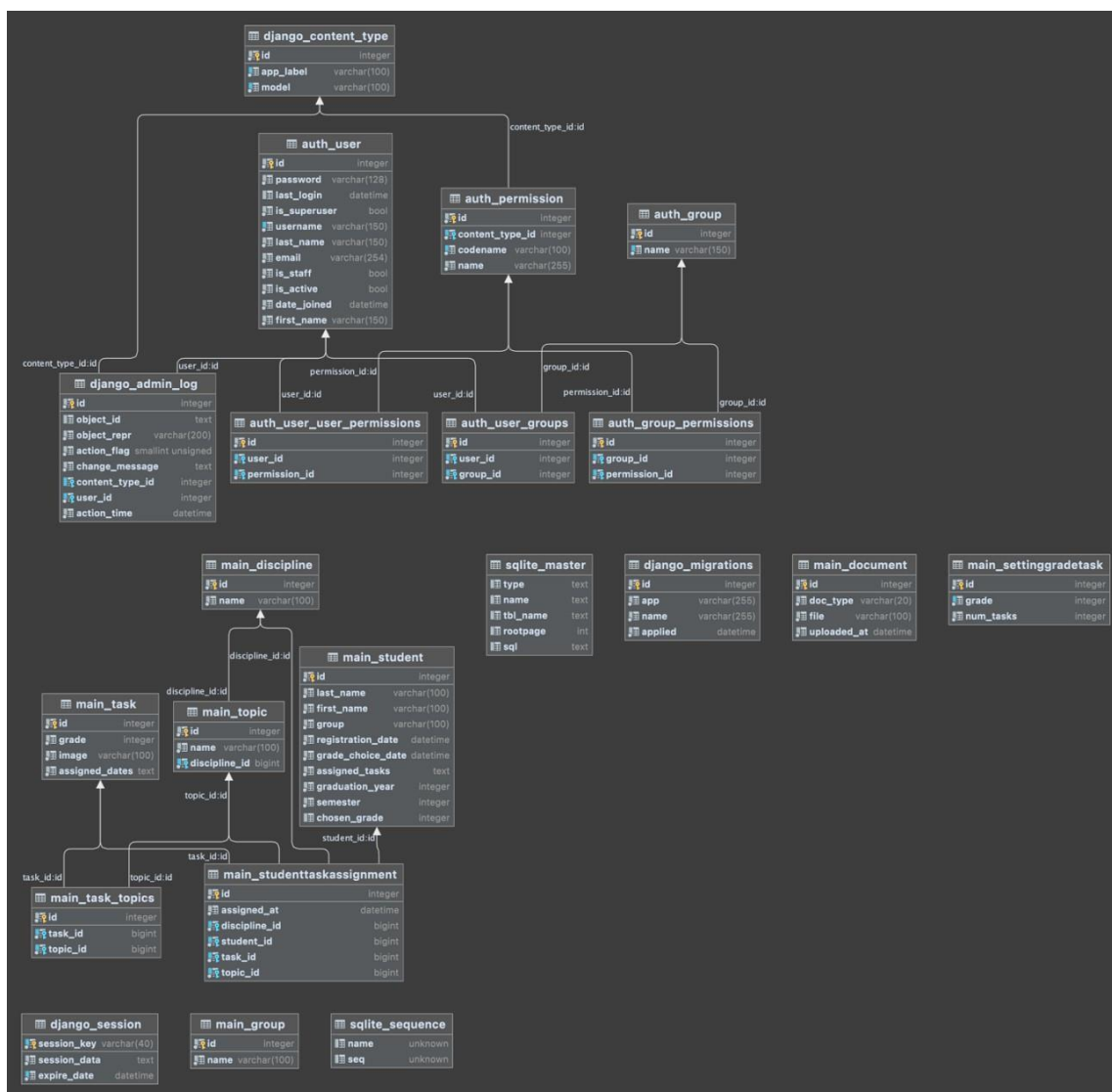


Рисунок 2 – Структура базы данных

Программа представляет собой web-приложение. Front-end написан на языке программирования Python версии 3.10 с использованием фреймворка Django. Back-end разработан в интегрированной среде разработки PyCharm на языке Python.

Web-сервис состоит из модулей, которые обрабатывают данные в программе. Так, в интерфейсе у студента находится модуль регистрации (рис. 3), модуль авторизации (рис. 4), модуль получения задания (рис. 5), модуль отображения материала или, другими словами, личный кабинет.

Интерфейс актора 2 позволяет ему совершать следующие действия:

1. Ввести информацию, входящую в кортеж 2, при регистрации.
 2. Выбрать оценку, на которую претендует студент.
 3. Просмотреть и скачать задания, соответствующие выбранной оценке.
 4. Ознакомиться с рекомендациями по выполнению задания, а также с методическими пособиями.
- Личный кабинет студента представлен на рисунке 6.

[Уже есть аккаунт?](#) [Войти в личный кабинет](#)

Регистрация студента

Фамилия

Имя

Год сдачи

Семестр

Группа

Рисунок 3 – Окно регистрации студента

Вход

Фамилия

Имя

Группа

Рисунок 4 – Окно входа в личный кабинет

Выбор задания

Дисциплина

Тема

Оценка

Рисунок 5 – Модуль получения задания

В интерфейсе преподавателя находятся следующие модули:

- модуль – начальная страница. В модуле отображаются последние действия в программе и ссылки на другие модули;
- модуль для добавления методических пособий и рекомендаций по выполнению заданий;
- модуль добавления, удаления и изменения групп. Модуль создан для унификации названий групп в программе и для выбора студентам при регистрации;
- модуль о студентах. В модуле хранится вся информация о студентах, а также о выданных им заданиях и попытках сдачи экзамена;
- модуль добавления заданий. В данном модуле преподаватель выбирает оценку, соответствующую заданию, и загружает само задание, например, в виде файла pdf;
- модуль ранжирования заданий. В нем преподаватель выбирает количество заданий, соответствующих уровню сложности, оценке.
- модуль добавления дисциплин. В нем преподаватель добавляет дисциплины;
- модуль добавления тем. В нем преподаватель добавляет темы;
- модуль просмотра выданных заданий.

Личный кабинет

Фамилия: Ишкин

Имя: Тимур

Группа: ИТ11

Оценка: 4

Скачать Рекомендации

Скачать Методическое пособие

Получить задание

Ваши задания:



Рисунок 6 – Интерфейс студента (личный кабинет)

Действия преподавателя приведены в таблице 3.

Таблица 3 – Функционал актора 1

Наименование кнопки	Действие преподавателя
Documents	Добавление дополнительных материалов, а именно преподаватель выбирает тип документа (рекомендации, методическое пособие) и загружает файл
Groups	Добавление групп (названия)
Settings grade tasks	Выбор оценки и количество оценок, которые будут выданы студенту в соответствии с выбранной оценкой
Students	Просмотр преподавателем информации о студенте, а именно: <ol style="list-style-type: none"> 1. Фамилия и имя студента 2. Группа студента 3. Выбранная студентом оценка 4. Год и семестр обучения 5. Выданные студенту задания, соответствующие уровню сложности, оценки
Tasks	Выбор оценки и загрузка заданий
Disciplines	Добавление дисциплин (названия)
Topics	Добавление тем (названия и к какой дисциплине принадлежит)
Student tasks assignments	Просмотр заданий, выданных учащимся
Открыть сайт	Открытие web-окна с интерфейсом для студента, для копирования ссылки для выдачи студентам
Изменить пароль	Смена пароля

Интерфейс актора 1 приведен на рисунке 7.

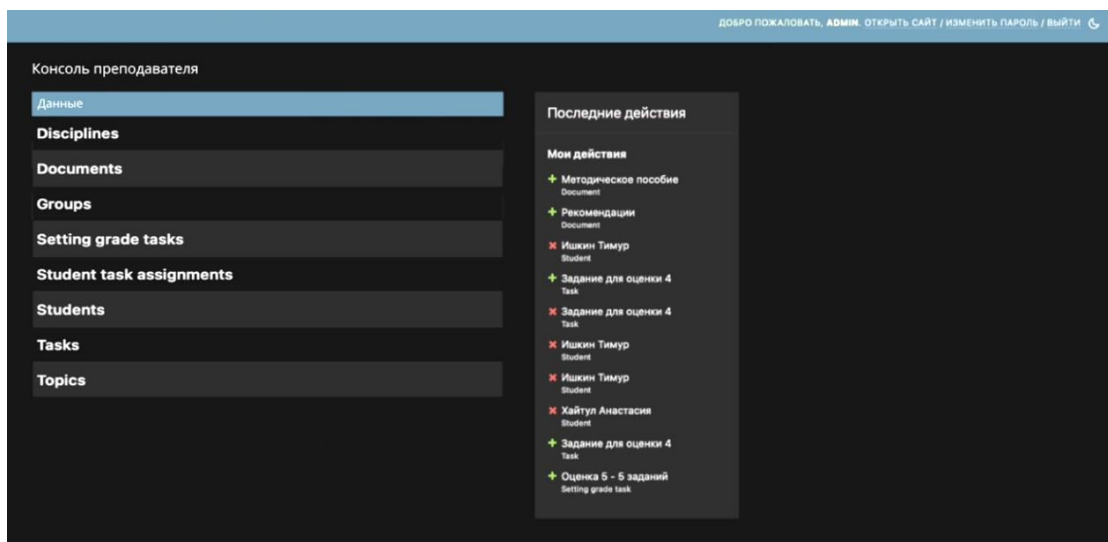


Рисунок 7 – Интерфейс преподавателя

В разработанной программе предусматривается проверка выданных заданий на их повтор, то есть если студенту выдано задание, то оно не может попасть ему или другому обучающемуся в течение определенного количества времени.

ЗАКЛЮЧЕНИЕ

Данная методика и программный продукт были протестированы и апробированы при проведении экзамена «Начертательная геометрия» в ФГБОУ ВО «Астраханский государственный университет им. В. Н. Татищева» в 2023–2024 учебном году и показали свою работоспособность.

Повышение объективности оценки знаний студентов является актуальной задачей, способствующей эффективности обучения в вузе. Разработка методики и соответствующего алгоритмического программного обеспечения позволила устранить ряд имеющихся у «классического» метода проведения экзамена недостатков, и успешная апробация предложенного подхода позволяет рекомендовать его для использования проверки знаний студентов по другим предметам технического профиля, изучаемым в рамках учебного процесса.

Список источников

1. Система оценки достижения планируемых результатов освоения основной образовательной программы начального общего образования // levashov.edu. – URL: https://levashov.edu.yar.ru/nachalnaya_shkola/sistema_otsenki_dostizheniya_planirue_51/sistema_otsenki_dostizheniya_planirue_51.html?with_template=blind (дата обращения: 04.02.2024).
2. Критическая рефлексия // Источник знаний. – URL: <http://sknowledge.ru/Msg.aspx?id=2505> (дата обращения: 04.02.2024).
3. Болонская система, рейтинги и традиции: как ставят оценки в университете // Без сменки. – URL: <https://bez-smenki.ru/bolonskaya-sistema-rejtingi-i-tradicii-kak-stavyat-ocenki-v-universitete/> (дата обращения: 04.02.2024).
4. Зачет, дифференцированный зачет и экзамен: ликбез для первокурсников // PCB. – URL: <https://tsv.ru/blog/zachet-differencirovannyj-zachet-i-ekzamen-likbez-dlya-pervokursnikov/> (дата обращения: 04.02.2024).
5. Кудрявцев, Л. Д. Об экзаменах / Л. Д. Кудрявцев // Вестник Московского университета. Серия 20. Педагогическое образование. – 2014. – № 1. – С. 86–99.
6. Пискотин, В. А. О методике проведения экзаменов и зачётов по тактико-специальной подготовке в вузах МВД России / В. А. Пискотин, В. В. Прищепа // Modern Science. – 2020. – № 12–2.
7. Крюков, Д. Н. Информатизация вуза: практический опыт / Д. Н. Крюков // Высшее образование в России. – 2010. – № 8/9. – С. 45–50.
8. Чоршанбиев, З. Э. Дифференцированное обучение студентов на занятиях высшей математики в техническом вузе / З. Э. Чоршанбиев // Academy. – 2021. – № 4. – С. 42–47.
9. Соловьева, О. А. Содержание и возможности дифференцированного обучения учащихся с особенностями психофизического развития в «инклюзивных группах» учреждений профессионального образования / О. А. Соловьева // Мастерство online. – 2016. – № 3.

References

1. The system of assessment of achievement of planned results of mastering the basic educational programme of primary general education. *levashov.edu*. URL: https://levashov.edu.yar.ru/nachalnaya_shkola/sistema_otsenki_dostizheniya_planirue_51/sistema_otsenki_dostizheniya_planirue_51.html?with_template=blind (accessed 04.02.2024) (In Russ.).

2. Critical reflexion. *Knowledge source*. URL: <http://sknowledge.ru/Msg.aspx?id=2505> (accessed 04.02.2024) (In Russ.).
3. Bologna system, ratings and traditions: how grades are set at the university. *No change*. URL: <https://bezsmenki.ru/bolonskaya-sistema-rejtingi-i-tradicii-kak-stavyat-ocenki-v-universitete/> (accessed 04.02.2024) (In Russ.).
4. Credit, differentiated credit and examination: a lesson for first-year students. *RSV*. URL: <https://rsv.ru/blog/zachet-differenczirovannyj-zachet-i-ekzamen-likbez-dlya-pervokursnikov/> (accessed 04.02.2024) (In Russ.).
5. Kudryavtsev, L. D. About examinations. *Bulletin of Moscow University. Series 20. Pedagogical Education*, 2014, no. 1, pp. 86–99 (In Russ.).
6. Piskotin, V. A., Prishchepa, V. V. On the methodology of examinations and credits for tactical and special training in the universities of the Ministry of Internal Affairs of Russia. *Modern Science*, 2020, no. 12–2 (In Russ.).
7. Kryukov, D. N. Informatisation of higher education institution: practical experience. *Higher Education in Russia*, 2010, no. 8/9, pp. 45–50 (In Russ.).
8. Chorshanbiev, Z. E. Differentiated training of students in the classes of higher mathematics in a technical university. *Academy*, 2021, no. 4, pp. 42–47 (In Russ.).
9. Solovieva, O. A. Content and possibilities of differentiated training of students with peculiarities of psychophysical development in "inclusive groups" of vocational education institutions. *Mastery online*, 2016, no. 3 (In Russ.).

Статья поступила в редакцию 15.03.2024; одобрена после рецензирования 21.03.2024; принята к публикации 22.03.2024.

The article was submitted 15.03.2024; approved after reviewing 21.03.2024; accepted for publication 22.03.2024.

УДК 004.021, 004.043, 004.4

АЛГОРИТМ ГЕНЕРАЦИИ ТЕСТОВ ДЛЯ ОЛИМПИАДНЫХ ЗАДАЧ ПО ПРОГРАММИРОВАНИЮ

Смирнова Юлия Александровна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
старший преподаватель кафедры информационных технологий, ORCID: 0000-0002-3807-5062,
e-mail: got-22@mail.ru

Скрипник Владислав Викторович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
студент, ORCID: 0009-0009-7181-0215, e-mail: Ill-ooo-2003@mail.ru

Головко Юлия Александровна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат технических наук, доцент кафедры информационных технологий, ORCID: 0000-0001-5401-270X, e-mail: pjulia2014@yandex.ru

Статья посвящена методике генерации тестовых заданий для оценки решений олимпиадных задач по программированию с использованием инновационных методов машинного обучения, генетических алгоритмов и классических подходов к генерации тестов. Также в статье подчеркивается важность качественной организации олимпиад по программированию, обсуждаются различные виды задач, которые могут встретиться у участников, и рассматриваются стратегии генерации тестов, включая ручной поиск, генерацию по шаблону и создание тестов против неэффективных решений. Основная цель алгоритма, описанного в исследовании, заключается в создании эффективного набора тестов, способного проверить правильность решений и выявить неэффективные подходы. Авторы предлагают комбинированный подход, состоящий из трех слоев: модели машинного обучения, генетического алгоритма и классической генерации, каждый из которых выполняет определенные функции в процессе создания тестов. Авторы подчеркивают важность корректного кодирования тестов и выбора функции приспособленности для генетического алгоритма. Было проведено сравнение тестов, созданных алгоритмом, с валидным набором тестов, чтобы оценить эффективность алгоритма. Результаты показывают, что процент пропуска неверных решений колеблется от 3 до 23 %, что свидетельствует о перспективности и эффективности предложенного подхода.

Ключевые слова: методика, генерация, тестовые задания, олимпиадные задачи, программирование, машинное обучение, генетические алгоритмы

Финансирование: исследование выполнено при финансовой поддержке Фонда содействия инновациям «УМНИК».

ALGORITHM OF TEST GENERATION FOR OLYMPIAD PROGRAMMING TASKS

Smirnova Yulia A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Senior Lecturer, Department of Information Technologies, ORCID: 0000-0002-3807-5062, e-mail: got-22@mail.ru

Skripnik Vladislav V., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
student, ORCID: Ill-ooo-2003@mail.ru.

Golovko Yulia A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,

Cand. Sci. (Engineering), Associate Professor, ORCID: 0000-0001-5401-270X, e-mail: pjulia2014@yandex.ru

The paper is devoted to the methodology of test generation for evaluating solutions of olympiad programming problems using innovative machine learning methods, genetic algorithms and classical approaches to test generation. It also emphasizes the importance of quality organization of programming Olympiads, discusses different types of problems that participants may encounter, and reviews test generation strategies, including manual search, template generation, and test generation against inefficient solutions. The main goal of the algorithm described in the study is to create an efficient test suite capable of verifying the correctness of decisions and identifying ineffective approaches. The authors propose a combined approach consisting of three layers: a machine learning model, a genetic algorithm and classical generation, each of which performs certain functions in the test generation process. The authors emphasize the importance of correct test coding and the choice of the fitness function for the genetic algorithm. A comparison was made between the tests generated by the algorithm and a validated test set to evaluate the effectiveness of the algorithm. The results show that the percentage of missing incorrect solutions ranges from 3 to 23 %, indicating that the proposed approach is promising and effective.

Keywords: keywords: methodology, generation, test cases, olympiad tasks, programming, machine learning, genetic algorithms

Financial support: the research was supported by the financial support of the Foundation for Assistance to Innovations “UMNIK”.

ВВЕДЕНИЕ

В современном мире информационных технологий и компьютерных наук, олимпиады по программированию занимают особое место, так как позволяют выявить способных и одаренных обучающихся в указанной предметной области, помогают развитию творческого и умственного потенциала, а также являются средством профессиональной ориентации.

Основная цель олимпиадного программирования (ОпП) – это демонстрация знаний и умений использования основных типов алгоритмов (с различными структурами данных, на графах, динамического программирования, целочисленной арифметики, теории игр и т. д.). Область разработки алгоритмов позволяет участникам продемонстрировать свои лучшие качества и навыки в решении олимпиадных задач и достичь понимания, что разработка качественного алгоритма имеет важное решение для производительности любой программной системы.

Участники соревнований используют различные языки программирования, алгоритмы, структуры данных и различные инструменты разработки для решения различной сложности задач. В ОпП обычно предлагаются задачи на различные алгоритмические и структурные задачи. Например:

1. Сортировка и поиск: например, написание алгоритма сортировки массива или поиска определенного элемента [1].
2. Динамическое программирование: создание оптимальных алгоритмов для решения задач, требующих оптимальной подструктуры и повторного использования подзадач [1].
3. Графы и деревья: задачи, связанные с поиском кратчайшего пути, нахождением связанных компонент, обходом графов и другими алгоритмами, связанными с графовыми структурами данных.
4. Дискретная математика: задачи, требующие знаний о комбинаторике, теории чисел, теории графов и других ветвях дискретной математики.
5. Геометрия: задачи, требующие решения задач о расположении точек, отрезков, плоскостей и других геометрических фигур.

Качественная организация олимпиады по программированию связана не только с разработкой и формированием заданий, но и подготовкой проверочных тестов к ним.

Генерация тестов для олимпиадных задач требует углубленных знаний в области математики, программирования и логического мышления, потому что основная и важная цель этого процесса заключается в выявлении неэффективных решений.

В данной статье излагается разработанная методика генерации тестовых заданий для оценки решений олимпиадных задач, которая опирается на инновационные методы машинного обучения, генетических алгоритмов и классических алгоритмов генерации [2, 3, 4, 5].

МЕТОДИКА ГЕНЕРАЦИИ ПРОВЕРОЧНЫХ ТЕСТОВ

Для эффективного выявления верных решений необходимо создать такой набор тестов, который должен покрывать все условия задачи, учитывая существующие стратегии поиска и создания. Для поиска таких тестов существует несколько стратегий поиска и создания:

1. Ручной поиск – тесты, которые пишутся вручную и проверяют некоторые специальные случаи и условия задачи, например, пограничные значения или специфические условия.
2. Генерация по шаблону – тесты большого размера, как правило, генерируются по определенному шаблону, например, если задача связана с временными рядами и поиском, то генерируется достаточно длинная серия цифр; если это задача, связанная с графами, то генерируется достаточно большая квадратная матрица; в общем случае такие тесты генерируются с помощью программы.
3. Поиск тестов против неэффективного решения – такие решения могут пройти некоторый набор тестов, но есть такие тесты, на которых оно выдает неверный ответ. При генерации теста против неэффективного решения предполагается, что такое решение уже имеется. Тесты, сгенерированные против некоторого неэффективного решения, обычно оказываются «фатальными» для достаточно большого множества неэффективных решений. При разумном выборе решений, против которых генерируются тесты, возможно «покрыть» почти все неэффективные решения [6].

Создание набора тестов представляет собой достаточно трудоемкий процесс из-за наличия различных стратегий поиска тестов, набора входных данных и условий задач, а также поиска единого алгоритма для их генерации.

Разные стратегии поиска тестов, набор входных данных и задач порождают проблему поиска единого алгоритма генерации тестов. Для каждой задачи создание набора тестов является очень долгим и трудоемким процессом.

СТЕПЕНЬ РАЗРАБОТАННОСТИ

В настоящий момент существует ряд научных работ, связанных с рассмотрением задач генерации тестов для проверки олимпиадных заданий [7, 8]. Можно заметить, что методики генерации, описанные в статьях, указанных выше, достаточно эффективные, однако их серьезным недостатком является написание алгоритма для каждой конкретной задачи и ее решения, что исключало его применение для остальных заданий.

В настоящее время многие разработчики генерируют тесты для написания программ с применением LLM (Large Language Model – большой языковой модели), одновременно сталкиваясь с проблемой плохого понимания данной технологией специфики содержательной части олимпиадных задач.

Анализ существующих аналогов программ для генерации тестов проверки олимпиадных заданий, а именно выявление их достоинств и недостатков, позволил сформулировать требования к разработке методики, рассмотренной в статье, для достижения эффективного процесса проверки решений и минимизации ошибок технического характера.

На основании предложенной методики был разработан первый прототип алгоритма, который будет эффективно генерировать набор тестов для любых видов олимпиадных задач по программированию.

Алгоритм представляет собой 3 слоя, 2 промежуточных и место хранения всех тестов (рис. 1). Каждый слой – это алгоритм, который генерирует часть общего набора, настроенный на определенные стратегии поиска.

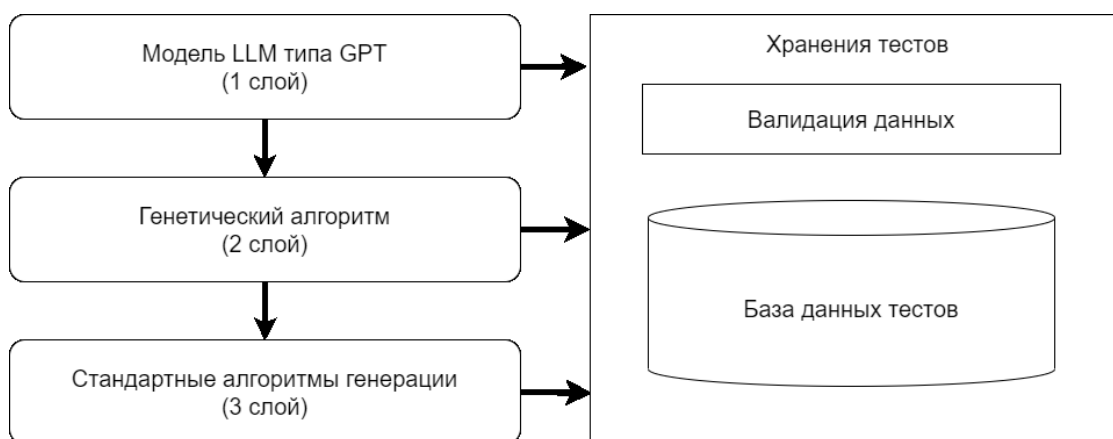


Рисунок 1 – Схема алгоритма

СЛОЙ 1. МОДЕЛЬ МАШИННОГО ОБУЧЕНИЯ

Модель машинного обучения типа GPT [9] используется для понимания естественного языка и создания специального шаблона prompt, в то же время для ускорения работы над алгоритмом применяется готовое решение от Open AI chat-gpt 3.5. На выходе получается набор из пяти тестов в формате .json, структура которого представлена на рисунке 2.

Для понимания естественного языка на первом слое алгоритма находится модель машинного обучения типа GPT. Для ускорения работы над алгоритмом используется готовое решение от Open AI chat-gpt 3.5. Модель машинного обучения сгенерировала для нас тесты, идет создание специального шаблона prompt, в который интегрируется задача. На выходе получается набор из 5 тестов в формате .json. Структура файла представлена на рисунке 2.

```
{
  "n": 1,
  "m": 5,
  "A": [1],
  "B": [1, 2, 3, 4, 5]
```

Рисунок 2 – Структура сгенерированного теста

Первый слой алгоритма генерирует простой набор тестов, его можно назвать входным. Это небольшие тесты, которые содержат разные случаи входных данных. Например, разных цифр, одинаково идущей подряд последовательности из чисел или символов и некоторые краевые случаи в задаче.

После генерации тестов они отправляются в слой хранения. Все тесты в дальнейшем будут в формате .json, для более удобной работы с ними.

СЛОЙ 2. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Специфика данного алгоритма заключается в том, что необходимо правильно кодировать тесты в виде генов [10]. Способ кодирования теста зависит от задачи, для которой генерируются тесты. Он должен эффективно учитывать ограничения, заданные в условии задачи. Поэтому для эффективного кодирования была предложена идея обобщить виды тестов и представить их в формате .json схемы (рис. 3).

```
{
  "schema": {
    "n": {
      "type": "int",
      "settings": {
        "min_val": 1,
        "max_val": 10
      },
      "mutation": false
    },
    "m": {
      "type": "int",
      "settings": {
        "min_val": 1,
        "max_val": 10
      },
      "mutation": false
    },
    "A": {
      "type": "array",
      "settings": {
        "item": "int",
        "len": "n",
        "min_val": 1,
        "max_val": 100
      },
      "mutation": true
    },
    "B": {
      "type": "array",
      "settings": {
        "item": "int",
        "len": "m",
        "min_val": 1,
        "max_val": 100
      },
      "mutation": true
    },
    "properties": ["n", "m", "A", "B"]
  },
  "representation": "template.txt"
}
```

Рисунок 3 – Структура схемы в формате .json

Данный подход поможет алгоритму корректно кодировать тесты для любых видов задач. Также данный подход позволяет правильно манипулировать генами и не допускать невалидных видов тестов.

Далее необходимо выбрать функцию приспособленности (P). Для правильного подбора функции нам необходимо понять, какая стратегия поиска будет у данного слоя. Нельзя выбрать стратегию поиска, которая завесила от каких-либо числовых показателей, таких как время работы (t), потребление памяти (I) или размер стека вызова (I_s). Данные характеристики могут отличаться в зависимости от языка программирования и машины, на которой решение запускается:

$$P \neq \{t, I, I_s\}, \tag{1}$$

где t – время работы программы в с.;

I – потребление памяти в байтах;

I_s – размер стека вызова процессора в у.е.

Если выбрать в качестве функции приспособленности число выполненных инструкций кода (C_i), то это способно решить проблемы, изложенные выше. Но тогда генетический алгоритм будет выбирать тесты с большим объемом входных данных, обходя стороной тесты, которые проверяют уникальные случаи.

Поэтому было принято решение в качестве функции приспособленности использовать следующую стратегию поиска: на вход функции будут подаваться несколько неэффективных решений задачи (C_a), то есть существуют такие тесты, на которых данные решения будут выдавать неправильный ответ или не проходить по ограничениям сгенерированный тест (T):

$$P(C_a, T) = \sum_{i=1}^n check(C_a^i, T), \tag{2}$$

где n – длина множества C_a ;

$$check(A, T) = \begin{cases} 10, & answer(A, T) \in \{WA\} \\ 5, & answer(A, T) \in \{TL, ML\}, \\ -10, & answer(A, T) \in \{OK\} \end{cases} \quad (3)$$

где $answer$ – функция проверки задания A на тесте T ;

WA – вердикт означает, что решение выдало неверный ответ;

TL – вердикт означает, что решение не уложилось в ограничение по времени;

ML – вердикт означает, что решение не уложилось в ограничение по памяти.

В ходе работы над алгоритмом было принято решение оставить только функцию мутации генов. Данное решение было обусловлено тем, что разработать правильный инструмент для скрещивания генов при условии многообразия задач было бы очень долгим процессом, который не давал бы значительного прироста в поиске необходимых тестов.

Мутация генов происходит после каждой итерации алгоритма. Для решения проблемы валидности тестов после мутации помогают некоторые настройки в схеме (рис. 4).

```
"settings": {
  "item": "int",
  "len": "n",
  "min_val": 1,
  "max_val": 100
},
```

Рисунок 4 – Пример настроек в схеме данных

Алгоритм на этапе мутации сверяется с данной схемой и промутирует только те значения, которые разрешены для мутации.

Общая схема работы второго слоя алгоритма представлена на рисунке 5.

Данный алгоритм работает циклически. Всего заедается несколько сотен итераций и от 3 до 5 эпох работы

После работы второго слоя, как и в конце первого слоя, все выбранные тесты попадают в общий слой хранения. Тесты также сохраняются в формате `.json`.

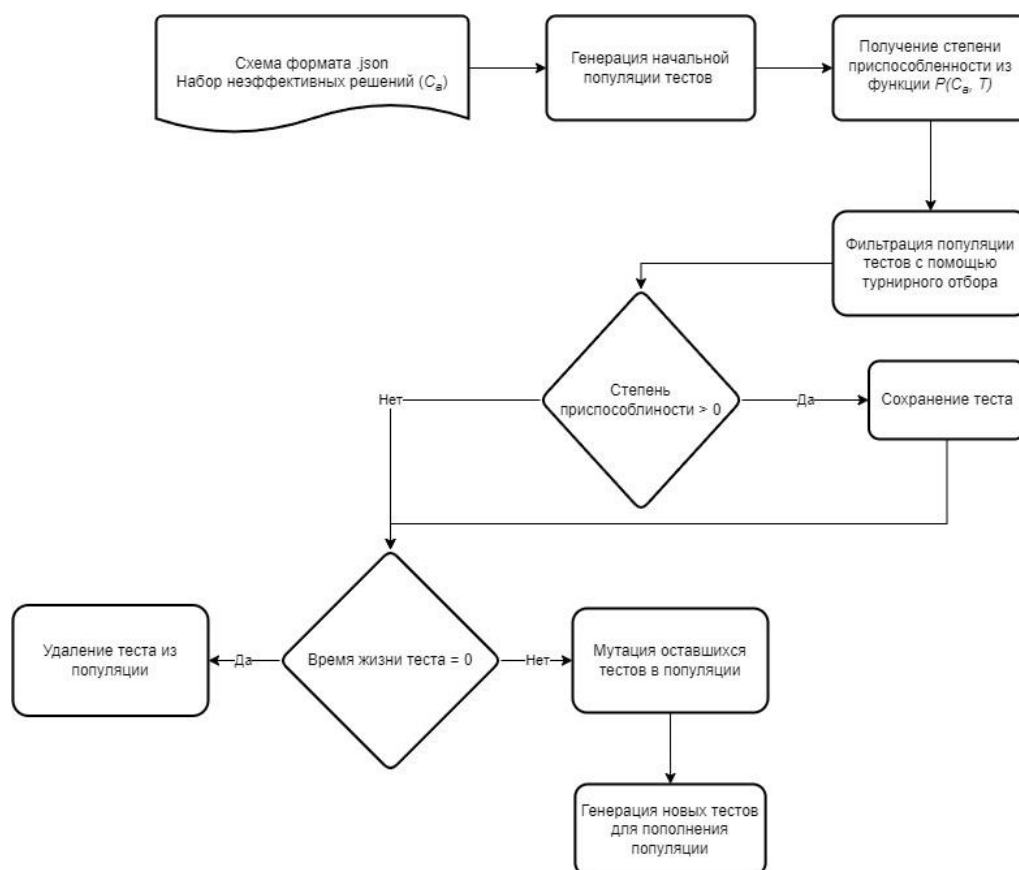


Рисунок 5 – Блок-схема слоя генетического алгоритма

СЛОЙ 3. КЛАССИЧЕСКИЙ

Последний слой алгоритма является классической случайной генерацией. За основу слоя был взят алгоритм создания начальной популяции в предыдущем слое. Для генерации тестов необходима схема в формате .json. По данной схеме генерируются тесты [11].

Данный слой необходим для создания «экстремальных» тестов. То есть данные тесты будут проверять краевые случаи: время работы и потребление памяти. Размер таких тестов обычно достигает нескольких мегабайтов.

Для проверки краевых случаев схема по ходу генерации модифицируется, чтобы генератор правильно мог генерировать большие тесты.

Из-за особенности данного слоя тесты, которые он генерирует, сразу конвертируются в формат, который будет использоваться для проверки. Данное решение обусловлено экономией оперативной памяти. В итоге в общий слой попадают тесты не формата .json, а сразу тесты формата .txt.

ОБЩИЙ СЛОЙ ХРАНЕНИЯ ТЕСТОВ

Для сохранения всех сгенерированных тестов по ходу работы алгоритма существует слой хранения тестов. В данном слое производится хранение тестов и существует возможность получения тестов для работы некоторых слоев. Все тесты кроме тестов с 3 слоя хранятся в формате .json.

После работы алгоритма в данном слое идет конвертация тестов из формата .json в формат, который будет использоваться в программе проверки.

ПРИМЕНЕНИЕ ПОДХОДА

Для тестирования данного подхода было выбрано 3 задачи: массивы-палиндромы, оптические каналы связи, наибольшая возрастающая подпоследовательность. Все задачи были взяты с различных этапов ВСОШ по информатике за 2021–2022 [12, 13].

Массивы-палиндромы

Условие данной задачи такое: даётся два массива A и B длин n и m соответственно. От каждого массива отбрасывается произвольный префикс и суффикс, чтобы получить массивы A' и B' длины k . Суммируются поэлементно массивы A' и B' , получается массив C длины k . Нужно найти максимальное k , при котором C является палиндромом.

Задача являлась заключительной для первого дня, что показывает ее сложность. Данная задача была выбрана из-за того, что просто так с помощью алгоритмов стандартной генерации сделать правильный набор было бы невозможно.

Оптические каналы связи

Условие данной задачи такое: даётся n центров подключения, пронумерованных от 1 до n , где 1 – главный центр. Центры подключены проводными каналами связи, образующими дерево.

Для города i , где $i > 1$, обозначим первый город на маршруте от города i до столицы как p_i . Стоимость замены канала на оптический, который соединяет город i с городом p_i , равна w_i .

Каждый центр может иметь не более k оптических каналов. Нужно выбрать для модернизации максимальное количество каналов с минимальной суммарной стоимостью.

Данная задача являлась предпоследней для второго дня и была основана на теории графов, поэтому и была выбрана для проведения эксперимента.

Наибольшая возрастающая подпоследовательность

Условие данной задачи такое: дана последовательность, требуется найти длину наибольшей возрастающей подпоследовательности.

Эта задача была выбрана для того, чтобы посмотреть, как алгоритм справится с достаточно простыми задачами, где тестовую выборку можно сгенерировать с помощью просто случайной генерации чисел в массиве.

Задачи требуют разного вида входных тестов и подходов решения. Для каждой из этих задач была написана схема описания входных параметров и собраны различные решения. Всего было собрано 26 различных решений и по каждой задаче было собрано не менее 3 неверных для работы слоя с генетическим алгоритмом.

НАЧАЛЬНЫЕ УСЛОВИЯ ЭКСПЕРИМЕНТА

Целью эксперимента было проверить работоспособность предлагаемого алгоритма. Для этого для каждой задачи была запущена генерация тестового набора. Для удобной работы с алгоритмом был написан микросервис с открытым API.

Всего было выбрано 3 задачи различной сложности. У каждой задачи были взяты ответы, которые были как правильные, так и неправильные. Выбранные задания были взяты с ВСОИШ предыдущих годов и сайтов, где представлены тренировочные задания на определённые темы информатики. Набор тестов для данных заданий был признан достаточно валидным и показал свое высокое качество.

Таблица 1 – Начальные параметры для каждой задачи

Название задания			
Количество различных ответов	Количество неправильных ответов	Количество тестов	Количество зачтённых решений
Массивы-палиндромы			
13	5	73	8
Оптические каналы связи			
17	7	68	10
Наибольшая возрастающая под последовательность			
28	3	38	25

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

Всего в сумме на все 3 задачи было потрачено 22056 секунд или около 6 часов. Дольше всего была генерация для 2 задачи (20123 с), самой быстрой генерация была для 3 задачи (1079 с).

По каждой задаче алгоритму задавалось то же самое количество тестов, что уже было на момент начала эксперимента. Также были заданы отдельные настройки каждого слоя. Для слоя с нейронной сетью было задано сгенерировать 5 тестов. Для слоя с генетическим алгоритмом были заданы следующие параметры: 2000 итераций, 2 эпохи, 30 генов в наборе и время жизни в 3 итерации. Для слоя классической генерации было написано условие, что он генерирует столько тестов, сколько осталось до заполнения тестового набора.

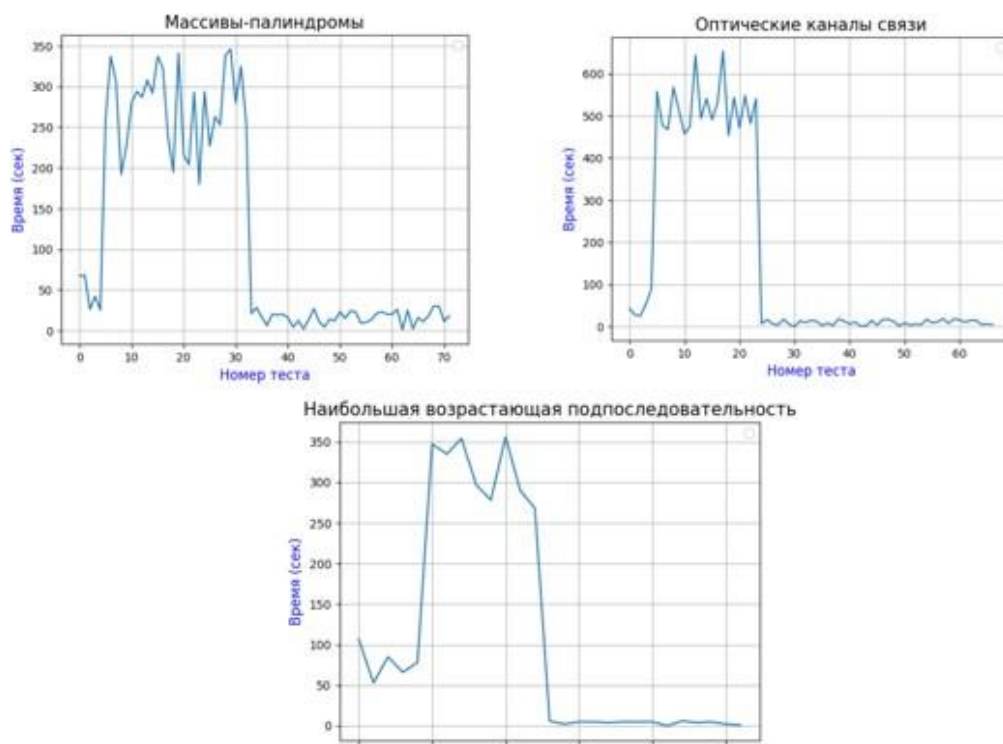


Рисунок 6 – График времени работы алгоритма

В итоге работы алгоритма мы получили новый сгенерированный набор тестов. После этого было проведено сравнение валидного набора тестов с сгенерированным, для того чтобы можно было оценить разницу в количестве пропущенных неправильных ответов на задачу.

Таблица 2 – Итоги эксперимента

Название задания				
Количество различных ответов	Количество неправильных ответов	Количество зачтённых решений	Количество ошибочно зачтенных решений	Процент ошибки
Массивы-палиндромы				
13	5	10	2	15 %
Оптические каналы связи				
17	7	14	4	23 %
Наибольшая возрастающая под последовательность				
28	3	26	1	3 %

В итоге можно увидеть, что процент пропуска неверных решений является нестабильным и варьируется от 23 до 3 %. После более тщательного анализа было решено, что процент пропуска от 25 % и менее является удовлетворительным и показывает состоятельность данного подхода для решения подобной задачи.

ЗАКЛЮЧЕНИЕ

В работе описан алгоритм, позволяющий автоматически генерировать тесты для олимпиадных задач по программированию с использованием комбинированного подхода. Данный алгоритм был протестирован на реальных заданиях и показал неплохой результат. Полученные результаты позволяют утверждать, что описанный подход является достаточно перспективным в плане его применения при подготовке тестов для задач по олимпиадному программированию в целях повышения качества олимпиад. Статья представляет важную методику исследования в области генерации тестов для олимпиадных задач по программированию с использованием современных методов машинного обучения и генетических алгоритмов, что может значительно облегчить процесс проверки решений и повысить качество проведения таких соревнований.

Библиографический список

1. Смирнова, Ю. А. Учебное пособие по дисциплине «Основы программирования на Python» для очно-заочной формы обучения / Ю. А. Смирнова. – Астрахань, 2016. – 76 с.
2. ACM International Collegiate Programming Contest. – URL: http://en.wikipedia.org/wiki/ACM_ICPC (дата обращения: 10.02.2023).
3. TopCoder. – URL: <http://www.topcoder.com/tc> (дата обращения: 10.02.2023).
4. International Olympiad in Informatics. – URL: <http://www.ioinformatics.org>, свободный. Яз. англ. (дата обращения: 10.02.2023).
5. Google Code Jam. – URL: <http://code.google.com/codejam>, свободный. Яз. англ. (дата обращения 10.02.2023).
6. Гэри М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – Москва : Мир, 1982. – 416 с.
7. Буздалов, М. В. Генерация тестов для олимпиадных задач по программированию с использованием генетических алгоритмов / М. В. Буздалов // Научно-технический вестник СПбГУ ИТМО. – 2011. – № 2 (72). – С. 72–77.
8. Буздалов, М. В. Генерация тестов для олимпиадных задач по теории графов с использованием эволюционных стратегий / М. В. Буздалов // Научно-технический вестник СПбГУ ИТМО. – 2011. – № 6 (76). – С. 123–127.
9. Generative pre-trained transformer // Википедия. – URL: https://ru.wikipedia.org/wiki/Generative_pre-trained_transformer (дата обращения: 10.02.2023).
10. Back, T. A Survey of Evolution Strategies / T. Back, F. Hoffmeister, H.-P. Schwefel // Proceedings of the Fourth International Conference on Genetic Algorithms. – Morgan Kaufman, 1991. – P. 2–9.
11. Колдаев, В. Д. Основы алгоритмизации и программирования : учебное пособие / В. Д. Колдаев ; под ред. проф. Л. Г. Гагариной. – Москва : ИД «ФОРУМ» : ИНФРА-М, 2009. – 416 с.
12. Интернет-олимпиады по информатике. – URL: <http://neerc.ifmo.ru/school/io/> (дата обращения 10.02.2023).
13. Russian Code Cup. – URL: <http://russiancodecup.ru> (дата обращения: 10.02.2023).

References

1. Smirnova, Yu. A. *Textbook on the discipline "Fundamentals of Python Programming" for part-time and part-time education*. Astrakhan, 2016. 76 p. (In Russ.).
2. *ACM International Collegiate Programming Contest*. URL: http://en.wikipedia.org/wiki/ACM_ICPC (accessed 10.02.2023).
3. *TopCoder*. URL: <http://www.topcoder.com/tc> (accessed 10.02.2023).
4. *International Olympiad in Informatics*. URL: <http://www.ioinformatics.org> (accessed 10.02.2023).
5. *Google Code*. URL: <http://code.google.com/codejam> (accessed 10.02.2023).
6. Gary, M., Johnson, D. *Computers and hard-to-solve problems*. Moscow, Mir Publ., 1982. 416 p. (In Russ.).
7. Buzdalov, M. V. Generation of tests for programming olympiad problems using genetic algorithms. *Scientific and Technical Bulletin of St. Petersburg State University ITMO*, 2011, no. 2 (72), pp. 72–77 (In Russ.).
8. Buzdalov, M. V. Generation of tests for olympiad problems in graph theory using evolutionary strategies. *Scientific and Technical Bulletin of St. Petersburg State University ITMO*, 2011, no. 6 (76), pp. 123–127 (In Russ.).
9. Generative pre-trained transformer. *Wikipediya*. URL: https://ru.wikipedia.org/wiki/Generative_pre-trained_transformer (accessed 10.02.2023) (In Russ.).
10. Back, T., Hoffmeister, F., Schwefel, H.-P. A Survey of Evolution Strategies. *Proceedings of the Fourth International Conference on Genetic Algorithms*. Morgan Kaufman, 1991, pp. 2–9.
11. Koldaev, V. D. *Fundamentals of algorithmization and programming : textbook*. Ed. by prof. L. G. Gagarina. Moscow, Publishing House "FORUM": INFRA-M, 2009. 416 p. (In Russ.).
12. *Internet Olympiads in computer science*. URL: <http://neerc.ifmo.ru/school/io/> (accessed 10.02.2023) (In Russ.).
13. *Russian Code Cup*. URL: <http://russiancodecup.ru> (accessed 10.02.2023) (In Russ.).

Статья поступила в редакцию 08.04.2024; одобрена после рецензирования 26.04.2024; принята к публикации 10.05.2024.

The article was submitted 08.04.2024; approved after reviewing 26.04.2024; accepted for publication 10.05.2024.

УДК 004.8

**УЛУЧШЕННЫЙ ПРОГРАММНЫЙ КОМПОНЕНТ ВИЗУАЛИЗАЦИИ
ГРАФОВ ФРЕЙМВОРКА GOLEM**

Гордеев Иван Иванович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат физико-математических наук, доцент, ORCID: 0000-0001-5036-4791, e-mail: g2i@mail.ru

Черкасова Валентина Андреевна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат физико-математических наук, доцент, старший научный сотрудник лаборатории «Математическое моделирование и информационные технологии в науке и образовании», ORCID: 0000-0002-7510-3042, e-mail: valyc@mail.ru

Бубеницкова Ирина Александровна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат педагогических наук, доцент, ORCID: 0000-0002-5811-5367, e-mail: iris_100@mail.ru
Алферин Станислав Евгеньевич, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
бакалавр, ORCID: 0009-0000-5792-4632, e-mail: genstasik@gmail.com

Целью работы является представление улучшенного программного компонента визуализации графов фреймворка GOLEM. Поставлены задачи: изучить предметную область, рассмотреть алгоритмы визуализации графов на плоскости, выбрать алгоритм для реализации улучшенного программного компонента, реализовать алгоритм, провести тщательное тестирование улучшенного программного компонента и сравнить качество визуализации графов с фреймворком GOLEM. Выдвинута гипотеза о том, что разработанный улучшенный программный компонент визуализации графов обеспечивает более высокое качество визуализации графов по сравнению с существующим фреймворком GOLEM. Предполагается, что улучшенный программный компонент визуализации графов должен обеспечивать не большее (по возможности, меньшее) количество пересечений ребер графа по визуальному контролю, не большее (по возможности, меньшее) количество наложений вершин друг на друга по визуальному контролю, не большее (по возможности, меньшее) количество пересечений ребер графа по геометрическому контролю на основе координат, не большее (по возможности, меньшее) количество наложений вершин друг на друга по геометрическому контролю на основе координат, а также задержка визуализации должна составлять не более 5 секунд по сравнению с существующим компонентом визуализации фреймворка GOLEM. В ходе тестирования было установлено, что разработанный улучшенный программный компонент визуализации графов показывает лучшие результаты.

Ключевые слова: визуализация графов, силовые алгоритмы, генетические алгоритмы, фреймворк GOLEM, networkx, алгоритм Фрухтермана – Рейнгольда

Благодарности: авторы выражают благодарность профессору университета ИТМО Бухановскому Александру Валерьевичу и разработчику GOLEM Киргизову Григорию Валерьевичу за участие в обсуждении реализации программного компонента. Также авторы благодарят студентов Астраханского государственного университета им. В. Н. Татищева Скибо Галину Максимовну, Ясько Тимофею Макаровича, Шаронову Анастасию Андреевну за участие в тестировании улучшенного программного компонента визуализации графов фреймворка GOLEM.

**IMPROVED GRAPH VISUALIZATION SOFTWARE COMPONENT
OF THE GOLEM FRAMEWORK**

Gordeev Ivan I., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Cand. Sci. (Physics & Mathematics), Associate Professor, ORCID: 0000-0001-5036-4791, e-mail: g2i@mail.ru

Cherkasova Valentina A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Cand. Sci. (Physics & Mathematics), Associate Professor, ORCID: 0000-0002-7510-3042, e-mail: valyc@mail.ru

Bubenschikova Irina A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Cand. Sci. (Pedagogy), Associate Professor, 0000-0002-5811-5367, e-mail: iris_100@mail.ru

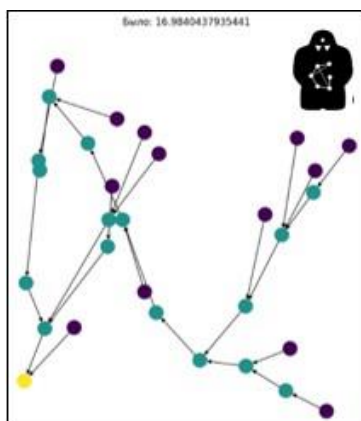
Alferin Stanislav E., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
bachelor, ORCID: 0009-0000-5792-4632, e-mail: genstasik@gmail.com

The aim of the work is to present an improved software component for graph visualization of the GOLEM framework. The tasks are to study the subject area, to consider the algorithms for graph visualization on the plane, to choose an algorithm for implementing the improved software component, to implement the algorithm, to conduct a thorough testing of the improved software component and to compare the quality of graph visualization with the GOLEM framework. The hypothesis is that the developed improved software component for graph visualization provides a higher quality of graph visualization, compared to the existing GOLEM framework. It is assumed that the improved software component for graph visualization should provide no more (if possible, less) number of edge crossings of the graph by visual control, no more (if possible, less) number of vertex overlaps by visual control, no more (if possible, less) number of edge crossings of the graph by geometric control based on coordinates, no more (if possible, less) number of vertex overlaps by geometric control based on coordinates, and the visualization delay should be no more than 5 seconds, compared to the existing visualization component of the GOLEM framework. In the course of testing, it was found that the developed improved software component for graph visualization shows better results.

Keywords: graph visualization, force-directed algorithms, genetic algorithms, GOLEM framework, networkx, Fruchterman – Reingold algorithm

Acknowledgements: the authors express their gratitude to ITMO University professor Alexander Valerievich Bukhanovsky and GOLEM developer Grigory Valerievich Kirgizov for their participation in the discussion of the implementation of the software component. Also, the authors thanks to the students of Astrakhan Tatishchev State University Skibo Galina Maksimovna, Yasko Timofey Makarovich, Sharonova Anastasia Andreevna for participation in testing the improved software component for graphs visualizing of the GOLEM framework.

Graphical annotation (Графическая аннотация)



Верхний рисунок:

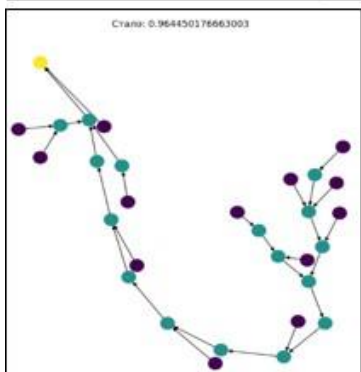
GOLEM — Open source фреймворк ИТМО, для оптимизации графовых структур.

Недостаток: графы изображаются с пересечениями ребер и наложением вершин.

Нижний рисунок:

улучшенный программный компонент визуализации графов фреймворка GOLEM.

Визуализация графов с меньшим числом пересечений ребер и наложений вершин на вершины в сравнении со стандартным компонентом визуализации GOLEM.



Top figure:

GOLEM — Open source ITMO framework, for optimising graph structures

Disadvantage: graphs are represented with edge intersections and overlapping vertices.

Bottom figure:

The improved graph visualisation software component of the GOLEM framework. Graph visualisation with fewer edge intersections and vertex-to-vertex overlaps compared to the standard GOLEM visualisation component.

ВВЕДЕНИЕ

Графы являются удобным и мощным средством представления и анализа различных видов данных, таких как социальные сети, биологические сети, транспортные сети, семантические сети и т. д. Для эффективного исследования и взаимодействия с графовыми данными необходимо иметь возможность визуализировать их в наглядной и понятной форме. Визуализация графов позволяет выявлять структурные свойства, паттерны, аномалии и взаимосвязи в графовых данных, а также облегчает коммуникацию и демонстрацию результатов анализа графов.

Во многих областях применяется визуализация графов без пересечений, чтобы лучше видеть и изучать данные. Это может быть полезно в таких сферах, как обработка данных, исследование социальных сетей, картография и биоинформатика. В форме графов, которые оптимизирует GOLEM, могут быть выражены описательные модели (уравнения), модели на данных (пайплайны

машинного обучения, баэсовские сети) и даже реальные объекты. Визуализация графов без пересечений дает возможность ясно показать структуру графа, что необходимо для осознания характеристик графа и удобно для демонстрации результатов исследований учеными-исследователями, студентами, научно-техническими работниками.

Для того чтобы визуализировать графы в удобной и понятной форме, необходимо учитывать многочисленные исследования по этой теме. Каждый год на международном симпозиуме по визуализации графов (International Symposium on Graph Drawing and Network Visualization) публикуются работы по этому направлению [1, 2, 3]. В 2013 г. вышла монография «Handbook of Graph Drawing and Visualization» [4], в которой системно изложены разные аспекты визуализации графов, включая разнообразные алгоритмы визуализации графов и библиотеки инструментов для визуализации графов. Существуют специальные алгоритмы для визуализации определенных типов графов, но для визуализации графов общего вида выделяются три класса алгоритмов:

- плоское прямолинейное изображение (Planar straight-line drawings) с минимизацией числа пересечений рёбер;
- силовые алгоритмы (Force-directed algorithms);
- многомасштабные методы (Multiscale methods).

В монографии подробно рассмотрены такие инструменты визуализации графов, как Open Graph Drawing Framework [4, с. 543–569], GDToolkit [4, с. 571–597], PIGALE (Public Implementation of a Graph Algorithm Library and Editor) [4, с. 599–620]. В настоящее время библиотека networkx стала популярным инструментом для визуализации графов, который достаточно подробно описывается в отдельных монографиях [5, 6]. В фреймворке GOLEM для работы с графами используется библиотека networkx [7]. На странице с описанием фреймворка GOLEM приводятся ссылки на пять статей для более подробного знакомства с применением фреймворка GOLEM [8–12]. Однако в этих статьях GOLEM даже не упоминается, а в статье о пропагаторе Флоке не упоминаются графы, хотя GOLEM предназначен для оптимизации графовых структур. Для размещения вершин графа при визуализации в библиотеке networkx используются силовые алгоритмы визуализации графов, которые используют целевые функции, основанные на физической аналогии (между смежными вершинами действуют силы, способствующие более близкому их размещению, а между несмежными вершинами действуют силы, способствующие более далёкому их размещению). Свежий обзор силовых алгоритмов визуализации графов был дан в статье [13]. В библиотеке networkx имеется функция `spring_layout`, реализующая силовой алгоритм Фрухтермана-Рейнгольда (Fruchterman-Reingold) [14].

Целью данной статьи является представление улучшенного программного компонента визуализации графов фреймворка GOLEM, который позволяет получать более качественные изображения графов.

Для достижения этой цели необходимо решить следующие задачи:

1. Улучшенный программный компонент визуализации графов должен обеспечивать не большее (по возможности, меньшее) количество пересечений ребер графа, чем существующий компонент визуализации фреймворка GOLEM, по визуальному контролю.
2. Улучшенный программный компонент визуализации графов должен обеспечивать не большее (по возможности, меньшее) количество наложений вершин друг на друга, чем существующий компонент визуализации фреймворка GOLEM, по визуальному контролю.
3. Улучшенный программный компонент визуализации графов должен обеспечивать не большее (по возможности, меньшее) количество пересечений ребер графа, чем существующий компонент визуализации фреймворка GOLEM, по геометрическому контролю на основе координат.
4. Улучшенный программный компонент визуализации графов должен обеспечивать не большее (по возможности, меньшее) количество наложений вершин друг на друга, чем существующий компонент визуализации фреймворка GOLEM, по геометрическому контролю на основе координат.
5. Задержка визуализации должна составлять не более 5 секунд по сравнению с существующей задержкой компонента визуализации фреймворка GOLEM.

АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ

Задача компоновки графа решается силовыми алгоритмами, которые определяют притягивающие и отталкивающие силы между вершинами и ищут оптимальную компоновку, минимизируя энергию системы.

В модели Фрухтермана и Рейнгольда, называемой пружинно-электрической моделью, действуют две силы. Сила отталкивания f_r существует между любыми двумя вершинами i и j и обратно пропорциональна расстоянию между ними. Притягивающая сила f_a существует только между соседними вершинами и пропорциональна квадрату расстояния между ними:

$$f_r(i, j) = -CK^2 / \|x_i - x_j\|, i \neq j, i, j \in V, \quad (1)$$

$$f_a(i, j) = \|x_i - x_j\|^2 / K, i \leftrightarrow j. \quad (2)$$

Следовательно, общая сила, действующая на вершину i , равна

$$f(i, x, K, C) = \sum_{i \neq j} - \frac{CK^2}{\|x_i - x_j\|^2} (x_j - x_i) + \sum_{i \leftarrow j} \frac{\|x_i - x_j\|}{K} (x_j - x_i). \quad (3)$$

В данных формулах K – параметр, известный как оптимальное расстояние или естественная длина пружины. Параметр C регулирует относительную величину отталкивающих и притягивающих сил.

Легко заметить, что для графа из двух вершин, соединенных ребром, сила, действующая на каждую вершину уменьшается, когда расстояние между ними равно $K(C)^{1/3}$.

Энергия системы может быть выражена как

$$Energy_{sc}(x, K, C) = \sum_{i \in V} f^2(i, x, K, C), \quad (4)$$

где вектор координат $x = \{x_i | i \in V\}$.

С математической точки зрения, изменение параметров K и C фактически не меняет минимальную энергетическую схему графа, а лишь масштабирует ее.

Реализация силового алгоритма Фрухтермана – Рейнгольда имеется в функции *spring_layout* из *networkx*. Алгоритм Фрухтермана – Рейнгольда будет генерировать различные размещения графов, а генетический алгоритм будет подбирать параметры для алгоритма Фрухтермана – Рейнгольда (количество итераций, заправка).

С помощью генетического алгоритма будет минимизироваться функция качества визуализации графа, оценивающая количество пересечений ребер и равномерность распределения вершин по площади.

Для подсчета количества пересечений ребер графа используется алгоритм Бентли – Оттманна в указанной реализации [19].

С использованием алгоритма проверки пересечений ребер делается подсчет количества пересечений ребер. Функция подсчета возвращает четные целые числа, указывающие количество пересечений (считаются пересечения по каждому ребру, поэтому каждая точка пересечения учитывается по два раза, в результате получаются четные числа), и далее используются в качестве составной части фитнес-функции в генетическом алгоритме. Вторая составляющая часть фитнес-функции будет предназначена для сравнения графов с одинаковым количеством пересечений ребер. Такие графы будут сравниваться между собой по равномерности расстояний между вершинами.

Равномерность расстояний между вершинами вычисляется как $1 - R_{min}/R_{max}$, где R_{min} – минимальное расстояние между вершинами, а R_{max} – максимальное расстояние между вершинами.

Расстояние R считается как евклидово расстояние между двумя точками:
 $R = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.

В итоге фитнес-функция определяется формулой:

$$\text{фитнес} = 2 \cdot \text{количество_пересечений_ребер} + 1 - R_{min}/R_{max}.$$

При такой формуле всегда считается более важным меньшее количество пересечений ребер, а следующим по важности фактором считается равномерность распределения вершин по площади, что обеспечивает меньшее количество наложений вершин друг на друга и меньшее количество наложений вершин на ребра.

СТРУКТУРА ПРОГРАММЫ

Улучшенный программный компонент состоит из двух модулей.

Модуль 1 – модуль визуализации с улучшениями.

Данный модуль содержит следующие функции:

а) для вычисления числа пересечений *def count_intersect(G, pos)*.

Аргумент G – граф в формате *networkx*, аргумент *pos* – позиции вершин графа, словарь, в котором ключами являются номера вершин графа, а значениями – пары координат x и y для соответствующих вершин;

б) для определения пропорции расстояний между вершинами графа *R_min/R_max def distance_proportion(pos)*.

Аргумент *pos* – позиции вершин графа, словарь, в котором ключами являются номера вершин графа, а значениями – пары координат x и y для соответствующих вершин. Функция возвращает действительное число от 0 до 1;

в) для определения качества визуализации графа *def vis_quality(G, pos)*. Аргумент *pos* – позиции вершин графа, словарь, в котором ключами являются номера вершин графа, а значениями – пары координат x и y для соответствующих вершин;

г) генерация наилучшей визуализации с помощью генетического алгоритма *def generate_best_viz(pos)*.

Аргумент *pos* содержит некоторое ранее сгенерированное размещение вершин графа и пытаются сгенерировать размещение, обеспечивающее лучшую визуализацию, чем переданное в аргументе размещение.

Модуль 2 – модуль проверки визуализации.

Модуль содержит функцию *test_viz*, которая показывает 2 изображения графа: одно сгенерировано с использованием стандартных средств GOLEM, а второе сгенерировано с использованием функции *generate_best_viz*.

Структура и форма представления данных представлены в таблицах 1–2.

Таблица 1 – Входные данные

Наименование данных	Обозначение	Структура данных	Способ ввода данных	Ограничения
Данные для тестирующего модуля: графы, на которых при визуализации в стандартном модуле получались пересечения или наложения	$G_i = (V_i, E_i)$	Список ребер, содержащий кортежи с парами вершин	Текстовый файл	$ V_i \leq 30$

Таблица 2 – Выходные данные

Наименование данных	Обозначение	Структура данных	Способ вывода данных	Ограничения
Число пересечений ребер в графе	EE_i	Список с количеством пересечений ребер для каждого графа	CSV-файл	Нет
Число наложений вершин на вершины в графе	V_i	Список с количеством наложений вершин на вершины для каждого графа	CSV-файл	Нет
Изображения графов	G_i	Объект matplotlib	PNG-файл	Разрешение до 2000x2000 px

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

На рисунке 1 приведен запуск автоматических тестов, демонстрирующий успешную работу модульных и интеграционных тестов.

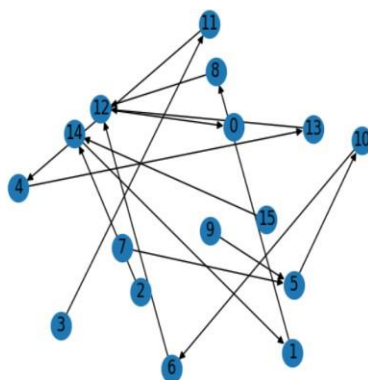


Рисунок 1 – Запуск модульного тестирования

Запустив модуль *gui_demo.py* с 30 вершинами, видим, что улучшенный компонент имеет меньше пересекающихся ребер, что демонстрируют рисунки 2 и 3.

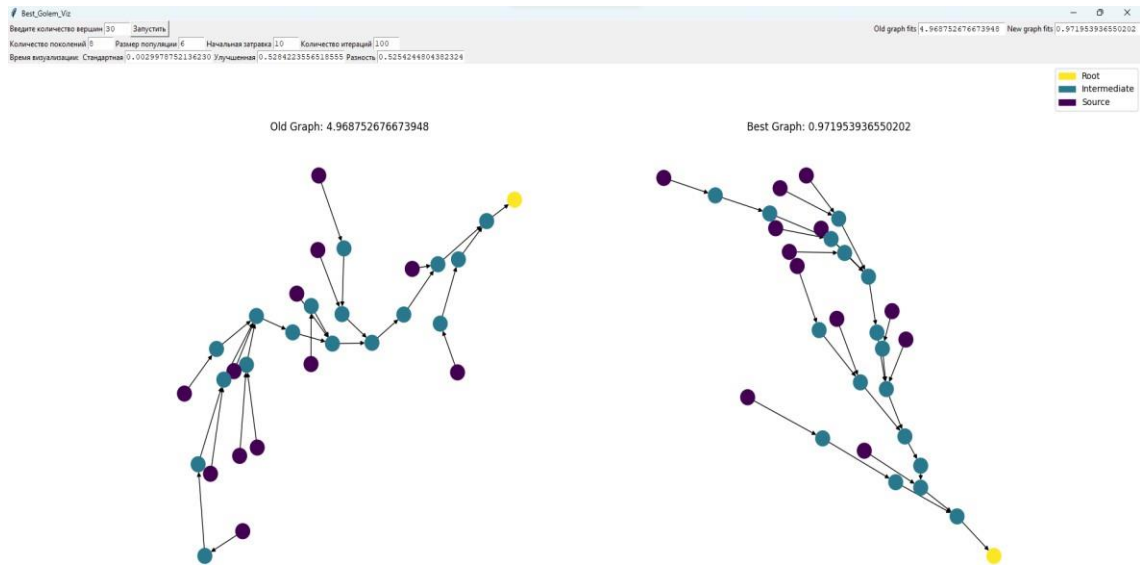


Рисунок 1 – Сравнение Best Graph и Old Graph – лучший результат только по пересечению ребер

На рисунках 4 и 5 количество пересечений ребер одинаково для обоих компонентов.

Улучшенный компонент также характеризуется меньшим наложением вершин, что видно на рисунках 3 и 4.

На рисунке 2 наложение вершин не является оптимальным, но зато минимально количество пересечений ребер.

На рисунке 5 визуально оба компонента выглядят одинаково.

По геометрическому контролю улучшенный компонент также показывает меньше пересечений ребер графа, как на рисунке 2. Целая часть фитнес-функции для Best Graph (0) меньше, чем для Old Graph (4). Это подтверждается и на рисунке 3: целая часть фитнес-функции Old Graph (4), а Best Graph (0).

На рисунках 4 и 5 целая часть фитнес-функции одинакова для обоих компонентов (0). По геометрическому контролю улучшенный компонент также не имеет большего наложения вершин при равном количестве пересечений.

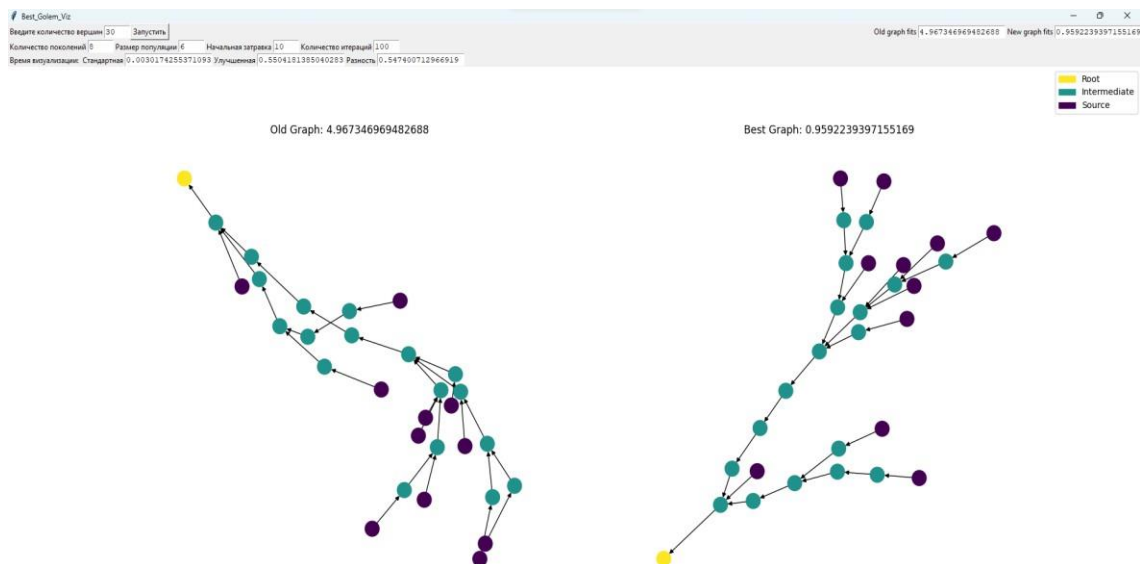


Рисунок 2 – Сравнение Best Graph и Old Graph – лучший результат и по пересечению ребер, и по наложению вершин друг на друга

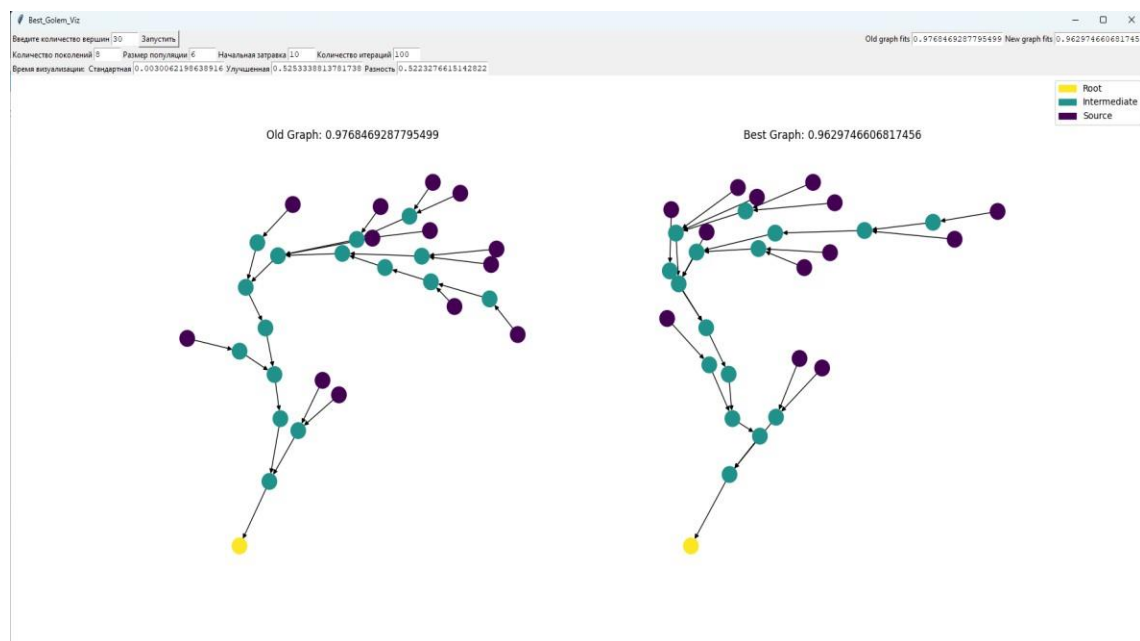


Рисунок 3 – Сравнение Best Graph и Old Graph – лучший результат только по наложению вершин друг на друга

На рисунке 2 дробная часть фитнес-функции для Old Graph – 0,968752676673948, а для Best Graph – 0,971953936550202, но это объясняется тем, что в приоритете меньшее число пересечений.

На рисунке 3 дробная часть фитнес-функции для Best Graph (0,9592239397155169) меньше, чем для Old Graph (0,967346969482688).

На рисунке 4 дробная часть фитнес-функции для Best Graph (0,9629746606817456) меньше, чем для Old Graph (0,9768469287795499).

На рисунке 5 дробные части фитнес-функции для Best Graph и Old Graph одинаковы (0,9652608629130555).

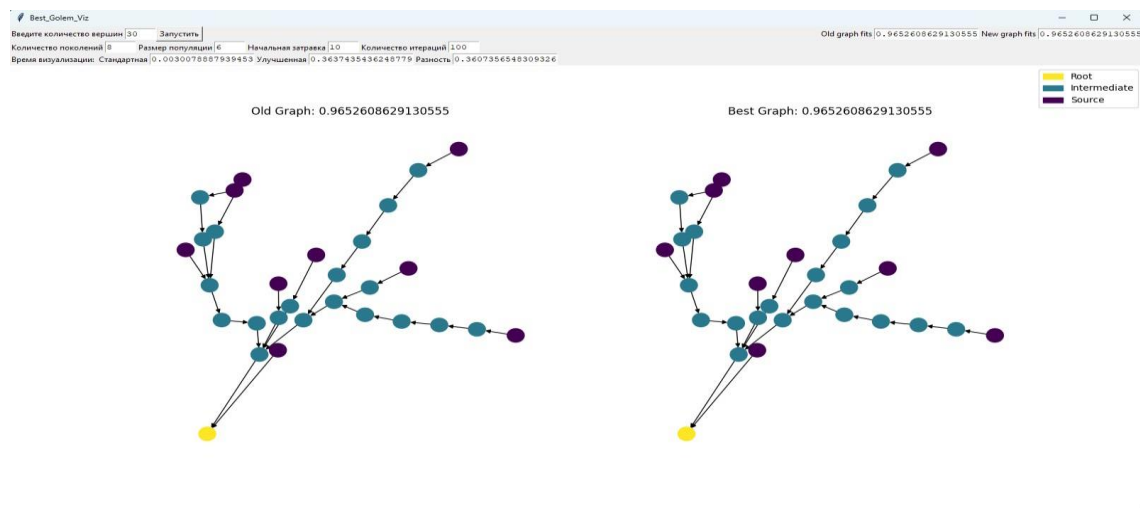


Рисунок 4 – Сравнение Best Graph и Old Graph – одинаковый результат и по пересечению ребер, и по наложению вершин друг на друга

ЗАКЛЮЧЕНИЕ

В рамках данной работы был разработан улучшенный программный компонент визуализации графов фреймворка GOLEM. В его основе лежит алгоритм Фрухтермана – Рейнгольда, который генерирует различные размещения графов, а генетический алгоритм подбирает параметры для алгоритма Фрухтермана – Рейнгольда, такие как количество итераций, затравка. После тщательного тестирования показано, что улучшенный программный компонент выигрывает по качеству визуализации графов на плоскости.

Список источников

1. Purchase, H. C., Rutter, I. (Eds.) // *Graph Drawing and Network Visualization : 29th International Symposium, GD 2021*. – Cham, Switzerland: Springer Nature Switzerland AG, 2021. – ISBN 978-3-030-92931-2.
2. Angelini, P., von Hanxleden, R. (Eds.) // *Graph Drawing and Network Visualization : 30th International Symposium, GD 2022*. Cham, Switzerland: Springer Nature Switzerland AG, 2022. – ISBN 978-3-031-22202-3.
3. Proceedings of the 31st International Symposium on Graph Drawing and Network Visualization (GD 2023) // [arxiv.org. 11.09.2023](https://arxiv.org/abs/2309.07166). – URL: <https://arxiv.org/html/2309.07166> (дата обращения: 02.11.2023).
4. Tamassia, R. (Ed.). *Handbook of Graph Drawing and Visualization / R. Tamassia (Ed.)*. – New York : Chapman and Hall/CRC, 2013. – 862 p.
5. Platt, E. L. *Network Science with Python and NetworkX Quick Start Guide / E. L. Platt*. – Birmingham : Packt Publishing, 2019. – ISBN 978-1-78995-531-6.
6. Ma E. J., Seth M. *Network Analysis Made Simple: An introduction to network analysis and applied graph theory using Python and NetworkX / E. J. Ma, M. Seth*. – Lean Publishing, 2021.
7. aimclub.GOLEM/README.rstatmainaimclub/GOLEM // GitHub. – URL: <https://github.com/aimclub/GOLEM/blob/main/README.rst> (дата обращения: 05.09.2023).
8. Nikitin, N. O. Automated evolutionary approach for the design of composite machine learning pipelines / N. O. Nikitin et al. // *Future Generation Computer Systems*. – 2022. – Vol. 127. – P. 109–125.
9. Sarafanov, M. Evolutionary Automated Machine Learning for Multi-Scale Decomposition and Forecasting of Sensor Time Series / M. Sarafanov, V. Pokrovskii, N. O. Nikitin // *2022 IEEE Congress on Evolutionary Computation (CEC)*. – 2022. – P. 1–8.
10. Hvatov, A. Data-Driven Approach for the Floquet Propagator Inverse Problem Solution / A. Hvatov // *ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. – 2022. – P. 3813–3817.
11. Maslyaev, M. Solver-Based Fitness Function for the Data-Driven Evolutionary Discovery of Partial Differential Equations / M. Maslyaev, A. Hvatov // *2022 IEEE Congress on Evolutionary Computation (CEC)*. – 2022. P. 1–8.
12. Deeva, I. Adaptive Learning Algorithm for Bayesian Networks Based on Kernel Mixtures Distributions / I. Deeva, A. V. Kalyuzhnaya, A. V. Boukhanovsky // *International Journal of Artificial Intelligence*. – 2023. – Vol. 21, № 1. – P. 90.
13. Cheong, S.-H. Force-directed algorithms for schematic drawings and placement: A survey / S.-H. Cheong, Y.-W. Si // *Information Visualization*. – 2020. – Vol. 19, № 1. – P. 65–91.
14. `spring_layout` – NetworkX 3.1 documentation. – URL: https://networkx.org/documentation/stable/reference/generated/networkx.drawing.layout.spring_layout.html (дата обращения: 05.09.2023).
15. `ogdf-python`. – URL: <https://pypi.org/project/ogdf-python/> (дата обращения: 05.09.2023).
16. `networkx`. – URL: <https://pypi.org/project/networkx/> (дата обращения: 05.09.2023).
17. Graph Drawing Tool. – URL: <http://www.dia.uniroma3.it/~gdt/gdt4/index.php> (дата обращения: 05.09.2023).
18. Pigale. – URL: <https://pigale.sourceforge.net/> (дата обращения: 05.09.2023).
19. `prochitecture/sweep_intersector` // GitHub. – URL: https://github.com/prochitecture/sweep_intersector (дата обращения: 05.09.2023).

References

1. Purchase, H. C., & Rutter, I. (Eds.). *Graph Drawing and Network Visualization: 29th International Symposium, GD 2021*. Cham, Switzerland, Springer Nature Switzerland AG, 2021. ISBN 978-3-030-92931-2.
2. Angelini, P., & von Hanxleden, R. (Eds.). *Graph Drawing and Network Visualization: 30th International Symposium, GD 2022*. Cham, Switzerland, Springer Nature Switzerland AG, 2022. ISBN 978-3-031-22202-3.
3. Proceedings of the 31st International Symposium on Graph Drawing and Network Visualization (GD 2023). *arxiv.org* (2023, September 11). URL: <https://arxiv.org/html/2309.07166> (accessed 02.11.2023).
4. Tamassia, R. (Ed.). *Handbook of Graph Drawing and Visualization*. New York, Chapman and Hall/CRC, 2013. 862 p.
5. Platt, E. L. *Network Science with Python and NetworkX Quick Start Guide*. Birmingham, Packt Publishing, 2019. ISBN 978-1-78995-531-6.
6. Ma, E. J., & Seth, M. *Network Analysis Made Simple: An introduction to network analysis and applied graph theory using Python and NetworkX*. Lean Publishing, 2021.
7. aimclub.GOLEM/README.rstatmainaimclub/GOLEM. *GitHub*. URL: <https://github.com/aimclub/GOLEM/blob/main/README.rst> (accessed 05.09.2023).
8. Nikitin, N. O., Sarafanov, M., Pokrovskii, V., Hvatov, A., Maslyaev, M., Deeva, I., & Boukhanovsky, A. V. Automated evolutionary approach for the design of composite machine learning pipelines. *Future Generation Computer Systems*, 2022, vol. 127, pp. 109–125.
9. Sarafanov, M., Pokrovskii, V., & Nikitin, N. O. Evolutionary Automated Machine Learning for Multi-Scale Decomposition and Forecasting of Sensor Time Series. *2022 IEEE Congress on Evolutionary Computation (CEC)*, 2022, pp. 1–8.
10. Hvatov, A. Data-Driven Approach for the Floquet Propagator Inverse Problem Solution. *ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 3813–3817.
11. Maslyaev, M., & Hvatov, A. Solver-Based Fitness Function for the Data-Driven Evolutionary Discovery of Partial Differential Equations. *2022 IEEE Congress on Evolutionary Computation (CEC)*, 2022, pp. 1–8.
12. Deeva, I., Kalyuzhnaya, A. V., & Boukhanovsky, A. V. Adaptive Learning Algorithm for Bayesian Networks Based on Kernel Mixtures Distributions. *International Journal of Artificial Intelligence*, 2023, vol. 21 (1), p. 90.
13. Cheong, S.-H., & Si, Y.-W. Force-directed algorithms for schematic drawings and placement: A survey. *Information Visualization*, 2020, vol. 19 (1), pp. 65–91.

14. *spring_layout – NetworkX 3.1 documentation*. URL: https://networkx.org/documentation/stable/reference/generated/networkx.drawing.layout.spring_layout.html (accessed 05.09.2023).
15. *ogdf-python*. URL: <https://pypi.org/project/ogdf-python/> (accessed 05.09.2023).
16. *networkx*. URL: <https://pypi.org/project/networkx/> (accessed 05.09.2023).
17. *Graph Drawing Tool*. URL: <http://www.dia.uniroma3.it/~gdt/gdt4/index.php> (accessed 05.09.2023).
18. *Pigale*. URL: <https://pigale.sourceforge.net/> (accessed 05.09.2023).
19. *prochitecture/sweep_intersector*. *GitHub*. URL: https://github.com/prochitecture/sweep_intersector (accessed 05.09.2023).

Статья поступила в редакцию 01.03.2024; одобрена после рецензирования 18.03.2024; принята к публикации 01.04.2024.

The article was submitted 01.03.2024; approved after reviewing 18.03.2024; accepted for publication 01.04.2024.

УДК: 004.043, 004.42

ИНФОРМАЦИОННАЯ СИСТЕМА ФОРМИРОВАНИЯ ВЕДОМОСТЕЙ УЧЕТА РАБОТЫ ПРЕПОДАВАТЕЛЕЙ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ

Калинина Юлия Юрьевна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
студент, ORCID: 0009-0009-8785-4634, e-mail: jilietka@mail.ru

Смирнова Юлия Александровна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
старший преподаватель кафедры информационных технологий, ORCID: 0000-0002-3807-5062,
e-mail: got-22@mail.ru

Головко Юлия Александровна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
кандидат технических наук, доцент кафедры информационных технологий, ORCID: 0000-0001-5401-270X, email: pjulia2014@yandex.ru

В данной статье представлено исследование и методика формирования ведомостей учета образовательной работы преподавателя. Авторы статьи осуществили анализ существующих проблем в процессе формирования ведомостей и предложили решение на основе использования информационных технологий. Решением стала методика сбора, анализа и обработки информации с использованием цифровых технологий, которая упростит работу профессорско-преподавательского состава (ППС) и сократит время, затрачиваемое на составление документации преподавателя. Во время исследования было принято решение о создании базы данных, позволяющей сохранять всю информацию о сотрудниках высшего учебного заведения и упрощающей процесс извлечения информации. Под документацией в статье подразумевается: заявление-контракт, заявление на оплату, а также ведомость. Описывается структура и функциональные возможности предлагаемой методики, которая включает процессы автоматического сбора, анализа и обработки информации, генерации документов, а также возможности по взаимодействию с другими информационными системами учебного заведения. Методика позволит ППС эффективно вести учет своих занятий, получать актуальную информацию о процессе обучения и быстро формировать электронные ведомости. Основными преимуществами методики является уменьшение вероятности ошибок, повышение эффективности работы преподавателей и улучшение качества информирования представителей администрации. Предпосылками автоматизации данной методики являются: большие затраты по рабочему времени, трудовых и материальных ресурсов, наведение и контроль документов, поддержание данных в достоверном состоянии; неизбежно большое количество ошибок и опечаток при проведении выборки необходимых сведений и подготовки данных к различным отчетам. Возможность взаимодействия предлагаемой методики с информационными системами учебного заведения позволит интегрировать систему в общую инфраструктуру и обеспечить ее совместную работу с различными системами, такими как система управления учебным процессом или система электронного документооборота.

Ключевые слова: информационные технологии, инновации, информационная система, документооборот, автоматизация, системы мониторинга, работа преподавателя, программа на Python, ведомости учета, база данных

INFORMATION SYSTEM FOR FORMATION OF WORK REPORTS OF TEACHERS OF HIGHER EDUCATIONAL INSTITUTIONS

Kalinina Yulia Yu., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
student, ORCID: 0009-0009-8785-4634, e-mail: jilietka@mail.ru

Smirnova Yulia A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Senior Lecturer of the Department of Information Technology, ORCID: 0000-0002-3807-5062,
e-mail: got-22@mail.ru

Golovko Yulia A., Astrakhan Tatishchev State University, 20a Tatishchev St., Astrakhan, 414056, Russian Federation,
Cand. Sci. (Engineering), Associate Professor, ORCID: 0000-0001-5401-270X, email: pjulia2014@yandex.ru

This article presents a research and development of an information system designed for the compilation of teacher's records. The authors of the article analyzed the existing problems in the process of record compilation and proposed an efficient solution based on the use of information technologies. The solution involved a methodology for data collection and processing using digital technologies, which simplifies the work of teachers and reduces the time spent on documentation. Additionally, a database was created to store all the information about the employees of the higher educational institution, facilitating information retrieval. The concept of documentation includes contract applications,

payment applications, and records. The article describes the structure and functional capabilities of the proposed methodology, which includes processes of automatic data collection, analysis and processing, document generation, as well as the ability to interact with other information systems of the educational institution. The proposed methodology will enable teachers to easily and efficiently manage their classes, obtain up-to-date information about the learning process, and quickly generate records. The main advantages of the methodology will be process automation, reduced error probability, increased efficiency of teachers' work, and improved communication of information to the administration. The ability of the proposed methodology to interact with the information systems of the educational institution will allow for integration into the overall infrastructure and ensure its collaboration with various systems, such as the educational process management system or the electronic document workflow system.

Keywords: information technologies, innovations, information system, document management, automation, monitoring systems, teacher's work, Python program, accounting statements, database

ВВЕДЕНИЕ

В соответствии с Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» [1], приказом Министерства образования и науки РФ от 22.12.2014 № 1601 «О продолжительности рабочего времени (нормах часов педагогической работы за ставку заработной платы) педагогических работников и о порядке определения учебной нагрузки педагогических работников, оговариваемой в трудовом договоре» [2], а также Постановлением Минтруда РФ от 30.06.2003 № 41 «Об особенностях работы по совместительству педагогических, медицинских, фармацевтических работников и работников культуры» [3], педагогическая работа на условиях почасовой оплаты устанавливается в определенном объеме и для ее осуществления преподаватель заключает с университетом трудовые отношения. Формирование таких отношений является однозначной последовательностью действий, каждый шаг которой прописан.

АНАЛИЗ ТРАДИЦИОННОГО ДОКУМЕНТООБОРОТА

Анализ современного способа составления документации показал, что существующие методы и инструменты для формирования ведомостей часто оказываются недостаточно эффективными и удобными [4]. Так, многие учебные заведения до сих пор используют бумажную документацию или устаревшие программные системы, что является затратным как по времени, так и по ресурсам. Более того, возрастает риск потери, повреждения документов или типичных ошибок вроде опечаток. В результате формализованный процесс нарушается, возникает ряд проблем с единообразием, правописанием, расчётами. Цифровая трансформация интегрируется во все сферы деятельности, привычные методы устаревают и становятся все менее эффективным подходом [5].

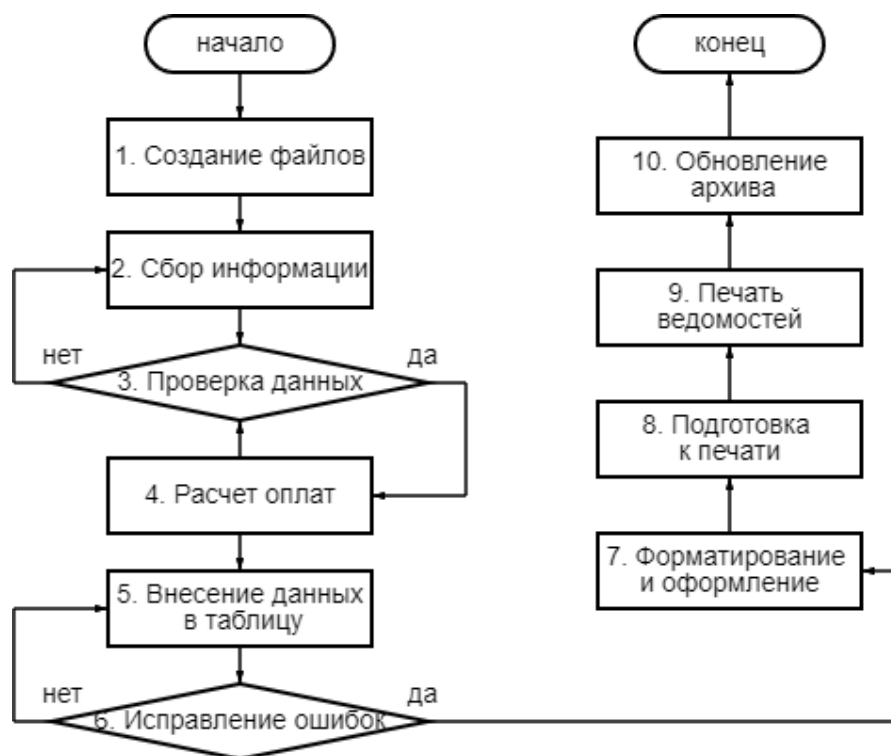


Рисунок 1 – Процесс формирования документации вручную

Рассмотрим процесс формирования документации преподавателем вручную (рис. 1):

1. Создание файлов. На этом этапе ППС, используя шаблоны, создает файлы для работы с ними в дальнейшем.
 2. Сбор информации. Преподаватель анализирует свое расписание: группы, количество пар в день, в неделю, в месяц и выбирает необходимую для заполнения документации информацию.
 3. Проверка данных. Собранные в предыдущем пункте информацию преподаватель перепроверяет. Это необходимо для минимизации количества ошибок на последующих этапах создания документации.
 4. Расчет оплат. Преподаватель вручную рассчитывает часы своей работы и их оплату, учитывая все пары и виды работ на парах.
 5. Внесение данных в таблицу. После расчета оплаты преподаватель вводит информацию в подготовленные на первом этапе шаблоны.
 6. Исправление ошибок. На этом этапе происходит финальное исправление ошибок, который преподаватель мог пропустить на втором этапе или которые могли возникнуть во время последующих.
 7. Форматирование и оформление. Преподаватель вручную форматирует файлы и оформляет таблицы в соответствии с ГОСТ.
 8. На этапе подготовки файлов к печати происходит финальное исправление ошибок и форматирование таблиц.
 9. Печать ведомостей. На этом этапе преподаватель печатает документацию.
 10. Обновление архива. Преподавателю необходимо отправлять документацию заведующему кафедрой, а также хранить копию у себя на случай непредвиденных обстоятельств. Заведующий кафедрой обновляет архив, в котором хранятся ведомости каждого преподавателя и сохраняет его.
- Анализ создания документации вручную показал, что это сложный процесс, который затрачивает много времени и сил преподавателя.
- После анализа блок-схемы рассмотрим каждый из подпроцессов с точки зрения их длительности. Для этого составим таблицу (табл. 1) по результатам наблюдения за работой преподавателя во время создания документации.

Таблица 1 – Процесс формирования документации

Процесс	Затраченное время, мин.
Создание файлов	5
Сбор информации	44
Проверка данных	12
Расчет оплат	57
Внесение данных в таблицу	32
Исправление ошибок	13
Форматирование и оформление	21
Подготовка файлов к печати	7
Печать ведомостей	4
Обновление архива	28
Итого	223

По информации из таблицы составим диаграмму (рис. 2), представляющую собой визуальное изображение последовательности этапов, которые преподаватель проходит при формировании ведомостей и времени, в процентном соотношении, затрачиваемом на каждый из них.

Используя систему шаблонов, можно полностью отказаться от следующих этапов: создание файлов, расчёт оплат, внесение данных в таблицу, форматирование и оформление, подготовка к печати, так как будут использоваться уже готовые образцы. Время, затраченное на этапы сбора информации, проверки данных и исправления ошибок, сократится.

Формирование документации



Рисунок 2 – Диаграмма «Формирование ведомостей»

Помимо системы шаблонов для автоматизации процесса формирования документации преподавателя было принято решение о создании базы данных. В учебных заведениях они помогают эффективно управлять информацией о структуре учебного процесса и персонале.

МЕТОДИКА ФОРМИРОВАНИЯ ВЕДОМОСТЕЙ УЧЕТА ОБРАЗОВАТЕЛЬНОЙ РАБОТЫ ПРЕПОДАВАТЕЛЯ

Автоматизация процесса составления документации для преподавателей имеет ряд преимуществ. Во-первых, она сокращает время, затрачиваемое на каждый этап процесса [6–8]. Это позволяет преподавателям сэкономить время и сосредоточиться на других важных задачах.

Во-вторых, автоматизация снижает возможность ошибок. Ручной ввод информации и ручные расчеты могут быть подвержены ошибкам, особенно при большом объеме данных. Использование автоматизированных средств позволяет минимизировать риски возникновения ошибок и повысить точность данных.

В-третьих, создание базы данных для хранения информации обеспечивает более эффективное и удобное управление данными. База данных позволяет хранить информацию в структурированном и организованном виде, что облегчает доступ к данным и управление ими. Преподавателям не придется искать и собирать информацию в разных файлах или документах – все данные будут легко доступны в базе данных.

Кроме того, база данных позволяет быстро извлекать нужные данные и обновлять архивы [9]. Преподаватели смогут легко получать актуальные данные и использовать их при составлении документации. Это способствует более гибкой и эффективной работе и позволяет сократить временные и ресурсные затраты.

Таким образом, создание методики автоматизации процесса составления документации и использование базы данных являются необходимыми мерами для повышения эффективности и точности работы преподавателей, сокращения времени и усилий, а также облегчения доступа к управлению информацией.

Целью работы является создание методики формирования ведомостей учета преподавателя, которая позволит автоматизировать процесс создания и обработки документов. Эта система будет предназначена для значительного упрощения и ускорения работы преподавателей и административного персонала, связанной с формированием и обработкой ведомостей учета.

ИНФОРМАЦИОННАЯ МОДЕЛЬ

Использование базы данных для хранения информации о структуре образовательного учреждения, дисциплинах, учебных группах и сотрудниках позволит максимально оперативно реагировать на все изменения, корректируя данные путём интуитивно понятных действий, без необходимости изменения самой программы [10].

Для отображения информационное модели использовалась ER-диаграмма с отображением сущностей и связей между ними (рис. 3). База данных стала ключевым инструментом для хранения и управления информацией, делая процесс формирования ведомостей более эффективным и менее подверженным ошибкам. Всего было разработано 9 сущностей.

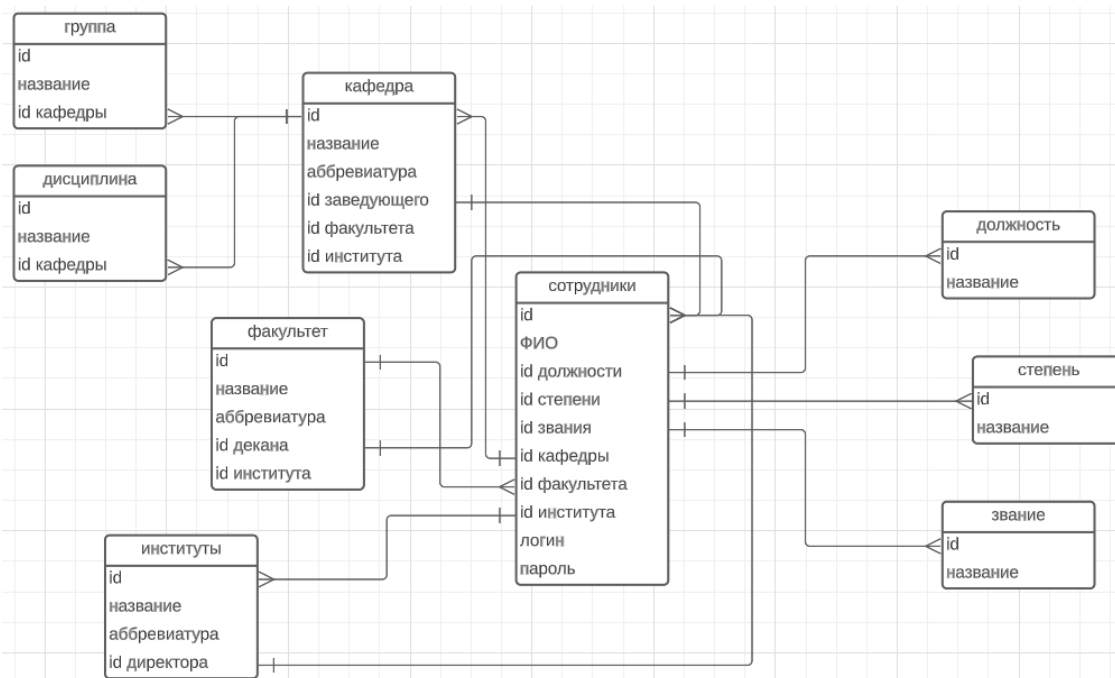


Рисунок 3 – ER-диаграмма базы данных

Создание базы данных позволило хранить информацию об учебных заведениях, их структуре, преподавателях, дисциплинах и других связанных с учетом данных в цифровой форме. База данных содержит таблицы с информацией о:

- кафедрах;
- сотрудниках;
- факультетах;
- группах;
- институтах;
- должностях;
- степенях;
- званиях преподавателей.

Кроме того, созданная база данных включает в себя таблицу с общим количеством записей в каждой из таблиц, что позволяет получать статистическую информацию без необходимости ручного подсчета.

База данных для сложных информационных систем должна быть спроектирована с учетом множества потребностей пользователей и требований к системе. В данном случае в основу легли следующие принципы:

1. Гибкость и масштабируемость для обеспечения возможности расширения функционала без изменения структуры.
2. Безопасность данных и защита от несанкционированного доступа.
3. Высокая производительность и эффективность для быстрого доступа к информации.
4. Интеграция с другими компонентами информационной системы для сохранения целостности данных.

Важным аспектом использования структурированных баз данных для информационных систем является полная автоматизация. Путем добавления, изменения и удаления данных в реальном времени достигается возможность автоматизированной генерации отчетов и ведомостей. Это позволяет существенно упростить и ускорить процесс работы преподавателей и администраторов системы, а также снизить вероятность ошибок в результате ручного вмешательства. Важно также отметить возможность развития данного направления, а именно автоматическое резервное копирование и восстановление данных для предотвращения потери информации в случае сбоев или аварий.

Использование базы данных в рамках предлагаемой методики автоматизации процесса формирования ведомостей учета работы преподавателей позволит улучшить качество создаваемой документации, а также избежать возможных ошибок, неточностей и неактуальной информации о структуре университета.

В рамках исследования необходимости разработки подобной системы были выдвинуты следующие преимущества и перспективы, которые она может предоставить:

1. Автоматизация процесса формирования ведомостей позволит существенно сократить время, затрачиваемое на эту задачу. Преподаватели смогут быстро и легко заполнять ведомости, используя готовые шаблоны и автоматические расчеты. Формализация документов при этом только возрастёт, так как снизится риск отхода от единого стандарта.

2. Информационная система позволит централизованно хранить и обрабатывать данные. Это упростит доступ к информации и обеспечит возможность анализа данных в различных ракурсах. Преподаватели смогут быстро получать необходимую информацию о студентах, успеваемости, посещаемости и других показателях.

3. Использование информационной системы позволит снизить риск ошибок при заполнении ведомостей. Система будет автоматически проверять данные на соответствие заданным правилам и выдавать предупреждения о возможных ошибках.

Главным результатом работы станет методика, которая включает в себя модули для создания ведомостей, ввода данных о студентах и предметах, автоматического подсчета и формирования отчетов, а также программный блок для работы со структурированными данными, включая их добавление, редактирование и изменение[11] (рис. 4).

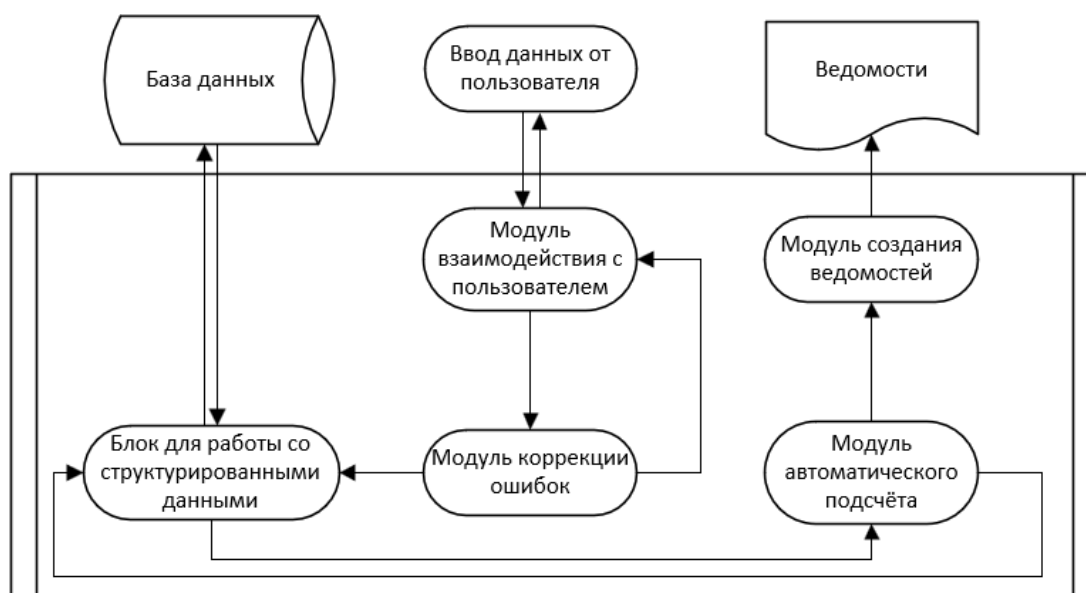


Рисунок 4 – Архитектура информационной системы

ЗАКЛЮЧЕНИЕ

Методика заключается в объединении нескольких модулей для обеспечения максимально полного функционала в эффективной и удобной для взаимодействия форме. Она позволит автоматизировать рутинные задачи по работе с документами, произвести минимальную проверку и коррекцию данных, а также обеспечить универсальность путём составления алгоритма, использующего минимальное количество взаимодействий с внешними объектами. Данная архитектура имеет ряд преимуществ, таких как простота обновления кода отдельных модулей, их масштабируемость и относительная независимость.

Эффективность предлагаемой методики будет состоять в значительном сокращении времени, затрачиваемом на формирование и обработку ведомостей, поиск, выборку и коррекцию необходимой информации. Это позволит сосредоточиться на преподавательской деятельности и сократить время, затрачиваемое на административные задачи.

Унификация и приведение имеющихся шаблонов к единому виду позволит расширить зону использования информационной системы и дать возможность другим учебным заведениям использовать её, чтобы стандартизировать весь процесс формирования ведомостей в масштабах государства. В свою очередь структура базы данных уже приведена к обобщенному виду, так как во всех учреждениях сохраняется сходное построение подразделений. Соответственно, изменение наполнения является единственно необходимым действием для начала взаимодействия с ней.

Список источников

1. Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» // СПС КонсультантПлюс.
2. Приказ Минобрнауки России «О продолжительности рабочего времени (нормах часов педагогической работы за ставку заработной платы) педагогических работников и о порядке определения учебной нагрузки педагогических работников, оговариваемой в трудовом договоре» от 22.12.2014 № № 1601 // СПС КонсультантПлюс. – С изм. и допол. в ред. от 13.05.2019.
3. Постановление Минтруда РФ от 30.06.2003 № 41 «Об особенностях работы по совместительству педагогических, медицинских, фармацевтических работников и работников культуры» // СПС КонсультантПлюс.
4. Умаров, А. А. Проектирование информационной системы управления вузом / А. А. Умаров // Современные тенденции в экономике и управлении: новый взгляд. – 2010. – № 1–2.
5. Матюшенко, В. А. Информационная система автоматизации подготовки документов учебного процесса / В. А. Матюшенко, В. Ю. Филон, Н. И. Белодед // Системный анализ и прикладная информатика. – 2016. – № 2.
6. Неретина, В. В. Автоматизированная система документооборота структурных подразделений учебных заведений / В. В. Неретина // Перспективы развития информационных технологий. – 2013. – № 13.
7. Семенов, С. М. Разработка технологии ведения электронного документооборота кафедры / С. М. Семенов // Территория новых возможностей. – 2011. – № 2 (11).
8. Торобеков, Б. Т. Автоматизация процессов приемной комиссии вузов на основе информационной системы / Б. Т. Торобеков // Наука, техника и образование. – 2017. – № 4 (34).
9. Аргучинцев, А. В. Платформа «1С: Предприятие» как основа построения современной корпоративной информационной системы вуза / А. В. Аргучинцев, В. С. Кедрин, Е. С. Чуйко // Известия Иркутского государственного университета. Серия: Политология. Религиоведение. – 2017.
10. Поплёвин, О. И. Автоматизированная система документооборота для ооо «автоматико-электротехнические системы» / О. И. Поплёвин, А. Г. Алёхин // Вестник магистратуры. – 2013. – № 9 (24).
11. Свидетельство о государственной регистрации программы для ЭВМ № 2023682139 Российская Федерация. Информационная система формирования заявления-контракта для преподавателей высших учебных заведений : № 2023680827 ; заявл. 10.10.2023 ; опублик. 23.10.2023 / Ю. Ю. Калинина, Ю. А. Смирнова, К. Е. Лебедева. – EDN DISSES.

References

1. Federal Law No. 273-FZ dated 12.29.2012 "On Education in the Russian Federation". *SPS ConsultantPlus*.
2. The order of the Ministry of Education and Science of the Russian Federation "On the duration of working hours (norms of hours of pedagogical work for the salary rate) of teaching staff and on the procedure for determining the workload of teaching staff, stipulated in the employment contract" dated 12.22.2014 No. 1601. *SPS ConsultantPlus*, with changes and additions in ed. dated 05.13.2019.
3. Resolution of the Ministry of Labor of the Russian Federation dated 30.06.2003 No. 41 "On the peculiarities of part-time work of pedagogical, medical, pharmaceutical and cultural workers". *SPS ConsultantPlus*.
4. Umarov, A. A. Designing an information system for university management. *Modern trends in economics and management: a new look*, 2010, no. 1–2.
5. Matyushenko, V. A., Filon, V. Yu., Beloded, N. I. Information system for automating the preparation of documents of the educational process. *System analysis and applied informatics*, 2016, no. 2.
6. Neretina, V. V. Automated document management system of structural divisions of educational institutions. *Prospects for the development of information technology*, 2013, no. 13.
7. Semenov, S. M. Development of technology for electronic document management of the department. *The territory of new opportunities*, 2011, no. 2 (11).
8. Torobekov, B. T. Automation of university admissions processes based on an information system. *Science, Technology and Education*, 2017, no. 4 (34).
9. Arguchintsev, A. V., Kedrin, V. S., Chuiko, E. S. 1C: Enterprise platform as the basis for building a modern corporate information system of the university. *Proceedings of Irkutsk State University. Series: Political Science. Religious studies*, 2017.
10. Poplevin, O. I., Alyokhin, A. G. Automated document management system for llc "automatic and electro-technical systems". *Bulletin of the magistracy*, 2013, no. 9 (24).
11. Kalinina, Yu. Yu., Smirnova, Yu. A. Lebedeva, K. E. Certificate of state registration of the computer program No. 2023682139 Russian Federation. Information system for the formation of a contract application for teachers of higher educational institutions : no. 2023680827 , appl. 10.10.2023 , publ. 23.10.2023. EDN DISSES.

Статья поступила в редакцию 08.04.2024; одобрена после рецензирования 26.04.2024; принята к публикации 06.05.2024.

The article was submitted 08.04.2024, approved after reviewing 26.04.2024; accepted for publication 06.05.2024.

ПРИБОРОСТРОЕНИЕ, МЕТРОЛОГИЯ И ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ ПРИБОРЫ И СИСТЕМЫ

ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ И УПРАВЛЯЮЩИЕ СИСТЕМЫ

УДК 004.9

РАЗВИТИЕ ИНФОРМАЦИОННЫХ СИСТЕМ ДЛЯ ЛОКАЛЬНЫХ ЭЛЕКТРОЭНЕРГЕТИЧЕСКИХ ОБЪЕКТОВ

Деримедведь Константин Григорьевич, АНО ВО «Российский новый университет», 105005, Российская Федерация, г. Москва, ул. Радио, 22, аспирант, ORCID: 0009-0009-0507-8441, e-mail: mmm99@mail.ru

В статье разработаны предложения по развитию информационных систем для локальных электроэнергетических объектов. Установлено, что для обеспечения необходимой эффективности систем электроснабжения может потребоваться формирование информационно-управляющих систем в рамках локальных электрообъектов, которые учитывают особенности некоторых задач и обеспечивают решение разноплановых сложностей, принципы, методы и модели, составляющие инструментарий программных средств на уровне автоматизированной системы дистанционного обеспечения. При этом анализ тенденций построения автоматизированных систем управления свидетельствует о децентрализации управления, приводящей к появлению проблем, которые необходимо решать для обеспечения надежности электроснабжения. Доказано, что создание информационно-управленческих систем должно осуществляться в виде системы открытого типа, реализация которой при относительно незначительных затратах повышает эффективность автоматизированного управления объектами и элементами электроэнергетической системы.

Ключевые слова: информация, система, электроэнергия, управление, электроснабжение, модель, обеспечение, эффективность, автоматизация, надежность

DEVELOPMENT OF INFORMATION SYSTEMS FOR LOCAL ELECTRIC POWER FACILITIES

Derimedved Konstantin G., Russian New University, 105005, Russian Federation, Moscow, stRadio, 22, postgraduate student, ORCID: 0009-0009-0507-8441, e-mail: mmm99@mail.ru

The article develops proposals for the development of information systems for local electric power facilities. It is established that in order to ensure the necessary efficiency of power supply systems, it may be necessary to form information and control systems within the framework of local electrical facilities that take into account the specifics of some tasks and provide solutions to diverse complexities, principles, methods and models that make up the software tools at the level of an automated remote support system. At the same time, the analysis of trends in the construction of automated control systems indicates the decentralization of management, leading to problems that need to be solved to ensure the reliability of power supply. It is proved that the creation of information management systems should be carried out in the form of an open-type system, the implementation of which, at relatively low costs, increases the efficiency of automated management of facilities and elements of the electric power system.

Keywords: information, system, electricity, management, power supply, model, provision, efficiency, automation, reliability

ВВЕДЕНИЕ

Совершенствование автоматизированного управления локальными системами объектов и в целом электроэнергетической системой (ЭЭС) означает процесс, обусловленный изменением объектов ЭЭС, условиями их функционирования, достижениями в развитии средств вычислительной и микропроцессорной техники, технологий программирования и ведения информации.

Для обеспечения необходимой эффективности систем электроснабжения может потребоваться формирование информационно-управляющих систем (ИУС) в рамках локальных электрообъектов, которые учитывают особенности некоторых задач и обеспечивают решение разноплановых сложностей, принципы, методы и модели, составляющие инструментарий программных средств на уровне автоматизированной системы дистанционного обеспечения (АСДУ) ЭЭС.

Тем не менее, исходя из особенностей технологических задач, решаемых на уровне АСДУ региональными ЭЭС и в меньшей степени АСДУ регионэнерго, сохраняются концепции позадачного подхода в технологиях программирования и ведения информации; осуществляется наращивание

в режимной модели топологии коммутационных схем; выполнение программного обеспечения (ПО) как надстроек одних программных средств над вторыми. При этом модели объектов не являются адаптивными и поэтому теряют свою адекватность при изменении актуального состояния оригинала и в большинстве случаев не обеспечивают автоматическое формирование алгоритма управления для перевода объекта из актуального состояния в состояние, соответствующее цели управления.

Проблемам управления информационными данными при проектировании систем электроснабжения посвящены работы О. Ю. Борисова [1], Т. Волковой [2], В. С. Дячук [3], А. Колмогорова [4], Р. Э. Косолапушкина [5], Г. Л. Липкина [6] и др. Тем не менее достаточное количество теоретических и научно-прикладных вопросов развития информационных систем для локальных электроэнергетических объектов остаются еще мало изученными и недостаточно решенными.

Целью статьи является формирование подходов к развитию информационных систем для локальных электроэнергетических объектов. Для достижения указанной цели необходимо с помощью подмножеств, определяющих соответственно надежность, качество, экономичность выработки, передачу, распределение и потребление энергии сформулировать условия для различных режимов, обосновать ряд входных положений, правил и принципов, составляющих основы теории построения ИУС, разработать метод автоматического создания графа.

РЕЗУЛЬТАТЫ

Практика показала, что уровень реализации интеллектуальных методов в энергетике не позволяет признать его удовлетворительным, ибо имеет место разрозненное использование интеллектуальных способов (экспертных систем, нейронных сетей) для решения отдельных задач контроля, управления, оценки состояния различных объектов ЭЭС. При этом современное состояние систем автоматизированного управления электроэнергетическими объектами ЭЭС Российской Федерации характеризуется неоднозначно: в рамках одной стороны создаются локальные вычислительные сети (ЛВС) на электрических станциях, на крупных подстанциях, в рамках другой стороны могут использоваться устаревшие информационные технологии и программы.

Отсутствие единой информационной основы и интегрированного использования современной технологии формирования информации в рамках решения технологических задач, которые по функциональному признаку возможно разбить на группы: имитационное моделирование, поддержка управленческих решений, диагностирование состояния элементов и объектов, анализ функционирования релейной защиты (РЗ) и противоаварийной автоматики (ПА), не позволяет устранить отмеченный недостаток автоматизированного управления.

В то же время следует отметить, что информатизация является самым перспективным и как можно менее затратным приемом повышения эффективности систем управления ЭЭС. Отмеченное требует нового уровня эффективности от ИУС ЭЭС. Из-за отсутствия инструментария построения ИУС локальных объектов, определяемого особенностями современных технологических задач, наблюдается механический перенос инструментария верхнего уровня управления на уровень управления в рамках локальных объектов.

При этом важным является разработка методологии ИУС в рамках информационных технологий, которые адекватны задачам автоматизированного управления для местных объектов ЭЭС. С решением различных задач по управлению ЭЭС связаны многочисленные исследования и разработки [7–9]. Тем не менее комплексный характер проблемы и особенности существующего состояния автоматизированных систем в энергетике Российской Федерации вызывают необходимость методологического объединения и дальнейшего развития методов решения данной проблемы.

Исследования свидетельствуют, что при имеющемся подходе в имитационном моделировании в режимной модели путем наращивания математической модели учитывается топология коммутационных схем, правил выполнения переключений, информационных свойств элементов. Этот подход является причиной построения моделей, которые сложно адаптировать под актуальное состояние объекта, и согласно цели управления создает проблемы по повышению эффективности математического моделирования.

В то же время средства управления, которые установлены на локальных объектах ЭЭС, позволяют эффективно решать технологические задачи в рамках текущих информационных технологий [10–12]. Поэтому технология управления локальными элементами ЭЭС предусматривает оценку фактической ситуации, выполнение имитационного моделирования с последующим анализом экономичности и надежности режимов, оценки принимаемых решений, получения информационной поддержки, то есть интегрированного решения задач различных групп.

Применение систем, реализующих принципы искусственного интеллекта, характеризуется направленностью на отдельные технологические задачи, то есть для одного объекта создаются индивидуальные программы контроля отдельных режимов, узлов и, соответственно, несколько баз данных, которые отличаются как по способам представления и хранения, так и по методикам обработки. При этом используемая технология программирования в большей степени соответствует

требованиям инженера-исследователя, чем инженера-технолога, поскольку не обеспечивает автоматического формирования составляющих моделей в рамках принципиальных схем конкретных электрических соединений, адаптации составляющих к актуальному состоянию локального объекта и целевой посылки и интеграции составляющих модели при решении задачи.

Разработка ИУС, обеспечивающих реализацию управления в рамках позиций инженера-технолога при эффективном решении разнообразных динамически складывающихся технологических задач, обуславливает необходимость применения системного подхода к автоматизации процессов по принятию управленческих решений. Для проектирования эффективной системы обоснования решений необходимо создать механизмы, которые обеспечивают как минимум целостное видение объектов.

Создание ИУС должно осуществляться в виде системы открытого типа, реализация которой при относительно незначительных затратах повышает эффективность автоматизированного управления объектами и элементами ЭЭС. Так, например, при построении ИУС, учитывающей особенности технологических задач, элементы ЭЭС представлены в виде обобщенного объекта с управляющими параметрами. При этом используется интегрированное описание объекта в графической (с множеством модификаций графического образа) и табличной формах. В то же время могут использоваться другие принципы и алгоритмы, которые могут быть полезными для совершенствования АСДУ ЭЭС.

Технология моделирования показывает, что в рамках инженера-электрика могут решать задачи управления при формировании составляющих моделей, при формировании определенных признаков информации, оперативно вводимой персоналом. При этом концепция отдельной адаптации составляющих модели локального объекта и дальнейшей интеграции составляющих для создания модели данных является адекватной к текущей ситуации. Поэтому необходимо учитывать методологию ведения информации, которая отличается от известных технологий методами для интегрированного ведения графической и символьной информации и создания интегрированной информационной модели предметной области с целью функционирования прикладных программ в рамках определенной информационной основы и упрощения взаимодействия пользователя с электронными машинами.

С помощью подмножеств, определяющих, соответственно, надежность, качество, экономичность выработки, передачу, распределение и потребление энергии, можно сформулировать условия для режимов: нормального, когда выполняются все условия; аварийного, когда нарушается условие надежности; послеаварийного, когда восстанавливается только условие надежности.

На основании анализа требований к современным системам управления, а также на основе анализа деятельности персонала можно сформулировать ряд входных положений, правил и принципов, составляющих основы теории построения ИУС. Так, например, метод адаптации модели основан на использовании положения об обобщенном объекте с параметрами, в которые входят и те, которые определяются связями с другими объектами и возмущениями, и которые влияют на составляющие модели объекта в отличие от объектно-ориентированного программирования, при котором они разделены на разнообразные группы по функциональному назначению.

Реализация концепции отдельного ведения составляющих модели с их последующей интеграцией основана на положениях и правилах взаимосвязи графических образов и реляционного описания их свойств. Использование положений о множестве состояний элементов и структур моделей, с которыми работает инженер-технолог стало основой для создания методов адаптации составляющих модели к актуальной цели управления. Положение о составляющих математической модели, методах их хранения и способах выполнения может позволить избежать необходимости замены программного кода при коррекции моделей устройств, директивных указаний по эксплуатации объектов ЭЭС.

В то же время использование локальных вычислительных узлов (рис. 1) не только позволяет распределить процессы и тем самым повысить быстродействие, но и повышает надежность работы локальных вычислительных сетей (ЛВС).

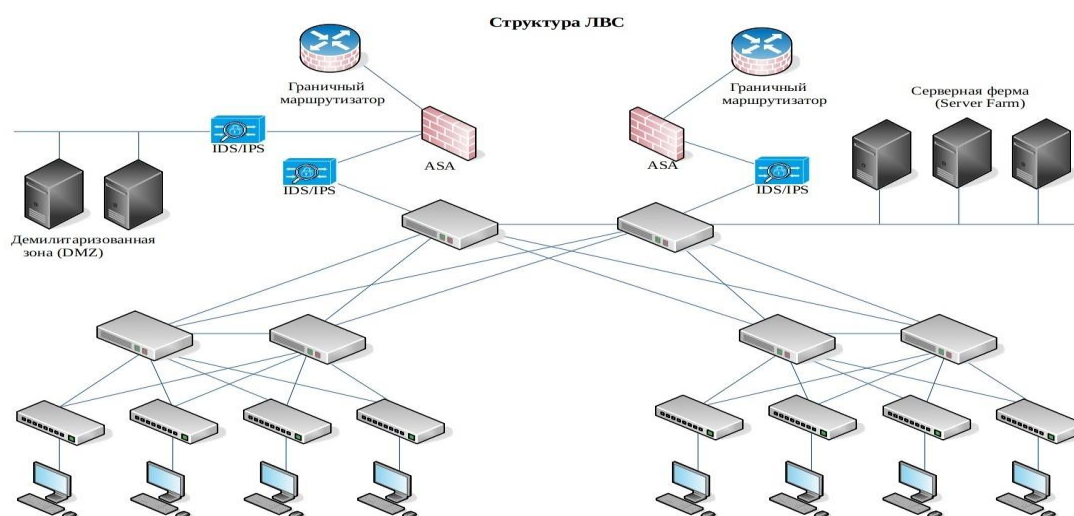


Рисунок 1 – ЛВС с функциональным распределением узлов

Построение программного обеспечения в виде системы, состоящей из информационного обеспечения, координирующего уровень обеспечивает открытость системы и соответствует международному стандарту построения открытых систем. Координирующий уровень (графический интерфейс пользователя) обеспечивает связи информационной модели и является основой для работы всего программного обеспечения.

Координирующий уровень представляет собой средство реализации адаптируемых моделей и диалогового взаимодействия с пользователем. В этом случае прикладное программное обеспечение с управляемой степенью интеграции технологических задач включает в себя: имитационное моделирование, информационную поддержку решений оперативного персонала, оценку режимов и состояния оборудования, анализ причин ситуаций при использовании систем в рамках искусственного интеллекта.

Взаимодействие человека с системой в этом случае происходит на основе запросов, включающих вид оборудования, условия эксплуатации, состояние изменения параметров. Анализ ситуаций и организация системы запросов приводит к тому, что все реализуемые модулем отчеты доступны пользователю через механизм запросов к базе данных. Имеющийся набор запросов обеспечивает текущие потребности как технических служб, так и аппарата управления и реализуется в виде списка меню, что максимально упрощает взаимодействие пользователей с программным обеспечением.

Отражая реальные связи между компонентами электрических сетей, модули программного комплекса взаимодействуют между собой, реализовывая механизмы для установления: связи между объектами на графических изображениях и символьной информацией, относящейся к этим объектам, а также между множеством модификаций графического образа элемента с технологической задачей; связей между модулями с использованием общих объектов учета, поименованных в различных модулях; данных из нескольких модулей при решении одной задачи.

Графический интерфейс пользователя в этом случае может обеспечивать работу инженера с терминологией его предметной области. Он отличается тем, что обеспечивает: оценку допустимости команд, ведение баз данных с помощью универсального строителя, создание и использование многоуровневых изображений с помощью графического редактора; формирование режимов работы электроэнергетического объекта с помощью редактора режимов; автоматическое и при необходимости в режиме интерактивного диалога построение моделей данных задач на основании текущего режима электроэнергетического объекта; оценку состояния оборудования с помощью интерпретатора состояния.

В то же время метод внепроцедурной технологии программирования включает методологию построения ИУС и набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать вид модели на всех этапах работы ИУС и обеспечивать информационную поддержку путем агрегации информации (рис. 2).

В этом случае технология агрегации информации основана на реализации различных исходных форм информации, динамически адаптируемых с помощью команд, кнопок, выпадающих меню, переключателей. Графическое изображение определяется также решаемой прикладной задачей. Поскольку в процессе принятия решения необходимо выполнение задач, относящихся к различным функциональным направлениям, то необходимо обеспечение связи между соответствующими графическими изображениями. Поэтому графический редактор должен работать с библиотеками графических образов, которые разбиты на характерные: соединительные элементы и наборы блоков.



Рисунок 2 – Структурная схема управления информационной моделью в электроэнергетике

Необходимо использовать ввод для графических образов ряда параметров (состояния, связи, математической модели элемента и др.), которые позволяют повысить эффективность адаптации составляющих модели, поскольку определяются либо сигналами, переданными дистанционно, либо введенными вручную, либо полученными расчетным путем. Поэтому необходимо разработать методы, обеспечивающие целостность информации, модификацию блоков и расчетных схем за счет сочетания атрибутов блока с портами, повышение надежности формирования модели локального объекта на основе графического изображения для блоков с тремя портами и более, оперативную перестройку электрической схемы под решаемую технологическую задачу.

В процессе исследования нами разработан метод автоматического создания графа, который базируется на следующих операциях с графами: объединении; пересечении; разности; приросте и замыкании одного графа в другом; сжатии графа. Каждая вершина графа соответствует одному элементу схемы и имеет тип графического образа: точка, соединительная линия, шина, блок. Первые три типа могут использоваться для облегчения отображения схемы пользователем. Оставшийся тип-блок, в свою очередь, может иметь еще ряд параметров, находящихся в его имени. Достаточно указать, что это информация о состоянии элемента: исправный, неисправный, включенный, отключенный, в ремонте.

ЗАКЛЮЧЕНИЕ

Подводя итоги, можно отметить, что анализ тенденций построения автоматизированных систем управления свидетельствует о децентрализации управления, которая приводит к появлению проблем, которые необходимо решать для обеспечения надежности электроснабжения. Отмечаются такие факторы, как нарушение связей системы управления, охватывавшей ранее единую электроэнергетическую сеть, значительный рост технологических задач, динамичность анализируемых ситуаций при управлении локальными электроэнергетическими объектами.

При этом отсутствует единый концептуальный подход к проблеме информационного обеспечения задач управления, реализуется позадачный способ ведения информации. Объектно-ориентированная технология программирования, современные специализированные интегрированные пакеты в большей степени соответствуют требованиям инженера-исследователя, чем инженера-технолога. Поэтому необходимо создать механизмы, обеспечивающие целостное видение альтернативных вариантов в рамках развития электроэнергетических сетей.

Список источников

1. Борисов, О. Ю. Состояние тенденции развития электроэнергетических систем России в региональной дифференциации / О. Ю. Борисов, О. Д. Старченкова, Д. С. Краснова // Естественно-гуманитарные исследования. – 2023. – № 4 (48). – С. 87–96.
2. Волкова, Т. Перспектива обмена данными информационных моделей в электроэнергетике между субъектами электроэнергетики в ПО «Россети Тюмень» в соответствии со стандартом CIM / Т. Волкова // Электроэнергия. Передача и распределение. – 2023. – № S3 (30). – С. 10–12.
3. Дячук, В. С. Комплексный подход к моделированию системы защиты данных в электроэнергетике / В. С. Дячук // Ученые записки Крымского федерального университета имени В.И. Вернадского. Экономика и управление. – 2016. – Т. 2 (68), № 3. – С. 49–57.

4. Колмогоров, А. Практические вопросы использования информационных моделей электрических сетей в деловых процессах электросетевой компании / А. Колмогоров // *Электроэнергия. Передача и распределение*. – 2023. – № S3 (30). – С. 2–9.
5. Косолапушкин, Р. Э. Проблемы BIM-моделирования в области электроэнергетики в России / Р. Э. Косолапушкин, И. И. Добросотских // *Интернаука*. – 2023. – № 2–1 (272). – С. 17–19.
6. Липкин Г. Л. Типовые ошибки проектирования и наладки систем управления в энергетике / Г. Л. Липкин, Г. В. Азаренков // *Управление качеством*. – 2023. – № 10. – С. 40–46.
7. Унижаев, Н. В. Особенности внедрения нейросетей и систем искусственного интеллекта на предприятиях электроэнергетики / Н. В. Унижаев // *Вопросы инновационной экономики*. – 2023. – Т. 13, № 1. – С. 215–232.
8. Чугин, А. В. Внедрение информационной системы ведения оперативной и технологической документации в электронном виде на Юго-Западной ТЭЦ / А. В. Чугин, К. В. Бабкин, С. А. Полоротов // *Электрические станции*. – 2023. – № 1 (1098). – С. 50–53.
9. Шеметов, А. С. Программные комплексы сопровождения жизненного цикла РЗА и АСУ ТП / А. С. Шеметов, О. В. Кириенко, А. А. Горчаков // *Электроэнергия. Передача и распределение*. – 2023. – № 2 (77). – С. 72–78.
10. Юхтанов, Р. А. Управление электроэнергетикой в процессе цифрового преобразования экономики России / Р. А. Юхтанов, М. А. Смирнова, В. П. Максимов // *Наукофера*. – 2023. – № 4–2. – С. 321–325.
11. Boychenko, O. V. Principles of integrated system of information security in the management on electric power facilities / O. V. Boychenko, V. S. Dyachuk // *East European Scientific Journal*. – 2016. – № 9 (3). – P. 26–29.
12. Palchevsky, E. V. Artificial neural network for forecasting electricity consumption in energy enterprises / E. V. Palchevsky, V. Antonov., L. A. Kromina, L. E. Rodionova and A. R. Fakhruullina // *Programmynaya Ingeneria*. – 2023. – № 14 (1). – P. 34–41.

References

1. Borisov, O. Yu., Starchenkova, O. D., Krasnova, D. S. State of the development trend of electric power systems in Russia in regional differentiation. *Natural-Humanitarian Studies*, 2023, no. 4 (48), pp. 87–96.
2. Volkova, T. Prospects for data exchange of information models in the electric power industry between subjects of the electric power industry in the Rosseti Tyumen software in accordance with the CIM standard. *Electricity. Transmission and distribution*, 2023, no. S3 (30), pp. 10–12.
3. Dyachuk, V. S. An integrated approach to modeling a data protection system in the electric power industry. *Scientific notes of the Crimean Federal University named after V.I. Vernadsky. Economy and Management*, 2016, vol. 2 (68), no. 3, pp. 49–57.
4. Kolmogorov, A. Practical issues of using information models of electrical networks in business processes of an electric grid company. *Electricity. Transmission and Distribution*, 2023, no. S3 (30), pp. 2–9.
5. Kosolapushkin, R. E., Dobrosotkikh, I. I. Problems of BIM modeling in the field of electric power industry in Russia. *Internauka*, 2023, no. 2–1 (272), pp. 17–19.
6. Lipkin, G. L., Azarenkov, G. V. Typical errors in the design and adjustment of control systems in the energy sector. *Quality Management*, 2023, no. 10, pp. 40–46.
7. Chugin, A. V. Babkin, K. V., Polorotov, S. A. Introduction of an information system for maintaining operational and technological documentation in electronic form at the South-West CHPP. *Electric stations*, 2023, no. 1 (1098), pp. 50–53.
8. Shemetov, A. S., Kirienko, O. V., Gorchakov, A. A. Software systems for supporting the life cycle of relay protection and automation systems and automated process control systems. *Electricity. Transmission and distribution*, 2023, no. 2 (77), pp. 72–78.
9. Yukhtanov, R. A., Smirnova, M. A., Maksimov, V. P. Electric power industry management in the process of digital transformation of the Russian economy. *Scienceosphere*, 2023, no. 4–2, pp. 321–325.
10. Boychenko, O. V., Dyachuk, V. S. Principles of integrated system of information security in the management on electric power facilities. *East European Scientific Journal*, 2016, no. 9 (3), pp. 26–29.
11. Palchevsky, E. V., Antonov, V. V., Kromina, L. A., Rodionova, L. E., Fakhruullina, A. R. Artificial neural network for forecasting electricity consumption in energy enterprises. *Programmynaya Ingeneria*, 2023, no. 4 (1), pp. 34–41.

Статья поступила в редакцию 14.03.2024; одобрена после рецензирования 02.05.2024; принята к публикации 05.05.2024.

The article was submitted 14.03.2024; approved after reviewing 02.05.2024; accepted for publication 05.05.2024.

УДК 004.89

ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА УПРАВЛЕНИЯ ПОЛИВОМ СЕЛЬСКОХОЗЯЙСТВЕННЫХ УЧАСТКОВ

Старов Дмитрий Владимирович, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
старший преподаватель кафедры технологии материалов и промышленной инженерии,
ORCID: 0009-0002-0377-3044, e-mail: bortv715ke@mail.ru

Корякова Виктория Андреевна, Астраханский государственный университет им. В. Н. Татищева, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,
ассистент кафедры информационной безопасности, ORCID: 0000-0002-2472-9248, e-mail:
koryakova-01@mail.ru

Цель работы – повышение эффективности контроля и управления поливом сельскохозяйственных участков путем разработки интеллектуальной системы для управления системой полива. В данной статье предлагается разработать интеллектуальную систему для управления системой полива сельскохозяйственных участков. Предложенный подход заключается в планировании орошения с использованием метеорологических данных и показателей с датчиков в реальном времени и предоставлении графика орошения сельскохозяйственных культур на разных фазах фенологического развития на основе характеристик почвы, ирригационной системы и погодных условий с использованием данных прогнозирования нейронной сети. Разработанная интеллектуальная система позволит определить функциональность и конфигурацию системы полива с учетом специфических особенностей участка. Разработанное программное обеспечение может быть интегрировано с другими системами умного дома или сторонними сервисами. Проведен анализ существующих решений. Описана модель интеллектуальной системы полива сельскохозяйственных участков. Спроектированы основные сценарии предложенного решения.

Ключевые слова: умное земледелие, контроль микроклимата почвы, мобильные устройства, мобильные приложения, сельское хозяйство

Финансирование: исследование выполнено при поддержке Программы развития Астраханского государственного университета (Приоритет-2030)».

INTELLIGENT WATERING SYSTEM FOR AGRICULTURAL AREAS

Starov Dmitry V., Astrakhan Tatishchev State University, 20a Tatishchev St., 414056, Astrakhan, Russian Federation,

Senior Lecturer of the Department of Materials Technology and Industrial Engineering, ORCID: 0009-0002-0377-3044, e-mail: bortv715ke@mail.ru

Koryakova Victoria A., Astrakhan Tatishchev State University, 20a Tatishchev St., 414056, Astrakhan, Russian Federation,

Assistant of the Department of Information Security, ORCID: 0000-0002-2472-9248, e-mail: koryakova-01@mail.ru

The purpose of the work is to increase the efficiency of control and management of irrigation of agricultural plots by developing an intelligent system for controlling the irrigation system. This article proposes to develop an intelligent system for controlling the irrigation system of agricultural plots. The proposed approach is to schedule irrigation using meteorological data and real-time sensor indicators and provide irrigation schedule for crops at different phases of phenological development based on soil characteristics, irrigation system and weather conditions using neural network forecasting data. The developed intelligent system will allow us to determine the functionality and configuration of the irrigation system, taking into account the specific characteristics of the site. The developed software can be integrated with other smart home systems or third-party services. An analysis of existing solutions was carried out. A model of an intelligent irrigation system for agricultural plots is described. The main scenarios of the proposed solution have been designed.

Keywords: smart farming, soil microclimate control, mobile devices, mobile applications, agriculture

Financial support: this research was supported by the Astrakhan State University Development Program (Priority-2030)».

Graphical annotation (Графическая аннотация)

**ВВЕДЕНИЕ**

Более 70 % сельскохозяйственных угодий в Российской Федерации находятся в малоувлажненных и засушливых районах, где высокие урожаи возможны лишь при использовании методов орошения [1]. В последние годы наблюдается быстрый рост внедрения цифровых технологий в сфере орошения. Использование спутниковых данных, сенсорных сетей, анализа информации и беспилотных летательных аппаратов значительно повышает эффективность процесса орошения, что, в свою очередь, приводит к увеличению урожайности. Технологические компании усиливают свое присутствие в сельском хозяйстве, предлагая фермерам передовые технологии и услуги обработки данных, направленные на максимизацию прибыли. Инновационными стратегиями развития являются «точное земледелие», внедрение роботов и автоматизация производственных процессов, а также использование альтернативных источников энергии в аграрном секторе. Точное земледелие является важным аспектом умного земледелия и может быть определено как стратегия управления, которая собирает, обрабатывает и анализирует временные, пространственные и индивидуальные данные и объединяет их с другой информацией для поддержки управленческих решений в соответствии с предполагаемой изменчивостью для улучшения использования ресурсов.

Актуальными задачами в цифровизации сельского хозяйства являются создание новых принципов, методов и алгоритмов для перехода к «умному» сельскому хозяйству, основанному на использовании автоматизированных систем принятия решений, автоматизации и роботизации производства, а также разработка систем управления, обеспечивающих дистанционный контроль работы, сбор, обработку и анализ данных о ходе выполнения технологических процессов. С помощью разработки систем с использованием умных технологий сельскохозяйственные отрасли смогут повысить урожайность, производить более качественные товары, свести к минимуму человеческий труд и в то же время быть хорошо подготовленными к преодолению кризисов, возникающих в результате стихийных бедствий.

В данной статье предлагается разработать интеллектуальную систему для управления системой полива сельскохозяйственных участков. Предложенный подход заключается в планировании орошения с использованием метеорологических данных и показателей с датчиков в реальном времени и предоставлении графика орошения сельскохозяйственных культур на разных фазах фенологического развития на основе характеристик почвы, ирригационной системы и погодных условий с использованием данных прогнозирования нейронной сети. Разработанная интеллектуальная система позволит определить функциональность и конфигурацию системы полива с учетом специфических особенностей участка.

АНАЛИЗ ИССЛЕДОВАНИЙ В ОБЛАСТИ УПРАВЛЕНИЯ СИСТЕМОЙ ПОЛИВА

Исследования, проведенные в последние годы, показали, что использование инновационных технологий, управленческих подходов и инструментов моделирования позволяет улучшить планирование орошения, сэкономить воду, повысить доход фермера и снизить нагрузку на окружающую среду [2–8].

Авторы в работе [9] разработали интеллектуальную систему орошения для сельского хозяйства на основе беспроводных датчиков. Каждый узел беспроводных датчиков состоит из датчика влажности почвы, датчика температуры, микроконтроллера для сбора данных и радиопередатчика. Блок приемника состоит из задающего микроконтроллера, радиопередатчика, GSM и насоса. Связь между передатчиком и приемником осуществляется по протоколу ZigBee. График полива можно контролировать с помощью приложения для Android.

Исследователи в работе [10] создали интеллектуальную ирригационную систему на солнечной энергии. Система состоит из водяного насоса на солнечной энергии и автоматического контроля расхода воды с помощью датчика влажности. Солнечная энергия с солнечных панелей используется для автоматической перекачки воды из скважины непосредственно в резервуар для хранения на уровне земли в зависимости от интенсивности солнечного света. Это позволяет сэкономить значительное количество энергии и эффективно использовать возобновляемые источники энергии. Клапан управляется с помощью интеллектуального алгоритма, который регулирует поток воды на поле в зависимости от потребности почвы во влажности.

Автор в работе [11] разработал автоматизированную систему орошения на базе Android с использованием Raspberry Pi, которая эффективно контролирует работу системы капельного орошения. Сбор данных осуществляется с помощью беспроводных станций сбора данных на солнечных батареях с целью управления клапанами для орошения. Разработанная система состоит из трех блоков: блока базовой станции, блока клапанов и блока датчиков. Автоматизированная ирригационная система является осуществимой и экономически эффективной для оптимизации водных ресурсов для сельскохозяйственного производства.

Авторы исследования [12] разработали веб-систему автоматического полива, использующую беспроводную сенсорную сеть и встроенную плату Linux. Целью их исследования было дистанционное управление устройствами и мониторинг системы через веб-страницу. Raspberry Pi используется в качестве встроенной платы Linux, разработанной на основе архитектуры микроконтроллера ARM 11. Встроенная плата Linux обеспечивает связь со всеми распределенными сенсорными узлами, расположенными в ферме, через протокол ZigBee и сама выступает в качестве координированного узла в беспроводной сенсорной сети. Цель узла-координатора – собирать по беспроводной сети такие параметры, как влажность и температура почвы. Каждый сенсорный узел состоит из датчика влажности и температуры почвы и одного антенного устройства ZigBee RF для связи с узлом-координатором. Raspberry Pi сохраняет собранные данные в базе данных и анализирует сохраненные данные. Система будет работать по алгоритму, разработанному для полива урожая. Плата имеет интерфейс Ethernet и запускает простой веб-сервер данных. Следовательно, координатор собирает данные по протоколу беспроводной связи ZigBee и позволяет пользователю отслеживать данные через веб-браузер. Пользователь может удаленно включать и выключать ирригационную систему.

Автор в работе [13] предложил интеллектуальную систему орошения на основе Arduino с использованием интернета вещей. Разработка выполнялась с использованием контроллера Arduino без использования датчиков влажности почвы. Система позволяет осуществлять полив более эффективно, основываясь на значениях влажности почвы, температуры и водородного показателя, передаваемых на микроконтроллер (Arduino UNO), который управляет электромагнитным клапаном через транзистор схемы управления (BC547). На основе измерений датчика при заранее определенных пороговых значениях осуществляется автоматический полив. В зависимости от значений датчика значения обновляются в устройствах интернета вещей, а также на мобильном устройстве Android.

МОДЕЛЬ ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПОЛИВОМ СЕЛЬСКОХОЗЯЙСТВЕННЫХ УЧАСТКОВ

Интеллектуальная система управления поливом использует данные о погоде, почве, урожае и ирригационной системе для оценки компонентов водного баланса в течение цикла выращивания сельскохозяйственных культур и для разработки графика орошения.

Датчики отвечают за измерение и мониторинг всех факторов в интеллектуальной системе, например, для мониторинга состояния почвы могут использоваться специальные датчики, направленные на измерение содержания питательных веществ, фосфатов, влажность почвы, ее уплотнение [14].

Общая схема интеллектуальной системы управления поливом показана на рисунке 1.

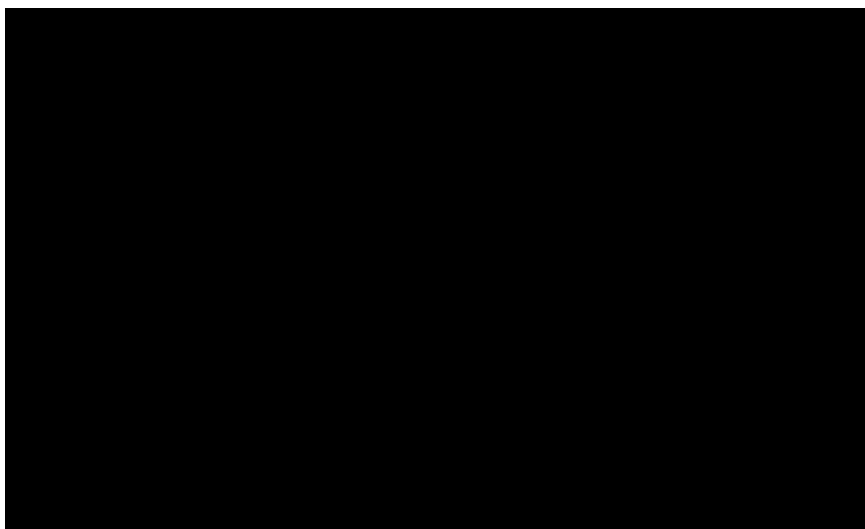


Рисунок 1 – Общая схема интеллектуальной системы управления поливом

Система управления координируется путем получения сигналов, которые представляют собой обобщенные оценки качества работы всей системы управления объектами с точки зрения внешнего воздействия. Данное воздействие может быть другими системами управления на текущем уровне или на более высоких уровнях иерархии. Поэтому важно изменять работу системы управления не только на основе данных о объектах и их окружении, но также учитывать качественные характеристики для согласования целей функционирования различных систем управления и объектов управления, которые могут быть взаимосвязанными и взаимодействующими.

Усовершенствование предложенной системы управления включает в себя добавление дополнительного контура принятия решений на основе прогнозирования нейронных сетей.

Модуль сбора данных в реальном времени для составления графика полива сельскохозяйственных культур на основе нейросети включает в себя следующие этапы:

1. Сбор данных о погодных условиях (температура, влажность, осадки и т. д.) с помощью датчиков и метеостанций на сельскохозяйственных участках.
2. Сбор данных о состоянии почвы (влажность почвы, уровень pH и питательность) с помощью соответствующих датчиков.
3. Обработка полученных данных с помощью нейронных сетей для прогнозирования оптимального графика полива для каждого участка поля.
4. Управление системой полива на основе прогноза с учетом индивидуальных потребностей культур и текущих погодных условий.

Такая система позволит оптимизировать расход воды, улучшить качество урожая и снизить затраты на производство сельскохозяйственных культур.

ПРОЕКТИРОВАНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ПОЛИВОМ

Были спроектированы диаграммы, описывающие основные сценарии: управление поливом, работа программно-аппаратной системы.

Просмотр и настройка параметров полива, а именно изменение частоты измерения датчика, и параметры порогов максимальной и минимальной влажности показаны с помощью диаграммы деятельности на рисунке 2.

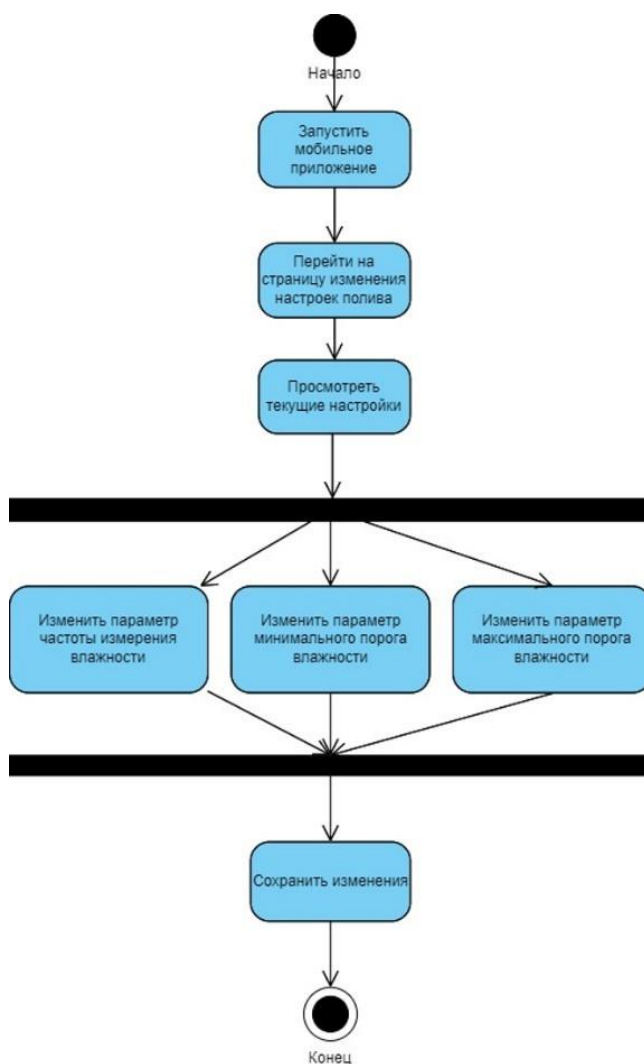


Рисунок 2 – Диаграмма деятельности: настройка полива

Требования к данным о почве ограничиваются знанием объемного содержания влаги в почве при полевой вместимости, точки увядания и насыщения, электропроводности, органического вещества и фактической глубины почвы. Пользователь может выбрать характеристики почвы из базы данных почв, вставить свои собственные данные или использовать специальную функцию переноса для создания и сохранения новых данных в соответствии с анализом почвы для конкретного участка.

Кроме того, система может быть оснащена емкостными датчиками для мониторинга содержания влаги в почве в режиме реального времени и проверки/обновления результатов моделирования водного баланса почвы.

База данных культур по умолчанию организована с использованием ориентировочных параметров роста и развития культур (например, коэффициент культуры, продолжительность циклов выращивания, минимальная и максимальная глубина корней, оптимальный порог урожайности).

Более подробное описание и обновление параметров развития и роста культур возможно с помощью наблюдений пользователя в поле в течение вегетационного периода.

На рисунке 3 показан алгоритм работы разрабатываемой системы, которая может управляться как физически, находясь в непосредственной близости к микроконтроллеру, так и автоматически, подгружая параметры из памяти контроллера, либо с удаленного источника, используя встроенный Wi-Fi-модуль.

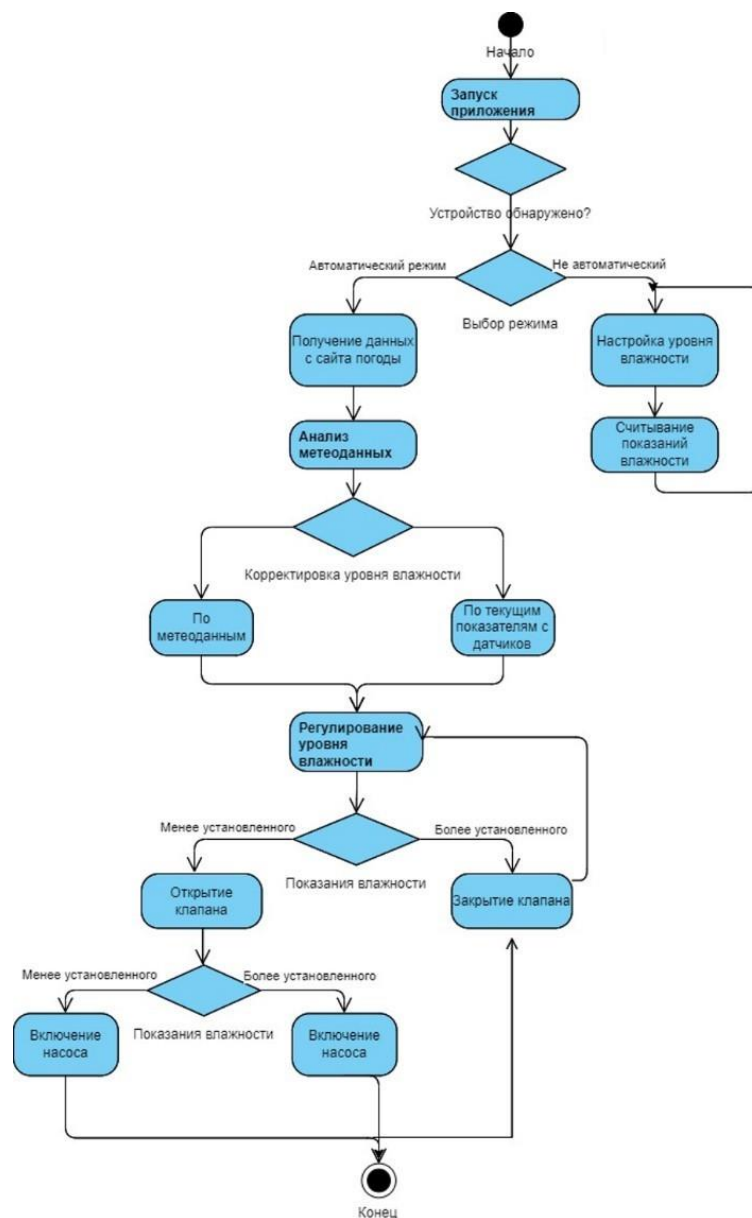


Рисунок 3 – Алгоритм работы программно-аппаратной системы

Алгоритм работы программно-аппаратной системы начинается с запуска приложения на смартфоне оператора системы. Далее блок выбора – физический, в приложении появляется список возможных подключений и если устройство не обнаружено, то первостепенное действие заключается в проверке питания, если оно отсутствует после проведения измерений, в таком случае блок управления нуждается в ремонтно-восстановительных работах. При условии наличия питания и отсутствия имени устройства в списке обнаружения выполняется перезагрузка. Если подключение осуществлено, следующим шагом будет выбор режима работы системы.

Режим работы системы делится на два типа:

1. Физический циклический субалгоритм, выполняемый оператором системы, состоящий из последовательности повторяющихся действий: настройки уровня влажности в процентном соотношении и считывание показаний влажности. Такой метод может применяться во время межсезонья, когда почва нуждается в отдыхе и ее полив производится намного реже.

2. Автоматический. Устроен таким образом, что программа по Wi-fi-соединению подключается к сети Интернет и получает данные с гидрометеостанции (при условии наличия сети), далее происходит анализ между полученными данными, данными внутренних отчетов и показаниями, установленными оператором при текущем подключении. В случае несовпадения контрольных цифр предполагаемой влажности почвы в результате анализа источника и установленного предела программа может предложить изменить показания уровня влажности.

Блок выбора «Корректировка уровня влажности» предоставляет оператору выбор: установить свои данные или использовать данные условного источника. В любом из вариантов выбора программа перейдет к шагу проверки «Показаний влажности». На данном этапе производится опрос датчиков влажности на предмет соответствия заявленной влажности почвы в грядках к фактической. В случае действительных показаний менее установленного значения происходит открытие электромагнитного клапана трубы, идущей к соответствующей грядке до тех пор, пока следующий замер влажности не покажет значение больше установленного и наоборот. Этот этап образует цикл, выход из которого приводит к завершению работы системы.

Алгоритм работы системы включает следующие шаги: сбор данных – система собирает данные о почве, погоде, влажности, уровне урожайности и других параметрах, используя сенсоры, дроны или другие устройства; анализ данных – полученные данные обрабатываются и анализируются с помощью алгоритмов машинного обучения, чтобы определить оптимальные условия для роста растений; принятие решений – на основе результатов анализа система принимает решения о необходимых действиях, таких как полив, удобрение, обработка почвы и защита от вредителей; выполнение действий – система автоматически управляет технологическими процессами, такими как полив и удобрение, согласно принятым решениям; мониторинг и адаптация: система постоянно осуществляет мониторинг результатов своих действий и может корректировать стратегию в реальном времени, чтобы обеспечить оптимальные условия для роста растений.

Входные данные для интеллектуальной системы управления поливом включают:

Параметры полива. Пользователь может задавать режим работы системы (ручной, по сенсору, по расписанию), а также длительность полива.

Данные о влажности почвы. Система может получать информацию о влажности почвы от датчиков.

Данные о погоде. Система может использовать сразу несколько источников данных о погодных условиях в конкретное время. Это позволяет заранее спланировать, а затем проконтролировать процесс автоматического полива.

Данные об осадках. Система может измерять и фиксировать количество воды, которое было сэкономлено в ходе эксплуатации системы.

Список изменений, которые вносились в цикл полива. Это позволяет отслеживать все процессы без исключения.

Географические данные участка: координаты местоположения садового участка или его размеры для определения области покрытия системой полива.

Информация о растениях: виды растений, их потребности в воде, интенсивность полива и временные интервалы для полива каждого вида растения.

Данные о погоде: информация о прогнозе погоды, уровне осадков, температуре и влажности воздуха для оптимизации расписания полива.

Расписание полива: требуемые параметры для составления расписания полива, такие как дни недели, время суток, продолжительность и интенсивность полива.

Состояние системы полива: данные о работоспособности системы полива, датчики уровня влажности почвы, статус клапанов и насосов для мониторинга и управления текущим состоянием системы.

Информация о водопотреблении: данные о расходе воды, использовании источников воды (например, скважина, водопроводная система) для эффективного планирования и мониторинга водопотребления.

Выходные данные:

Отчёты об осадках. Приложение использует сразу несколько источников данных о погодных условиях в конкретное время. Это позволяет заранее спланировать, а затем проконтролировать процесс автоматического полива.

Измерение и фиксация количества воды, которое было сэкономлено в ходе эксплуатации системы.

Создание списка изменений, которые вносились в цикл полива. Это позволяет отслеживать все процессы без исключения.

Расписание полива: приложение позволяет установить дни и время полива для каждой зоны сада. Пользователь может настроить частоту и продолжительность полива в зависимости от потребностей растений.

Управление зонами полива: пользователь может выбрать конкретную зону сада, которую хочет полить, и установить параметры полива для этой зоны.

Мониторинг влажности почвы: приложение может отображать данные о влажности почвы в разных зонах сада. Пользователь может отслеживать уровень влажности и регулировать программу полива соответственно.

Уведомления и оповещения: приложение может отправлять уведомления о необходимости замены фильтров, наполнении резервуаров с водой, проблемах с системой полива и других важных событиях.

Отчеты и статистика: приложение может предоставлять отчеты о количестве потраченной воды, эффективности полива, сэкономленных ресурсах и других данных, полезных для анализа работы системы полива.

ЗАКЛЮЧЕНИЕ

Для совершенствования производства сельскохозяйственной продукции необходимо повышение технического уровня сельскохозяйственной техники и технологий. Особое внимание следует уделить улучшению технологий обработки почвы, внесения удобрений, ядохимикатов и систем орошения. Важным направлением является проведение научно-исследовательских работ с целью создания новых технологий, основанных на цифровых, интеллектуальных системах управления сельскохозяйственными участками. Эти системы обеспечивают точное внесение удобрений, опрыскивание, опыливание и полив, что делает подход к производству более эффективным. В статье был предложен подход для сбора, обработки метеорологических данных и показателей с датчиков в реальном времени для составления графика полива сельскохозяйственных культур и управления им. Спроектирована модель интеллектуальной системы управления поливом сельскохозяйственных участков. Описаны основные сценарии использования данной системы. Разработанная интеллектуальная система позволит повысить качество полива, путем изменения поливной нормы в соответствии с уровнем влагозапасов участков поля в режиме реального времени, адаптации режима полива к изменяющимся в течение поливного периода условиям.

Список источников

1. Ольгаренко, Г. В. Реализация программы импортозамещения в области производства техники полива в Российской Федерации / Г. В. Ольгаренко // Мелиорация и водное хозяйство. – 2018. – № 1. – С. 44–47.
2. Zotarelli, L. Irrigation scheduling for green bell peppers using capacitance soil moisture sensors / L. Zotarelli, M. D. Dukes, J. M. S. Scholberg, K. Femminella, R. Munoz-Carpena // J. Irrig. Drain. Eng. – 2011. – Vol. 137. – P. 73–81.
3. Rosa, R. D. Implementing the dual crop coefficient approach in interactive software. Background and computational strategy / R. D. Rosa, P.; Paredes, G. C. Rodrigues, I. Alves, R. M. Fernando, L. S. Pereira, R. G. Allen // Agric. Water Manag. – 2012. – Vol. 103. – P. 8–24.
4. Dobbs, N. A. Evaluating irrigation applied and nitrogen leached using different smart irrigation technologies on bahiagrass (*Paspalum notatum*) / N. A. Dobbs, K. W. Migliaccio, Y. C. Li, M. D. Dukes, K. T. Morgan // Irrig. Sci. – 2014. – Vol. 32. – P. 193–203.
5. Pereira, L. S. Modeling barley water use and evapotranspiration partitioning in two contrasting rainfall years. Assessing SIMDualKc and AquaCrop models / L. S. Pereira, P. Paredes, G. C. Rodrigues, M. Neves // Agric. Water Manag. – 2015. – Vol. 159. – P. 239–254.
6. Todorovic, M. Eco-efficiency of agricultural water systems: Methodological approach and assessment at meso-level scale / M. Todorovic, A. Mehmeti, A. Scardigno // J. Environ. Manag. – 2016. – Vol. 165. – P. 62–71.
7. Mehmeti, A. Assessing the eco-efficiency improvement of Sinistra Ofanto irrigation scheme / A. Mehmeti, M. Todorovic, A. Scardigno // J. Clean. Prod. – 2016. – Vol. 138. – P. 208–216.
8. Seidel, S. J. Field Evaluation of Irrigation Scheduling Strategies using a Mechanistic Crop Growth Model / S. J. Seidel, S. Werisch, K. Barfus, M. Wagner, N. Schutze, H. Laber // Irrig. Drain. – 2016. – Vol. 65. – P. 214–223.
9. Paredes, P. Assessing potato transpiration, yield and water productivity under various water regimes and planting dates using the FAO dual Kc approach / P. Paredes, D. D'Agostino, M. Assif, M. Todorovic, L. S. Pereira // Agric. Water Manag. – 2018. – Vol. 195. – P. 11–24.
10. Harishankar, S. Solar Powered Smart Irrigation System / S. Harishankar, R. S. Kumar, K. P. Sudharsan, U. Vignesh, and T. Viveknath // Advance in Electronic and Electric Engineering. – 2014. – Vol. 4. – P. 341–346.
11. Jadhav, S. Android Based Automated Irrigation System using Raspberry Pi / S. Jadhav, and H. Hambarde // International Journal of Science and Research. – 2016. – Vol. 5, issue 6.
12. Ata, S. R. Web Based Automatic Irrigation System Using Wireless Sensor Network and Embedded Linux Board / S. R. Ata // International Journal of Advancement in Engineering Technology Management & Applied Science. – 2016. – Vol. 3, issue 2.
13. Parameswaran, G. Arduino Based Smart Drip Irrigation System Using Internet of Things / G. Parameswaran and K. Sivaprasath // International Journal of Engineering Science and Computing. – 2016. – Vol. 6, issue 5.
14. Kumar, T. U., Periasamy, A., IoT Based Smart Farming (E-FARM)*S / T. U. Kumar, A. Periasamy // International Journal of Recent Advances in Multidisciplinary Topics. – 2021. – Vol. 2 (4), – P. 85–87.

References

1. Olgarenko, G. V. Implementation of the import substitution program in the production of irrigation equipment in the Russian Federation. *Melioration and water management*, 2018, no. 1, pp. 44–47 (In Russ.).
2. Zotarelli, L., Dukes, M. D., Scholberg, J. M. S., Femminella, K., Munoz-Carpena, R. Irrigation scheduling for green bell peppers using capacitance soil moisture sensors. *J. Irrig. Drain. Eng.*, 2011, vol. 137, pp. 73–81.
3. Rosa, R. D., Paredes, P., Rodrigues, G. C., Alves, I., Fernando, R. M., Pereira, L. S., Allen, R. G. Implementing the dual crop coefficient approach in interactive software. Background and computational strategy. *Agric. Water Manag.*, 2012, vol. 103, pp. 8–24.

4. Dobbs, N. A., Migliaccio, K. W., Li, Y. C., Dukes, M. D., Morgan, K. T. Evaluating irrigation applied and nitrogen leached using different smart irrigation technologies on bahiagrass (*Paspalum notatum*). *Irrig. Sci.*, 2014, vol. 32, pp. 193–203.
5. Pereira, L. S., Paredes, P., Rodrigues, G. C., Neves, M. Modeling barley water use and evapotranspiration partitioning in two contrasting rainfall years. Assessing SIMDualKc and AquaCrop models. *Agric. Water Manag.*, 2015, vol. 159, pp. 239–254.
6. Todorovic, M., Mehmeti, A., Scardigno, A. Eco-efficiency of agricultural water systems: Methodological approach and assessment at meso-level scale. *J. Environ. Manag.*, 2016, vol. 165, pp. 62–71.
7. Mehmeti, A., Todorovic, M., Scardigno, A. Assessing the eco-efficiency improvement of Sinistra Ofanto irrigation scheme. *J. Clean. Prod.*, 2016, vol. 138, pp. 208–216.
8. Seidel, S. J., Werisch, S., Barfus, K., Wagner, M., Schutze, N., Laber, H. Field Evaluation of Irrigation Scheduling Strategies using a Mechanistic Crop Growth Model. *Irrig. Drain.*, 2016, vol. 65, pp. 214–223.
9. Paredes, P., D'Agostino, D., Assif, M., Todorovic, M., Pereira, L. S. Assessing potato transpiration, yield and water productivity under various water regimes and planting dates using the FAO dual Kc approach. *Agric. Water Manag.*, 2018, vol. 195, pp. 11–24.
10. Harishankar, S., Kumar, R. S., Sudharsan, K. P., Vignesh, U. and Viveknath, T. Solar Powered Smart Irrigation System. *Advance in Electronic and Electric Engineering*, 2014, vol. 4, pp. 341–346.
11. Jadhav, S. and Hambarde, H. Android Based Automated Irrigation System using Raspberry Pi. *International Journal of Science and Research*, 2016, vol. 5, issue 6.
12. Ata, S. R. Web Based Automatic Irrigation System Using Wireless Sensor Network and Embedded Linux Board. *International Journal of Advancement in Engineering Technology Management & Applied Science*, 2016, vol. 3, issue 2.
13. Parameswaran, G. and Sivaprasath, K. Arduino Based Smart Drip Irrigation System Using Internet of Things. *International Journal of Engineering Science and Computing*, 2016, vol. 6, issue 5.
14. Kumar, T. U., Periasamy, A., IoT Based Smart Farming (E-FARM)'S. *International Journal of Recent Advances in Multidisciplinary Topics*, 2021. vol. 2 (4), pp. 85–87.

Статья поступила в редакцию 27.05.2024; одобрена после рецензирования 03.06.2024; принята к публикации 04.06.2024.

The article was submitted 27.05.2024; approved after reviewing 03.06.2024; accepted for publication 04.06.2024.

ПРАВИЛА ДЛЯ АВТОРОВ

1. В журнале публикуются материалы на английском и русском языках по тематике, соответствующей утвержденным для журнала отраслям наук, группам специальностей.

2. В список соавторов работ включаются только те лица, которые внесли творческий вклад в подготовку представленных материалов. Лицам, оказавшим только техническую помощь, можно выразить благодарность в конце статьи. Один человек может быть автором (соавтором) не более чем двух статей в одном номере журнала, причем единственным автором он может быть только в одной статье.

3. Объем публикаций для научных статей должен быть не менее 8 страниц, а количество источников в библиографическом списке (списке литературы) – не менее 10 позиций.

4. Содержание каждой статьи должно включать следующие элементы: УДК; название статьи; сведения об авторах, включая их место работы, должность, адрес электронной почты; аннотацию объемом от 100 до 250 слов, ключевые слова (от 9 до 13); графическую аннотацию, отражающую содержание статьи; название статьи, сведения об авторах, аннотацию и ключевые слова на английском языке (для англоязычных статей – на русском языке); введение – оно должно заканчиваться формулировкой цели работы в явной форме; собственно текст статьи – очень желательна его сегментация на разделы, имеющие содержательные заголовки; выводы или заключение (должны соответствовать формулировке цели статьи).

5. Для русскоязычных статей приводится два библиографических списка: на языке оригинала статьи; список с транслитерацией русскоязычных источников на латиницу и (дополнительно) приведением в квадратных скобках переводов названий статей и названий источников на английский язык.

В «русскоязычном» библиографическом списке (списке литературы) порядок следования источников – по алфавиту фамилий авторов (сначала русскоязычные источники, потом иноязычные). На все источники, включенные в библиографический список, должны быть даны ссылки в тексте статьи в квадратных скобках. При необходимости авторы могут указывать номера страниц в источниках, на которые даются ссылки. Приветствуются ссылки на иноязычные источники, а также на материалы, опубликованные ранее в журнале «Прикаспийский журнал: управление и высокие технологии». Однако в последнем случае количество таких ссылок не должно превышать 20 % от общего количества источников, включенных в библиографический список. Для источников, имеющих DOI, целесообразно его указывать. При ссылках на статьи, опубликованные в журнале «Прикаспийский журнал: управление и высокие технологии», целесообразно в конце библиографического описания источника в круглых скобках указывать гиперссылку, указывающую на место размещения статьи на страничке сайта Астраханского государственного университета.

Ссылки в библиографическом списке на материалы, размещенные в интернете, допускаются при соблюдении следующих условий: если у материала, на который дается ссылка, имеется автор и/или название, то они должны быть указаны для этого источника; должен быть приведен полный маршрут доступа к источнику в интернете; должна быть указана дата обращения (доступа) к источнику.

Ограничения по списку литературы: доля самоцитирований для любого из авторов статьи, а также по совокупности всех авторов статьи, не должна превышать 25 %; доля ссылок на статьи с участием одного автора, не являющегося автором (соавтором) статьи, не должна превышать 25 %.

6. Суммарная доля таблиц и иллюстраций в общем объеме представляемой статьи не должна превышать 40 %. Под иллюстрациями понимаются следующие объекты: диаграммы; графики; рисунки; эскизы; фотографии; карты и т. п.

7. Доля оригинального текста в статьях (оцениваемого через систему «Антиплагиат» на сайте www.antiplagiat.ru) должна быть не менее 80 %.

8. Указание на то, что работа финансируется по какому-либо гранту, в рамках Федеральной целевой программы, государственного заказа и пр. дается в виде постраничной сноски после заголовка (названия) работы.

9. В сведения об авторах работ помимо места работы и должности целесообразно включать ORCID автора и гиперссылку на страничку с его личными наукометрическими показателями на сайте www.elibrary.ru. По желанию можно привести также ссылки на странички с наукометрическими показателями на Scopus, в ResearchGate; на личную страничку, размещенную на сайте организации.

10. Основные технические требования к оформлению статей (материалов):

10.1. Текст должен быть расположен по ширине страницы формата А4 с учётом полей (все поля по 2,5 см), набран шрифтом Times New Roman, кегль 12, межстрочный интервал 1,0. В таблицах, подрисовочных надписях допускается уменьшенный шрифт – вплоть до 10 кегля. Альбомная ориентация страниц допускается только в порядке исключения для следующих случаев: широкоформатные таблицы с большим количеством колонок; иллюстрации большого размера, которые не умещаются на странице с книжной ориентацией.

Абзацные отступы одинаковы по всему тексту – 0,75 см. Кавычки («»), скобки ([], ()), маркеры и другие знаки должны быть аналогичными на протяжении всего предоставляемого для публикации материала.

ПРИКАСПИЙСКИЙ ЖУРНАЛ: управление и высокие технологии

НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

**2024
№ 2 (66)**

Свидетельство о регистрации средства массовой информации
Федеральной службы по надзору в сфере массовых коммуникаций,
связи и охраны культурного наследия
ПИ № ФС77-31932 от 16 мая 2008 г.

Учредитель
Астраханский государственный университет имени В.Н. Татищева
Российская Федерация, 414056, г. Астрахань, ул. Татищева, 20а

Адрес редакции:
Российская Федерация, 414056, г. Астрахань, ул. Татищева, 20

Адрес издателя:
Российская Федерация, 414056, г. Астрахань, ул. Татищева, 20а

Издание включено в Интернет-каталог
ООО «Агентство «Книга-Сервис» 2022/1

Главный редактор И.М. Ажмухамедов

Редактирование,
компьютерная правка, верстка *Н.Н. Сахно*

Дата выхода в свет **30.06.2022 г.**

Цена свободная
Уч.-изд. 12,3. Усл. печ. л. 17,2.
Заказ № 4413. Тираж 500 экз. (первый завод – 22 экз.)

Астраханский государственный университет имени В.Н. Татищева
414056, г. Астрахань, ул. Татищева, 20а Тел. (8512) 24-64-95, тел./факс (8512) 24-68-37
E-mail: asupress@yandex.ru

Отпечатано в АО «Т 8 Издательские Технологии» (АО «Т 8»)
г. Москва, Волгоградский проспект, дом 42, корп. 5