ISSN 1818-1015
eISSN 2313-5417

2025 Volume 32 No. 1

MODELING AND ANALYSIS OF INFORMATION SYSTEMS

SCIENTIFIC JOURNAL

Start date of publication — 1999
Published quarterly

FOUNDER
P.G. Demidov Yaroslavl State University

EDITORIAL OFFICE
14 Sovetskaya str., Yaroslavl 150003, Russian Federation

Website: http://mais-journal.ru
E-mail: mais@uniyar.ac.ru
Phone: +7 (4852) 79-77-73

2025 Том 32 № 1

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ

НАУЧНЫЙ ЖУРНАЛ

Издается с 1999 года
Выходит 4 раза в год

УЧРЕДИТЕЛЬ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Ярославский государственный университет им. П. Г. Демидова»

РЕДАКЦИЯ
ул. Советская, 14, Ярославль, 150003, Российская Федерация
Website: http://mais-journal.ru
E-mail: mais@uniyar.ac.ru
Телефон: +7 (4852) 79-77-73

Свидетельство о регистрации СМИ ПИ № ФС77-66186 от 20.06.2016 выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций. Подписной индекс в каталоге «Урал-Пресс» — 31907. Технический редактор, компьютерная вёрстка — К. В. Лагутина. Дата выхода в свет 31.03.2025. Формат 200×265 мм. Объ-ем 94 с. Тираж 24 экз. Свободная цена. Издатель и его адрес: Ярославский государственный университет им. П. Г. Демидова; ул. Советская, 14, Ярославль, 150003, Россия. Типография и ее адрес: ИД «Канцлер»; Полушкина роща, д. 16, стр. 66-а, Ярославль, 150064, Россия.
Содержание предназначено для детей старше 12 лет.

Editor-in-Chief

Egor V. Kuzmin . . . . . Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)

Deputy Editor-in-Chief

Vladimir A. Bashkin . . . . . Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)

Editorial Board Secretary

Ilya V. Paramonov . . . . . Ph.D., P.G. Demidov Yaroslavl State University (Russia)

The Editorial Board

Sergei M. Abramov . . . . . Professor, Doctor of Sciences, Corresponding Member of Russian Academy of Sciences, Program Systems Institute of RAS (Pereslavl-Zalesskiy, Russia)
Lilian Aveneau . . . . . Professor, XLIM Laboratory, University of Poitiers (Poitiers, France)
Thomas Baar . . . . . Professor, Doctor, Hochschule für Technik und Wirtschaft Berlin, University of Applied Sciences (Berlin, Germany)
Olga L. Bandman . . . . . Professor, Doctor of Sciences, Supercomputer Software Department, Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk, Russia)
Vladimir N. Belykh . . . . . Professor, Doctor of Sciences, Volga State Academy of Water Transport (Nizhny Novgorod, Russia)
Vladimir A. Bondarenko . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Richard R. Brooks . . . . . Professor, Clemson University (South Carolina, USA)
Alex Dekhtyar . . . . . Professor, California Polytechnic State University (Cal Poly, California, USA)
Mikhail Dmitriev . . . . . Professor, Doctor of Sciences, Higher School of Economics (Moscow, Russia)
Vladimir L. Dolnikov . . . . . Doctor of Sciences, Moscow Institute of Physics and Technology (Moscow, Russia)
Valery G. Durnev . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Sergey D. Glyzin . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Yuri G. Karpov . . . . . Professor, Doctor of Sciences, St-Petersburg State Polytechnical University (Russia)
Sergey A. Kashchenko . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Lev S. Kazarin . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Andrei Yu. Kolesov . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Olga Kouchnarenko . . . . . Professor at the Burgundy-Franche-Comte University, The FEMTO-ST Institute (CNRS 6174) (Besancon, France)
Nikolai A. Kudryashov . . . . Professor, Doctor of Sciences, MEPhI (Russia)
Irina A. Lomazova . . . . . Professor, Doctor of Sciences, Higher School of Economics (Moscow, Russia)
George G. Malinetskiy . . . . Professor, Doctor of Sciences, M.V. Keldysh Institute of Applied Mathematics RAS (Moscow, Russia)
Victor E. Malyshkin . . . . . Professor, Doctor of Sciences, Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk, Russia)
Alexander V. Mikhailov . . . Professor, Doctor of Sciences, University of Leeds, School of Mathematics (Leeds, Great Britain)
Nikolai Kh. Rozov . . . . . Professor, Doctor of Sciences, Lomonosov Moscow State University (Russia)
Philippe Schnoebelen . . . . . Senior Researcher, LSV, CNRS & ENS de Cachan (CACHAN, France)
Natalia Sidorova . . . . . Dr., Assistant Professor, Architecture of Information Systems group, Technische universiteit Eindhoven (Eindhoven, Netherlands)
Ruslan L. Smeliansky . . . . . Professor, Doctor of Sciences, Corresponding Member of RAS, Lomonosov Moscow State University (Russia)
Valery A. Sokolov . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Javid Taheri . . . . . Associate Professor, Ph.D., Karlstad University (Sweden)
Eugeniy A. Timofeev . . . . . Professor, Doctor of Sciences, P.G. Demidov Yaroslavl State University (Russia)
Mark Trakhtenbrot . . . . . Dr., Holon Institute of Technology (Holon, Israel)
Dimitry Turaev . . . . . Professor of Applied Mathematics & Mathematical Physics, Imperial College (London, Great Britain)
Vladimir Zakharov . . . . . Doctor of Sciences, Professor, Lomonosov Moscow State University (Russia)

Главный редактор

Е. В. Кузьмин . . . . . д-р физ.-мат. наук, ЯрГУ (Россия)

Заместитель главного редактора

В. А. Башкин . . . . . д-р физ.-мат. наук, ЯрГУ (Россия)

Ответственный секретарь

И. В. Парамонов . . . . . канд. физ.-мат. наук, ЯрГУ (Россия)

Редакционная коллегия

С. М. Абрамов . . . . . д-р физ.-мат. наук, чл.-корр. РАН, Институт программных систем РАН им. А.К. Айламазяна (Россия)
L. Aveneau . . . . . проф., Университет Пуатье (Франция)
T. Baar . . . . . д-р наук, проф., Университет прикладных технических и экономических наук Берлина (Германия)
О. Л. Бандман . . . . . д-р техн. наук, Институт вычислительной математики и математической геофизики СО РАН (Россия)
В. Н. Белых . . . . . д-р физ.-мат. наук, проф., Волжская государственная академия водного транспорта (Россия)
В. А. Бондаренко . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
R. Brooks . . . . . проф., Университет Клемсона (США)
С. Д. Глызин . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
A. Dekhtyar . . . . . проф., Калифорнийский политехнический университет, департамент компьютерных наук (США)
М. Г. Дмитриев . . . . . д-р физ.-мат. наук, проф., ВШЭ (Россия)
В. Л. Дольников . . . . . д-р физ.-мат. наук, проф., МФТИ (Россия)
В. Г. Дурнев . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
В. А. Захаров . . . . . д-р физ.-мат. наук, проф., МГУ (Россия)
Л. С. Казарин . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
Ю. Г. Карпов . . . . . д-р техн. наук, проф., Санкт-Петербургский государственный технический университет (Россия)
С. А. Кащенко . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
А. Ю. Колесов . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
Н. А. Кудряшов . . . . . д-р физ.-мат. наук, проф., Засл. деятель науки РФ, МИФИ (Россия)
O. Kouchnarenko . . . . . проф., Университет Бургундии - Франш-Комтэ (Франция)
И. А. Ломазова . . . . . д-р физ.-мат. наук, проф., ВШЭ (Россия)
Г. Г. Малинецкий . . . . . д-р физ.-мат. наук, проф., Институт прикладной математики им. М.В. Келдыша РАН (Россия)
В. Э. Малышкин . . . . . д-р техн. наук, проф., Институт вычислительной математики и математической геофизики СО РАН (Россия)
A. Mikhailov . . . . . д-р физ.-мат. наук, проф., Университет Лидса (Великобритания)
Н. Х. Розов . . . . . д-р физ.-мат. наук, проф., чл.-корр. РАО, МГУ (Россия)
N. Sidorova . . . . . д-р наук, университет Эйндховена (Нидерланды)
Р. Л. Смелянский . . . . . д-р физ.-мат. наук, проф., член-корр. РАН, академик РАЕН, МГУ (Россия)
В. А. Соколов . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
J. Taheri . . . . . доцент, Университет Карлстада (Швеция)
Е. А. Тимофеев . . . . . д-р физ.-мат. наук, проф., ЯрГУ (Россия)
M. Trakhtenbrot . . . . . д-р комп. наук, Холонский технологический институт (Израиль)
D. Turaev . . . . . проф., Имперский колледж Лондона (Великобритания)
Ph. Schnoebelen . . . . . проф., Национальный центр научных исследований и Высшая нормальная школа Кашана (Франция)

Contents

Содержание

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

COMPUTING METHODOLOGIES AND APPLICATIONS

Algorithm for Constructing Asymptotics of Periodic Solutions in Laser Models with a Rapidly Oscillating Delay

E. V. Grigorieva[1], D. V. Glazkov[2], A. O. Tolbey[2]

DOI: 10.18255/1818-1015-2025-1-6-15

[1]Belarus State Economic University, Minsk, Belarus
[2]P.G. Demidov Yaroslavl State University, Yaroslavl, Russia

MSC2020: 39A11, 34K18
Research article
Full text in Russian

Received December 17, 2024
Revised January 13, 2025
Accepted January 22, 2025

The problem of stability of the equilibrium state in a laser system with fast oscillating coefficients is considered. A system averaged over fast oscillations and with a distributed delay is constructed. Critical cases in the problem of the stability of the equilibrium state are singled out. It is shown that the threshold value of the feedback coefficient at which the equilibrium state becomes unstable increases due to rapid oscillations compared to the corresponding value in the absence of modulation. In critical cases, normal forms are constructed — equations for the slowly varying amplitude of the periodic solutions. The conditions for the existence, stability and instability of cycles are revealed.

Keywords: bifurcation analysis; wave structures; delay; laser dynamics

INFORMATION ABOUT THE AUTHORS

Grigorieva, Elena V.	ORCID iD: 0000-0003-4543-2598. E-mail: grigorieva@tutby.news Professor
Glazkov, Dmitry V.	ORCID iD: 0000-0003-0511-5088. E-mail: d.glazkov@uniyar.ac.ru Associate Professor, PhD
Tolbey, Anna O. (corresponding author)	ORCID iD: 0000-0001-5668-3929. E-mail: a.tolbey@uniyar.ac.ru Associate Professor, PhD

Funding: This work was carried out within the framework of a development programme for the Regional Scientific and Educational Mathematical Center of the Yaroslavl State University with financial support from the Ministry of Science and Higher Education of the Russian Federation (Agreement on provision of subsidy from the federal budget No. 075-02-2024-1442).

For citation: E. V. Grigorieva, D. V. Glazkov, and A. O. Tolbey, "Algorithm for constructing asymptotics of periodic solutions in laser models with a rapidly oscillating delay", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 6–15, 2025. DOI: 10.18255/1818-1015-2025-1-6-15.

© Grigorieva E. V., Glazkov D. V., Tolbey A. O., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

COMPUTING METHODOLOGIES AND APPLICATIONS

Алгоритм построения асимптотики периодических решений в моделях лазеров с быстро осциллирующей задержкой

Е. В. Григорьева[1], Д. В. Глазков[2], А. О. Толбей[2]

DOI: 10.18255/1818-1015-2025-1-6-15

[1]Белорусский государственный экономический университет, Минск, Беларусь
[2]Ярославский государственный университет им. П.Г. Демидова, Ярославль, Россия

УДК 519.7
Научная статья
Полный текст на русском языке

Получена 17 декабря 2024 г.
После доработки 13 января 2025 г.
Принята к публикации 22 января 2025 г.

Рассматривается задача об устойчивости состояния равновесия в лазерной системе с быстро осциллирующими коэффициентами. Построена усредненная по быстрым осцилляциям система с распределенным запаздыванием. Выделены критические случаи в задаче об устойчивости состояния равновесия. Показано, что пороговое значение коэффициента обратной связи, при котором состояние равновесия становится неустойчивым, увеличивается вследствие быстрых осцилляций по сравнению с соответствующим значением при отсутствии модуляции. В критических случаях построены нормальные формы — уравнения для медленной амплитуды периодических решений. Выявлены условия существования, устойчивости и неустойчивости циклов.

Ключевые слова: бифуркационный анализ; волновые структуры; запаздывание; лазерная динамика

ИНФОРМАЦИЯ ОБ АВТОРАХ

Григорьева, Елена Викторовна	ORCID iD: 0000-0003-4543-2598. E-mail: grigorieva@tutby.news Профессор
Глазков, Дмитрий Владимирович	ORCID iD: 0000-0003-0511-5088. E-mail: d.glazkov@uniyar.ac.ru Доцент, канд. физ.-мат. наук
Толбей, Анна Олеговна (автор для корреспонденции)	ORCID iD: 0000-0001-5668-3929. E-mail: a.tolbey@uniyar.ac.ru Доцент, канд. физ.-мат. наук

Финансирование: Работа выполнена в рамках реализации программы развития регионального научно-образовательного математического центра (ЯрГУ) при финансовой поддержке Министерства науки и высшего образования РФ (Соглашение о предоставлении из федерального бюджета субсидии № 075-02-2024-1442).

Для цитирования: E. V. Grigorieva, D. V. Glazkov, and A. O. Tolbey, “Algorithm for constructing asymptotics of periodic solutions in laser models with a rapidly oscillating delay”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 6–15, 2025. DOI: 10.18255/1818-1015-2025-1-6-15.

© Григорьева Е. В., Глазков Д. В., Толбей А. О., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

Запаздывающая обратная связь (ОС) применяется как для стабилизации состояния равновесия в нелинейных системах, так и для установления сложных осциллирующих режимов с заданными характеристиками. С целью расширения области эффективного управления режимами предлагались различные схемы ОС [1—4], а также дополнительная периодическая модуляция параметров ОС — уровня ОС [5] или времени задержки [6, 7]. Характеристики сложных режимов, индуцированных периодической модуляцией параметров ОС, могут быть востребованы во многих практических задачах. Например, в задаче об измерении расстояния до вибрирующей поверхности, формирующей внешний резонатор [8], в задаче о динамике импульсов излучения в лазерных диодах при малых вариациях длины резонатора [9]. Изменения характеристик ОС в пределах диапазона, обеспечивающего хаотический режим генерации, предлагались и для кодирования информации [10, 11].

В данной работе рассматривается вопрос об устойчивости состояния равновесия при высокочастотной модуляции параметров обратной связи. По аналогии с классическими результатами в задаче о динамике маятника с вибрирующим подвесом [12, 13] можно ожидать стабилизацию неустойчивого равновесия. Для лазерной модели построена усредненная по быстрым осцилляциям система с распределенным запаздыванием. Для усредненной системы выделены критические случаи в задаче об устойчивости состояния равновесия и показано, что вследствие быстрых осцилляций запаздывания граница неустойчивости в пространстве параметров смещается в сторону больших значений коэффициента обратной связи. Зависимость величины смещения от амплитуды модуляции имеет зонную структуру, поэтому быстрые осцилляции запаздывания могут как стабилизировать, так и дестабилизировать состояние равновесия.

1. Постановка задачи

Изучим вопрос об устойчивости состояния равновесия при высокочастотной модуляции параметров обратной связи (ОС). В схеме оптической ОС, предложенной в работе [14], поляризация отраженного от внешнего зеркала света ортогональна к поляризации падающего света, что позволяет не учитывать когерентное взаимодействие встречных волн электрической поля в математической модели. Динамика генерации такого устройства описывается одномодовыми скоростными уравнениями с запаздывающим аргументом:

$$\begin{aligned} \dot{u} &= vu(y-1), \\ \dot{y} &= q - y - y[u + \gamma u(t-T)], \end{aligned} \tag{1}$$

где u и y пропорциональны плотностям фотонов и инверсии населенностей, соответственно, q — скорость накачки, v — отношение скорости затухания фотонов в резонаторе к скорости релаксации населенностей, потери резонатора нормированы к единице, t и T — текущее время и время прохода излучения по внешнему резонатору, нормированные на время релаксации инверсии населенностей, γ — коэффициент обратной связи. Отметим, что предлагаемая ниже методика может быть использована и для модели Лэнга — Кобаяши [15, 16] с учетом когерентного взаимодействия встречных волн.

Сначала кратко остановимся на свойствах решений системы (1) с постоянным запаздыванием. Система имеет два состояния равновесия. Первое состояние с нулевой плотностью излучения, $u(t) = 0$, $y(t) = q$, становится неустойчивым при достижении накачкой порогового значения $q = 1$. Далее будем полагать $q > 1$. При этом появляется второе состояния равновесия $u(t) = u_s$, $y(t) = y_s$, где

$$u_s(\gamma) = \frac{q-1}{1+\gamma}, \; y_s = 1,$$

устойчивость которого стандартно определяется поведением корней характеристического уравнения

$$\lambda^2 + \lambda q + v u_s + \gamma v u_s e^{-\lambda T} = 0. \tag{2}$$

При $\gamma = 0$ характеристические корни имеют отрицательные действительные части, поэтому рассматриваемое состояние равновесия устойчиво. Пусть $\gamma_0 > 0$ наименьшее положительное значение, при котором уравнение (2) имеет пару чисто мнимых корней $\lambda_{1,2} = \pm i\,\omega_0$ ($\omega_0 > 0$) и $\Re\lambda'_{1,2}(\gamma_0) > 0$, а все другие корни имеют отрицательные действительные части. Тогда при $\gamma \in (0, \gamma_0)$ состояние равновесия (u_s, y_s) системы (1) асимптотически устойчиво, а при $\gamma > \gamma_0$ — неустойчиво. Бифуркационные значения γ_0 и ω_0 находятся из системы уравнений:

$$\operatorname{tg}(\omega_0 T) = \frac{\omega_0 q}{\omega_0^2 - v u_s(\gamma_0)},$$

$$\big(v u_s(\gamma_0)\gamma_0\big)^2 = \big(\omega_0^2 - v u_s(\gamma_0)\big)^2 + \omega_0^2 q^2.$$

Особенности бифуркационной диаграммы для аналогичной (совпадающей в линейном приближении) системы подробно обсуждались в разделе 2 монографии [17], поэтому здесь на них не останавливаемся. Отметим только, что в окрестности границы устойчивости возможно образование устойчивых и неустойчивых циклов, 2-х и 3-х частотных торов.

В настоящей работе рассматривается более сложная по сравнению с (1) система, включающая периодическую модуляцию запаздывания с амплитудой a и частотой ω:

$$\begin{aligned} \dot u &= v u (y - 1), \\ \dot y &= q - y - y[u + \gamma u(t - T - a\sin\omega t)]. \end{aligned} \tag{3}$$

Осцилляции запаздывания могут быть, в частности, вызваны вибрацией зеркал, образующих внешний резонатор, или организованы оптоэлектронными средствами. По физическому смыслу амплитуда осцилляций ограничена,

$$|a| \leqslant T.$$

Основное предположение, открывающее путь к применению асимптотических методов, состоит в том, что параметр ω является достаточно большим:

$$\omega \gg 1. \tag{4}$$

Отметим, что ω должна существенно превышать и значение собственной частоты ω_0, которая для твердотельных лазеров имеет порядок порядка $v^{1/2}$ и достаточно велика. Тем не менее, экспериментально такую ситуацию возможно реализовать оптоэлектронными средствами. Описанные ниже эффекты наблюдаются при численном моделировании уже при $\omega > 3\omega_0$. Еще отметим, что нелокальные режимы релаксационных колебаний при модуляции запаздывания с частотой, сопоставимой с собственной частотой лазера, изучались в работе [18] с помощью специального метода редукции системы к двумерному отображению.

2. Усредненная система

Прежде чем приступить к изучению систем с запаздыванием и быстро осциллирующими коэффициентами, напомним основные результаты, касающиеся принципа усреднения [19]. Для систем обыкновенных дифференциальных уравнений вида

$$\dot x = f(\omega t, x), \tag{5}$$

где $\omega \gg 1$, а вектор-функция $f(\Gamma, x)$ периодична (почти периодична) по первому аргументу $\Gamma = \omega t$ и достаточно гладкая по второму аргументу, показано, что при выполнении ряда естественных

ограничений поведение решений (5) определяется в главном поведением решений более простой — автономной — системой

$$\dot{x} = f_0(x), \text{ где } f_0(x) = \lim_{t_0\to\infty} \frac{1}{t_0}\int_0^{t_0} f(\Gamma, x)d\Gamma.$$

Соответствующее обоснование базируется на том, что в результате замены времени $\omega t = \Gamma$ система (5) принимает вид

$$\frac{dx}{d\Gamma} = \omega^{-1} f(\Gamma, x),$$

т. е. производная $\frac{dx}{d\Gamma}$ оказывается в некотором смысле достаточно малой. Для систем с запаздыванием такую замену времени выполнять нецелесообразно, поскольку промежуток запаздывания тогда неограниченно возрастает при $\omega \to \infty$. Тем не менее и для них принцип усреднения имеет место [20]. Присутствие запаздывания может приводить к новым динамическим эффектам, которые не встречаются в системах обыкновенных дифференциальных уравнений.

Рассмотрим систему (3). Фиксируем в произвольный момент t_0 начальные условия $u(t_0+\theta),\ \theta \in [-T-a, 0]$ и $y(t_0)$, подставим их в правую часть системы (3) и произведем усреднение по быстрому времени $\Gamma = \omega t$:

$$\begin{aligned} \dot{u} &= vu(y-1), \\ \dot{y} &= q - y - y\left[u + \gamma\frac{\omega}{2\pi}\int_0^{2\pi/\omega} u(t-T-a\cos(\omega s))ds\right]. \end{aligned}$$

Слагаемое с быстро осциллирующим запаздывающим аргументом при усреднении преобразуем следующим образом. Сделаем замену переменной $s_1 = \cos(\omega s)$, $ds_1 = -\omega\sin(\omega s)ds$ и учтем, что $\sin(\omega s) = \pm\sqrt{1-s_1^2}$ на интервалах $s \in [0, \pi/\omega]$ и $s \in [\pi/\omega, 2\pi/\omega]$, тогда получим

$$\frac{\omega}{2\pi}\int_0^{2\pi/\omega} u(t-T-a\cos(\omega s))ds = \frac{1}{\pi}\int_{-1}^{0}\frac{u(t-T-as_1)+u(t-T+as_1)}{\sqrt{1-s_1^2}}ds_1.$$

Окончательно, опуская индекс в формальной переменной интегрирования, приходим к усредненной системе уравнений с распределенным запаздыванием:

$$\begin{aligned} \dot{u} &= vu(y-1), \\ \dot{y} &= q - y - y\left[u + \frac{\gamma}{\pi}\int_{-1}^{0}\frac{u(t-T-as)+u(t-T+as)}{\sqrt{1-s^2}}ds\right]. \end{aligned} \tag{6}$$

Связь между решениями $u(t,\omega), y(t,\omega)$ системы (3) и решениями $u(t), y(t)$ усредненной системы (6) с одинаковыми начальными условиями из фазового пространства $C_{[-T-a,0]} \times R^1$ устанавливают асимптотические формулы

$$u(t,\omega) = u(t) + O(\omega^{-1}),\ y(t,\omega) = y(t) + O(\omega^{-1})$$

на каждом отрезке $[t_0, t_0+L]$, где $L > 0$ произвольно и фиксировано.

Справедлив также следующий вывод о близости режимов этих динамических систем. Пусть (6) имеет грубый цикл $u_0(t), y_0(t)$ (тор размерности k). Тогда при достаточно больших частотах ω система (3) имеет 2-мерный тор $u_0(t,\omega), y_0(t,\omega)$ (тор размерности $k+1$) той же устойчивости, для которого выполнены асимптотические равенства

$$u_0(t,\omega) = u_0\Big((1+o(1))t\Big) + o(1),\ y_0(t,\omega) = y_0\Big((1+o(1))t\Big) + o(1).$$

3. Устойчивость состояния равновесия усредненной системы

Состояния равновесия в системе (3), содержащей осцилляции в запаздывании, и в усредненной системе (6) одни и те же. Будем исследовать далее режимы системы (6) в окрестности ненулевого состояния равновесия $y = y_s$, $u = u_s$. Для анализа его устойчивости рассмотрим линеаризованную систему для малых отклонений $u_1(t) = u(t) - u_s$, $y_1(t) = y(t) - y_s$:

$$\begin{aligned} \dot{u}_1 &= v u_s y_1, \\ \dot{y}_1 &= -q y_1 - u_1 - \frac{\gamma}{\pi} \int_{-1}^{0} \frac{u_1(t - T - as) + u_1(t - T + as)}{\sqrt{1 - s^2}} ds, \end{aligned} \tag{7}$$

и ее характеристический квазиполином

$$\lambda^2 + q\lambda + v u_s + \gamma v u_s \ell(\lambda) = 0, \tag{8}$$

где

$$\ell(\lambda) = e^{-T\lambda} \frac{2}{\pi} \int_{-1}^{0} (1 - s^2)^{-1/2} \operatorname{ch}(\lambda a s) ds.$$

Как и для квазиполинома (2), определим условия для параметра γ, при которых корни квазиполинома (8) имеют отрицательные вещественные части. Положим $\lambda = i\omega_0$, $\gamma = \gamma_0$, ($\omega_0 = \omega_0(a) > 0$, $\gamma_0 = \gamma_0(a) > 0$) и получим систему

$$\operatorname{tg}(\omega_0 T) = \frac{\omega_0 q}{\omega_0^2 - v u_s(\gamma_0)}, \tag{9}$$

$$\left[\omega_0^2 - v u_s(\gamma_0)\right]^2 + q^2 \omega_0^2 = \left[\gamma_0 v u_s(\gamma_0)\right]^2 J_0^2(a\omega_0), \tag{10}$$

в которой $J_0(x)$ — функция Бесселя нулевого порядка

$$J_0(x) = \frac{2}{\pi} \int_{-\frac{\pi}{2}}^{0} \cos(x \sin s) ds.$$

При условии $0 < \gamma < \gamma_0(a)$ состояние равновесия (u_s, y_s) системы (6) асимптотически устойчиво, а при $\gamma > \gamma_0(a)$ — неустойчиво. На основании принципа усреднения заключаем, что для системы (3) имеет место следующее утверждение.

Утверждение 1. Пусть $\gamma \in (0, \gamma_0(a))$ $\quad (\gamma > \gamma_0(a))$. Тогда найдется такое $\Omega > 0$, что при всех $\omega > \Omega$ состояние равновесия (u_s, y_s) системы (3) асимптотически устойчиво (неустойчиво).

Обратим внимание, что

$$J_0^2(0) = 1, \quad \left.\frac{dJ_0(a\omega_0)}{da}\right|_{a=0} = 0, \quad \left.\frac{d^2 J_0(a\omega_0)}{da^2}\right|_{a=0} < 0,$$

а значит, при малых положительных значениях параметра a выполнено неравенство

$$\gamma_0(a) > \gamma_0(0).$$

Отсюда вытекает вывод о том, что быстрые осцилляции (с ненулевым средним) запаздывания могут расширять область $\gamma \in (0, \gamma_0)$ устойчивости состояния равновесия системы (3) в пространстве параметров и тем самым стабилизировать систему.

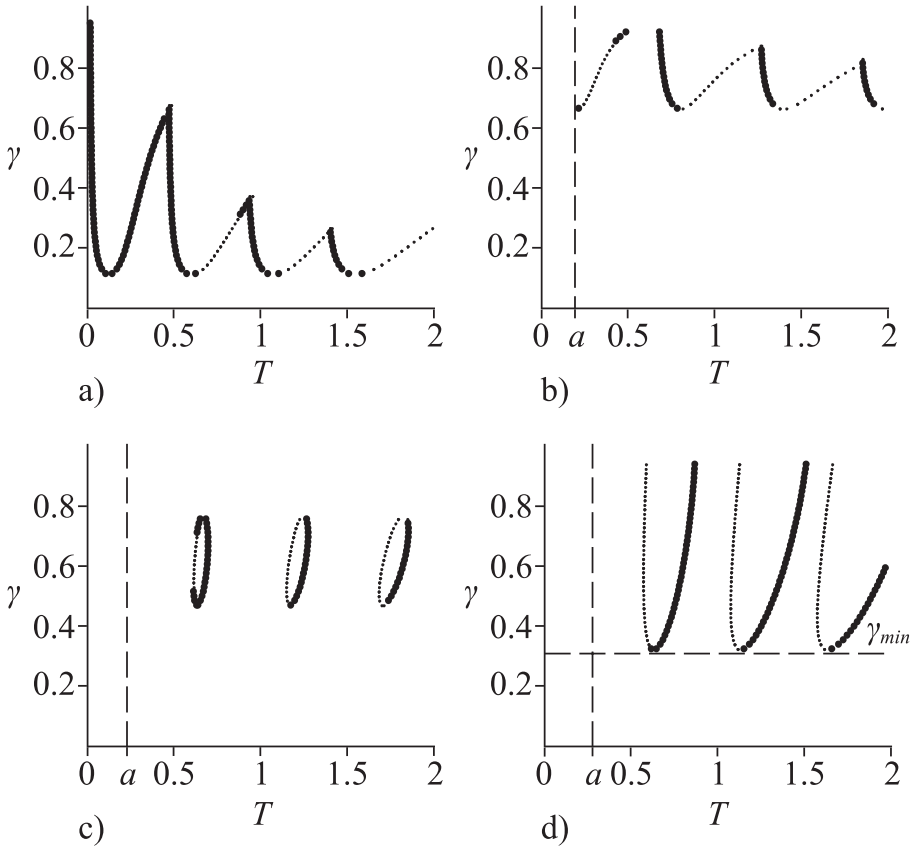

Fig. 1. Boundaries of the region of stability of the stationary state in the plane of the feedback parameters (γ, T) for the system (3) with $\upsilon = 400, q = 1.5$, with high-frequency modulation of the delay with amplitude a) $a = 0$, b) $a = 0.19$, c) $a = 0.26$, d) $a = 0.29$. The sections of the bifurcation curve on which the Lyapunov exponent is positive (negative) are indicated by a dotted line (bold line). The vertical dashed lines mark the minimum value of the delay time $T = a$.

Рис. 1. Границы области устойчивости стационарного состояния в плоскости параметров ОС (γ, T) для системы (3) при $\upsilon = 400, q = 1.5$ при высокочастотной модуляции запаздывания с амплитудой а) $a = 0$, b) $a = 0.19$, c) $a = 0.26$, d) $a = 0.29$. Участки бифуркационной кривой, на которых ляпуновская величина положительна (отрицательна), обозначены пунктиром (жирной линией). Вертикальные штриховые линии отмечают минимальное значение времени задержки $T = a$.

На рис. 1 представлены бифуркационные диаграммы в плоскости параметров обратной связи (T, γ), вычисленные по формулам (9) и (10) в случае а) постоянного запаздывания при $a = 0$ и в случаях b)–d) высокочастотной ($\omega \gg 1$) модуляции запаздывания при возрастающих значениях a. Видно, что для фиксированного T при увеличении амплитуды модуляции a значения γ_0, при которых состояние равновесия теряет устойчивость, сначала увеличиваются. На интервале $a \in [0.22, 0.25]$ при всех $T > 0$ значения $\gamma_0 > 1$, т. е. нет областей неустойчивости при физически допустимых уровнях ОС. Затем при $a > 0.25$ опять появляются области неустойчивости, которые далее опять сдвигаются вверх. Таким образом, наблюдается зонная структура областей неустойчивости.

Для объяснения зонной структуры воспользуемся тем фактом, что для полупроводниковых лазеров параметр υ принимает достаточно большое значение, $\upsilon \sim 10^3$. Тогда можно оценить минимальное значение коэффициента ОС γ_0:

$$\min\{\gamma_0(a)\} = \frac{1}{|J_0(a\omega_0)|}\frac{q}{\sqrt{\upsilon(q-1)}}\left(1 + O(\upsilon^{-1/2})\right).$$

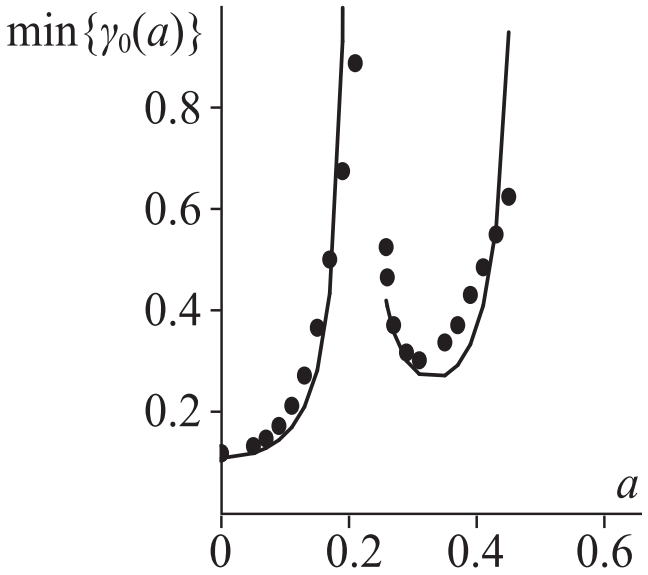

Fig. 2. Dependence of the minimum value $\min\{\gamma_0(a)\}$ of the feedback coefficient for which regions of instability of the equilibrium state appear, on the amplitude a of the modulation of the delay; other parameters of the system are $\upsilon = 400$, $q = 1.5$. On the interval $a \in [0.22, 0.25]$ the values $\gamma_0 > 1$.

Рис. 2. Зависимость минимального значения коэффициента ОС $\min\{\gamma_0(a)\}$, при котором возникают области неустойчивости равновесного состояния, от амплитуды модуляции запаздывания a, другие параметры системы $\upsilon = 400$, $q = 1.5$. На интервале $a \in [0.22, 0.25]$ значения $\gamma_0 > 1$.

Отметим, что в отсутствие модуляции запаздывания $a = 0$ и $J_0(0) = 1$. При $a > 0$ и $|J_0(a\omega_0)| < 1$ граница неустойчивости сдвигается в сторону больших значений γ. В точках $a = a_k$, $k = 1, 2, \ldots$, соответствующих нулям функции Бесселя $J_0(a_k\omega_0) = 0$, минимальное критическое значение стремится к бесконечности, следовательно, области неустойчивости равновесного состояния отсутствуют.

График зависимости минимального бифуркационного значения $\min\{\gamma_0(a)\}$, при котором возникает неустойчивость, от амплитуды модуляции запаздывания представлен на рис. 2. В частности, действительно существует интервал значений a, для которых минимальное критическое значение коэффициента ОС $\gamma_0 > 1$, т. е. области неустойчивости отсутствуют.

Заключение

Задача об устойчивости состояния равновесия в лазерной системе с быстро осциллирующими коэффициентами была решена с помощью построения усредненной по быстрым осцилляциям системы с распределенным запаздыванием. Для усредненной системы выделены критические случаи. Показано, что пороговое значение коэффициента ОС, при котором состояние равновесия становится неустойчивым, увеличивается вследствие быстрых осцилляций по сравнению с соответствующим значением при отсутствии модуляции. Зависимость величины смещения порога от амплитуды модуляции имеет зонную структуру. В частности, существуют интервалы значений амплитуды модуляции, при которых равновесное состояние остается устойчивым при любом (физически допустимом) уровне ОС. В критических случаях построены нормальные формы — уравнения для медленной амплитуды $\xi(t)$ периодических решений. Рассчитана ляпуновская величина, которая определяет направление бифуркации (суб- или суперкритическое), вследствие чего в окрестности границы возможно образование устойчивого или неустойчивого циклов.

References

[1] A. Kittel, K. Pyragas, and R. Richter, "Prerecorded history of a system as an experimental tool to control chaos", *Physical Review E*, vol. 50, no. 1, pp. 262–268, 1994. DOI: 10.1103/PhysRevE.50.262.

[2] K. Pyragas, "Control of chaos via an unstable delayed feedback controller", *Physical Review Letters*, vol. 86, no. 11, pp. 2265–2268, 2001. DOI: 10.1103/PhysRevLett.86.2265.

[3] K. Pyragas, V. Pyragas, I. Z. Kiss, and J. L. Hudson, "Stabilizing and tracking unknown steady states of dynamical systems", *Physical Review Letters*, vol. 89, no. 24, p. 244 103, 2002. DOI: 10.1103/PhysRevLett.89.244103.

[4] A. Ahlborn and U. Parlitz, "Controlling dynamical systems using multiple delay feedback control", *Physical Review E*, vol. 71, no. 1, p. 016 206, 2005. DOI: 10.1103/PhysRevE.72.016206.

[5] H. G. Schuster and M. P. Stemmler, "Control of chaos by oscillating feedback", *Physical Review E*, vol. 56, no. 6, pp. 6410–6417, 1997. DOI: 10.1103/PhysRevE.56.6410.

[6] A. Gjurchinovski and V. Urumov, "Variable-delay feedback control of unstable steady states in retarded time-delayed systems", *Physical Review E*, vol. 81, no. 1, p. 016 209, 2010. DOI: 10.1103/PhysRevE.81.016209.

[7] T. Jungling, A. Gjurchinovski, and V. Urumov, "Experimental time-delayed feedback control with variable and distributed delays", *Physical Review E*, vol. 86, no. 4, p. 046 213, 2012. DOI: 10.1103/PhysRevE.86.046213.

[8] A. V. Skripal, D. A. Usanov, V. A. Vagarin, and M. Y. Kalinkin, "Autodyne detection in a semiconductor laser as the external reflector is moved", *Technical Physics*, vol. 44, pp. 66–68, 1999. DOI: 10.1134/1.1259253.

[9] J. Martin-Regalado, G. H. M. Tartwijk, S. Balle, and M. S. Miguel, "Mode control and pattern stabilization in broad-area lasers by optical feedback", *Physical Review A*, vol. 54, no. 6, pp. 5386–5393, 1996. DOI: 10.1103/PhysRevA.54.5386.

[10] T. Yang, C. W. Wu, and L. O. Chua, "Cryptography based on chaotic systems", *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 44, no. 5, pp. 469–472, 1997. DOI: 10.1109/81.572346.

[11] J.-P. Goedgebuer, L. Larger, and H. Porte, "Optical cryptosystem based on synchronization of hyperchaos generated by delayed feedback tunable laser diode", *Physical Review Letters*, vol. 80, no. 10, pp. 2249–2252, 1998. DOI: 10.1103/PhysRevLett.80.2249.

[12] N. N. Bogoliubov and Y. A. Mitropolsky, *Asymptotic methods in the theory of non-linear oscillations*. Delhi : Hindustan Publishing Corporation (India), 1961.

[13] A. Stephenson, "On a new type of dynamical stability", *Memoirs and Proceedings of the Manchester Literary and Philosophical Society*, vol. 52, no. 8, pp. 1–10, 1908.

[14] J.-L. Chern, K. Otsuka, and F. Ishiyama, "Coexistence of two attractors in lasers with delayed incoherent optical feedback", *Optics Communications*, vol. 96, no. 4–6, pp. 259–266, 1993. DOI: 10.1016/0030-4018(93)90272-7.

[15] R. Lang and K. Kobayashi, "External optical feedback effects on semiconductor injection laser properties", *IEEE Journal of Quantum Electronics*, vol. 16, no. 3, pp. 347–357, 1980. DOI: 10.1109/JQE.1980.1070479.

[16] A. Levine, G. H. M. Tartwijk, D. Lenstra, and T. Erneux, "Diode lasers with optical feedback: Stability of the maximum gain mode", *Physical Review A*, vol. 52, no. 5, R3436–R3439, 1995. DOI: 10.1103/PhysRevA.52.R3436.

[17] E. V. Grigorieva, A. A. Kashchenko, and S. A. Kashchenko, *Local analysis of the dynamics of distributed laser models*. Moscow: LENAND, 2024, in Russian.

[18] E. Grigorieva, "Instabilities of periodic orbits in lasers with oscillating delayed feedback", *Nonlinear Phenomena in Complex Systems*, vol. 4, no. 1, pp. 6–12, 2001.

[19] Y. A. Mitropol'skii, *The method of averaging in nonlinear mechanics*. Kiev : Naukova Dumka, 1971, in Russian.

[20] Y. S. Kolesov, V. S. Kolesov, and I. I. Fedik, *Avtokolebaniya v sistemah s raspredelennymi parametrami*. Kiev : Naukova Dumka, 1979, in Russian.

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Extremal Estimates of the Wiener Index for Weakly Connected Directed Graphs

D. Y. Chalyy[1]

DOI: 10.18255/1818-1015-2025-1-16-31

[1]P.G. Demidov Yaroslavl State University, Yaroslavl, Russia

MSC2020: 05C45, 05C65, 05C85
Research article
Full text in Russian

Received February 15, 2025
Revised February 25, 2025
Accepted February 26, 2025

The article considers the Wiener index for weakly connected directed graphs. For such graphs, the distance $d(u, v)$ between vertices u and v is not always defined, which requires a correction for the Wiener index to be meaningful. The convention where it is assumed that $d(u, v) = 0$ in the absence of a path between vertices is well-studied. We consider the convention where $d(u, v)$ is equal to the number of vertices in the graph when there is no path between vertices u and v. The article presents graphs with n vertices for which the Wiener index with this convention reaches minimal and maximal values. We also present experimental results showing how the Wiener index (considering both conventions of distance) changes when arcs are added to a weakly connected directed graph with fixed and random structures.

Keywords: weakly connected graph; Wiener index

INFORMATION ABOUT THE AUTHORS

Chalyy, Dmitry Y. (corresponding author) | ORCID iD: 0000-0003-0553-7387. E-mail: chaly@uniyar.ac.ru
PhD, Associate professor, Dean of the Faculty of Information and Computer Science

Funding: Yaroslavl State University (project VIP-016).

For citation: D. Y. Chalyy, "Extremal estimates of the Wiener index for weakly connected directed graphs", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 16–31, 2025. DOI: 10.18255/1818-1015-2025-1-16-31.

© Chalyy D. Y., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Экстремальные оценки индекса Винера для слабо связных ориентированных графов

Д. Ю. Чалый[1]

DOI: 10.18255/1818-1015-2025-1-16-31

[1]Ярославский государственный университет им. П.Г. Демидова, Ярославль, Россия

УДК 519.17
Научная статья
Полный текст на русском языке

Получена 15 февраля 2025 г.
После доработки 25 февраля 2025 г.
Принята к публикации 26 февраля 2025 г.

В статье рассматривается индекс Винера для слабо связных ориентированных графов. Для таких графов из-за слабой связности не всегда определено расстояние $d(u, v)$ между вершинами u и v, что требует уточнения чтобы индекс Винера имел содержательный смысл. Достаточно хорошо изучен случай, когда полагают что $d(u, v) = 0$ при отсутствии пути между вершинами. Мы рассматриваем уточнение, когда $d(u, v)$ равно количеству вершин в графе при отсутствии пути между вершинами u и v. В статье представлены графы на n вершинах, где индекс Винера с таким уточнением достигает минимального и максимального значения. Мы также представляем результаты экспериментов, которые показывают как изменяется индекс Винера (с учетом обоих способов уточнения расстояния) при добавлении дуг в слабо связный ориентированный граф как фиксированной, так и случайной структуры.

Ключевые слова: слабо ориентированный граф; индекс Винера

ИНФОРМАЦИЯ ОБ АВТОРАХ

Чалый, Дмитрий Юрьевич (автор для корреспонденции) | ORCID iD: 0000-0003-0553-7387. E-mail: chaly@uniyar.ac.ru
Канд. физ.-мат. наук, доцент, декан факультета информатики и вычислительной техники

Финансирование: ЯрГУ (проект VIP-016).

Для цитирования: D. Y. Chalyy, “Extremal estimates of the Wiener index for weakly connected directed graphs”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 16–31, 2025. DOI: 10.18255/1818-1015-2025-1-16-31.

© Чалый Д. Ю., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

Индекс Винера (Wiener Index) [1], $W(G)$, является топологическим индексом связного неориентированного графа. Индекс определяется как сумма длин кратчайших путей между всеми парами вершин графа. Изначально этот индекс был разработан для изучения точки кипения парафина при помощи использования графовых моделей химических молекул. Данное приложение показало себя вполне успешным и ряд химических свойств смогли быть сопоставлены индексу Винера. Этот индекс является достаточно хорошо изученным (см. например, обзор в [2]), особенно в случае рассмотрения неориентированных графов. Одни из первых результатов, использующих индекс Винера для ориентированных графов, были получены в [3] при исследовании социометрических задач.

Формально индекс Винера для графа G определяется при помощи формулы

$$W(G) = \sum_{u,v \in V(G)} d_G(u,v),$$

где u, v — вершины графа, а $d_G(u,v)$ — расстояние (длина кратчайшего пути) между этими вершинами в графе G. Изначально индекс Винера рассматривался только для неориентированных связных графов, поскольку для этого класса графов расстояние между каждой парой вершин определено. В ориентированных графах могут существовать пары вершин u, v такие, что между ними отсутствует путь. В этом случае расстояние определяется как $d_G(u,v) = \infty$ и, следовательно, определение индекса Винера теряет содержательный смысл.

Существует два способа, позволяющих уточнить это определение для ориентированных графов. Первый способ (см. например, [4, 5]) определяет, что в случае отсутствия пути между вершинами u и v, полагается что $d_G(u,v) = 0$. В рамках этого подхода были получены результаты по оценке нижней границы индекса Винера для ориентированных графов [4], изучалась задача поиска минимальной и максимальной с точки зрения значения индекса Винера ориентации графа [5—7], были получены результаты поиска графов со следующим по убыванию наибольшим индексом Винера [8]. В работе [7] доказано, что для заданного графа задача определения ориентации рёбер, которая максимизирует величину индекса Винера, является NP-трудной.

Второй способ [9] устанавливает, что если между вершинами u и v отсутствует путь, то $d_G(u,v) = |V(G)|$. Эта статья имеет практический интерес, поскольку рассматривает применение графовых моделей для структурного анализа гипертекстовых систем. Однако исследование свойств индекса Винера для такого уточнения расстояния между вершинами до сих пор не было сделано. В настоящей статье рассматривается задача поиска верхней и нижней оценки индекса Винера для этого способа, а также исследование взаимосвязи с первым способом. В статье также рассматриваются экспериментальные результаты, которые позволяют провести оценку изменений индекса Винера по мере добавления дуг в граф. Действительно, произвольный слабо связный граф имеет индекс Винера, ограниченный верхней и нижней оценками и по мере добавления дуг, длины кратчайших путей между вершинами будут уменьшаться. Интерес вызывает характер изменения индекса с учетом двух способов к уточнению определения расстояния между вершинами в слабо связном графе.

1. Необходимые термины и определения

Введем необходимые определения. *Неориентированным графом* G назовем тройку, состоящую из множества вершин $V(G)$, ребер $E(G)$ и отношения, которое сопоставляет каждому ребру неупорядоченную пару вершин из множества $V(G)$. Чтобы подчеркнуть, что мы имеем дело с *ориентированным графом*, будем называть $E(G)$ множеством дуг и тогда отношение является функцией, которая сопоставляет дуге упорядоченную пару вершин из множества $V(G)$. В настоящей статье

рассматриваются графы, в которых множества $V(G)$ и $E(G)$ являются конечными, и в которых отсутствуют кратные ребра (дуги). Ориентированные графы могут содержать петли, т. е. дуги, которые соединяют вершину саму с собой. Мы ограничимся рассмотрением графов без петель.

Ориентацией графа G называется ориентированный граф D, получаемый из G выбором ориентации для каждого ребра $E(G)$.

Обозначим через $deg_G(v)$ *степень вершины* v в неориентированном графе G, которая равна количеству инцидентных v ребер. В ориентированном графе введем понятия *полустепени исхода* $deg^+_G(v)$ *и полустепени захода* $deg^-_G(v)$ как количество дуг, которые соответственно выходят и входят в v.

Маршрутом в графе назовем чередующуюся последовательность вершин и ребер (дуг, для ориентированных графов) $v_0, e_1, v_1, e_2, \ldots, v_{n-1}, e_n, v_n$, такую что $v_i \in V(G), e_i \in E(G)$, причем каждой e_i сопоставлена пара (v_{i-1}, v_i). Если все вершины маршрута попарно различны, то такой маршрут называется *простой цепью* (для неориентированных графов) или *путем* (для ориентированных). Маршрут называется *замкнутым* если $v_0 = v_n$. Будем называть *контуром* маршрут в ориентированном графе, в котором все вершины, за исключением первой и последней, различны.

Если в неориентированном графе любые две вершины соединены маршрутом, то такой граф называется связным. Для ориентированного графа D введем понятие *основного графа* G. Это граф, для которого $V(G) = V(D)$ и $(uv) \in E(G)$, если $(uv) \in E(D)$ или $(vu) \in E(D)$. Ориентированный граф называется *слабо связным*, если его основной граф является связным. Ориентированный граф называется *сильно связным*, если для любой упорядоченной пары вершин u, v существует маршрут из u в v. Очевидно, что из сильной связности графа следует его слабая связность.

Деревом называется связный неориентированный граф без циклов. *Ориентированным деревом* будем называть граф без контуров, у которого только одна вершина имеет полустепень захода, равную нулю, а остальные вершины вершины имеют полустепени захода, равные единице.

Расстоянием $d_G(u, v)$ между вершинами u, v графа G является длина кратчайшей простой цепи (пути для ориентированных графов) между этими вершинами. В том случае, если между вершинами в графе не существует простой цепи (пути), считается, что $d_G(u, v) = \infty$.

Индекс Винера (Wiener index) $W(G)$ для связного (неориентированного) графа G определяется как сумма расстояний между всеми неупорядоченными парами вершин графа:

$$W(G) = \sum_{\{u,v\} \subseteq V} d_G(u, v).$$

Как уже ранее говорилось, для ориентированных графов определение имеет смысл только для сильно связных графов. Обозначим через $d^0_G(u, v)$ расстояние (см., напр. [5]) которое устанавливает, что в случае отсутствия пути между вершинами u и v полагается, что $d^0_G(u, v) = 0$. Через $d^N_G(u, v)$ обозначим расстояние (см. [9]), которое устанавливает, что в этом случае $d^N_G(u, v) = |V(G)|$. Очевидно, что в произвольном графе на n вершинах расстояние между вершинами не может превышать $n - 1$. Далее будем обозначать как $W^0(G)$ и $W^N(G)$ индекс Винера, рассчитанный с использованием расстояний $d^0_G(u, v)$ и $d^N_G(u, v)$ соответственно.

Выделим несколько специальных неориентированных графов. Назовем *звездой* S_n дерево, в котором одна вершина смежна с остальными, а *путем* P_n дерево, в котором две вершины имеют степень один, а остальные вершины степень, равную двум. Через C_n обозначим граф, представляющий *замкнутую простую цепь*. При рассмотрении ориентированных графов, основные графы которых являются звездой или путем будем использовать эти же термины с уточнением касательно ориентации дуг. Так, через $\vec{C}_n$ обозначим ориентацию графа C_n, в которой дуги ориентированы по кругу в одну сторону. Через $\vec{P}_n$ обозначим ориентацию графа P_n, в которой дуги ориентированы таким образом, что полустепень исхода каждой вершины равна максимум единице.

2. Свойства индексов $W^0(G)$ и $W^N(G)$ для слабо связных ориентированных графов

Рассмотрим как изменяются значения индексов когда мы добавляем или удаляем дугу в ориентированном графе.

Лемма 1. *Добавление дуги в ориентированный граф G уменьшает значение индекса $W^N(G)$.*

Доказательство. Добавление дуги между вершинами u, v в граф G на n вершинах уменьшает значение $d_G^N(u, v)$ до единицы. Возможно также уменьшаются расстояния между другими вершинами графа, если добавленная дуга позволяет сформировать более короткий путь между ними. При этом не существует двух вершин, между которыми расстояние увеличивается. □

Если мы добавим дугу между вершинами u, v в слабо связный ориентированный граф G, то по поводу изменения индекса $W^0(G)$ ничего определенного сказать нельзя. С одной стороны, добавление дуги может сформировать более короткий путь между вершинами. С другой стороны, у нас образуются новые пути, которые увеличивают значение расстояния d_G^0 до какого-то ненулевого значения, внося вклад в расчет индекса.

Лемма 2. *Удаление дуги из ориентированного графа G увеличивает значение индекса $W^N(G)$.*

Доказательство. Удаление дуги между вершинами u, v из графа G увеличивает значение расстояния $d_G^N(u, v)$ между ними. Между другими вершинами графа расстояние уменьшиться не может, поскольку удаление дуги не может сформировать более короткие пути между этими вершинами. □

Что касается индекса $W^0(G)$, то здесь опять же, нет никакой определенности, вырастет, или уменьшится его значение. Допустим, дуга e входила в кратчайший путь между какими-то вершинами u, v. Тогда после ее удаления расстояние $d_G^0(u, v)$ может увеличиться, если есть есть альтернативная, но более длинный путь между этими вершинами, а может уменьшиться до нуля, если альтернативного пути не существует.

Можно заметить, что для слабо связного ориентированного графа справедливо тождество

$$W^N(G) - W^0(G) = m|V(G)|,$$

где m — количество пар вершин, между которыми отсутствует путь. Это позволяет сделать вывод, что для сильно связных ориентированных графов значение индексов совпадает, а для слабо связных (для которых не выполняется условие сильной связности) ориентированных графов они различны.

3. Экстремальные оценки индексов $W^0(G)$ и $W^N(G)$

Интерес представляет задача поиска верхней и нижней оценки индекса Винера в зависимости от количества вершин графа. Для деревьев известно [2], что если n — это количество вершин в дереве, то индекс Винера ограничен следующим образом:

$$W(S_n) \leqslant W(T_n) \leqslant W(P_n).$$

В [10] показано, что для любого неориентированного графа G выполняется $W(K_n) \leqslant W(G) \leqslant W(P_n)$, где K_n — полный граф.

Рассмотрим эту задачу для индексов $W^0(G)$ и $W^N(G)$. В работе [4] приводится нижняя оценка индекса Винера для сильно связного ориентированного графа:

$$W(G) \geqslant 2n(n-1) - m,$$

где n — это количество вершин, а m — количество дуг графа. С очевидностью, для сильно связных графов это же является нижней оценкой для индексов $W^0(G)$ и $W^N(G)$. Для нижней оценки $W^0(G)$ для слабо связного ориентированного графа можно использовать следующую гипотезу из [5].

Гипотеза 1. *Для каждого (неориентированного) графа G минимальная оценка ориентации G' $W(G')$ достигается для некоторой ациклической ориентации G'.*

В связи с этим можно доказать следующее утверждение.

Утверждение 1. *Минимальное значение индекса $W^0(G)$ для слабо связного ориентированного графа на n вершинах, достигается для звезд, у которых все дуги либо исходят из одной вершины, либо заходят в одну вершину.*

Доказательство. Минимальное количество ребер в связном неориентированном графе на n вершинах равно $n-1$. Таким образом, в любой ориентации этого графа будет $n-1$ дуга. Каждая дуга вносит в итоговое значение индекса $W^0(G)$ как минимум единицу. Следовательно, минимальное значение индекса $W^0(G)$ достигается для ациклических графов, где больше нет никаких других путей, кроме как $n-1$ путь длины один. Это в точности звезды, в которых $n-1$ дуга исходит из одной вершины, либо звезды, где эти дуги заходят в одну вершину. □

Это позволяет установить оценку снизу для рассматриваемого класса графов:

$$n - 1 \leqslant W^0(G).$$

Сверху можно дать грубую оценку в виде

$$W^0(G) \leqslant n(n-1)^2.$$

Действительно, длина наибольшего пути между парой вершин равна $n-1$, а максимальное количество таких путей равно $n(n-1)$. Для более точной оценки необходимо доказать несколько вспомогательных утверждений.

Лемма 3. *Удаление из ориентированного дерева вершины, полустепень исхода которой равна нулю, позволяет получить новое ориентированное дерево.*

Доказательство. Рассмотрим ориентированное дерево T и вершину u, полустепень исхода которой равна нулю. После удаления этой вершины из T мы получим граф T'. Докажем, что он является ориентированным деревом.

Поскольку полустепень исхода вершины u равна нулю, то эта вершина не может быть промежуточной в каком-то пути между другими вершинами в T. Следовательно, все пути из T также представлены в T'. Исключением являются пути, где u является заключительной вершиной. Следовательно, удаление вершины u не влияет на полустепени захода других вершин, которые равны единице. Единственное что изменяется, это полустепень исхода одной вершины $w \to u$. Следовательно, граф T' является ориентированным деревом. □

Лемма 4. *Среди всех ориентированных деревьев на n вершинах, индекс $W^0(T)$ максимален для дерева $\vec{P}_n$.*

Доказательство. Лемма 3 позволяет нам использовать индукцию по количеству вершин для ориентированных деревьев.

Базис индукции. Для $n = 1$ единственным ориентированным деревом является дерево из одной изолированной вершины.

Индуктивный переход. Пусть $n > 1$. Рассмотрим вершину u в дереве T, полустепень исхода которой равна нулю. Обозначим через $T - u$ новое ориентированное дерево без вершины u. Тогда

$$W^0(T) = W^0(T-u) + \sum_{v \in V(T)} \left(d^0(v,u) + d^0(u,v)\right).$$

По предположению индукции $W^0(T-u)$ максимален только для деревьев, у которых полустепень исхода каждой вершины равна максимум единице. Так как полустепень исхода вершины u равна нулю, по определению расстояния для всех v: $d^0(u,v)=0$.

Рассмотрим список расстояний от корня дерева T до u. В деревьях, где полустепень исхода каждой вершины равна максимум единице (и полустепень захода каждой вершины равна единице, за исключением корня дерева), это список $1,2,\ldots,n-1$, все эти числа различны, причем расстояние от корня до u равно $n-1$ и эта вершина является единственной вершиной с полустепенью исхода, равной нулю. Если T это произвольное дерево, то возникают вершины, от которых до u нет маршрутов, что во-первых, уменьшает максимальное значение в рассматриваемом списке, а во вторых обнуляет некоторые элементы этого списка. Таким образом, сумма $\sum_{v\in V(T)} d^0(v,u)$ максимальна, когда T является искомым деревом. □

Формулировку следующего утверждения можно найти в [8], в настоящей статье мы его доказываем несколько более строго.

Утверждение 2. *Максимальное значение индекса $W^0(G)$ для слабо связных ориентированных графов на n вершинах достигается графа $\vec{C}_n$.*

Доказательство. Пусть у нас есть произвольный ориентированный граф G на n вершинах. Рассмотрим произвольную вершину u в G и подграф G', состоящий из кратчайших путей из u до всех остальных вершин в G. В этом подграфе вершина u имеет нулевую полустепень захода, полустепень захода остальных вершин не является нулевой. Допустим в G' есть вершина v, полустепень захода которой больше единицы. Это означает, что из u в v ведут несколько кратчайших путей одинаковой длины. Удалим входящие в v дуги, кроме одной. Получим, что v имеет полустепень захода, равную единице. Проделаем эту операцию для всех вершин, у которых полустепень захода больше единицы.

В результате получим ориентированное дерево $T(u)$, которое задает кратчайшие пути из вершины u до других достижимых от нее вершин графа.

Можно заметить, что

$$W^0(G)=\sum_{u\in V(G)}\sum_{v\in V(G)} d^0(u,v)=\sum_{u\in V(G)}\sum_{v\in T(u)} d^0(u,v).$$

Таким образом, максимум достигается, если для каждого u дерево $T(u)$ во-первых, содержит n вершин, а во-вторых, по лемме 4 максимальная полустепень исхода каждой вершины равна единице. Графом, который одновременно удовлетворяет этим требованиям, является контур C_n. □

В этом случае оценка сверху составляет

$$W^0(G)\leqslant \sum_{u\in V(C_n)}\sum_{i=1}^{n-1} i=\sum_{u\in V(C_n)}\frac{n(n-1)}{2}=\frac{n^2(n-1)}{2}.$$

Минимум индекса $W^0(G)$ достигается для слабо связного, не являющего сильно связным, а максимум для сильно связного графа. Следующее утверждение отвечает на вопрос, для каких слабо связных (для которых не выполняется свойство сильной связности) графов достигается максимум $W^0(G)$.

Утверждение 3. *Максимальное значение индекса $W^0(G)$ достигается для слабо связного графа на n вершинах, не являющегося сильно связным, который представляет собой контур $\vec{C}_{n-1}$, куда либо заходит, либо откуда выходит дуга в оставшуюся n-ую вершину.*

Доказательство. Пусть нам дан слабо связный граф G_n на n вершинах. Поскольку этот граф не является сильно связным, то он состоит из сильно связных компонент, которые соединены дугами, причем эти дуги не должны принадлежать ни одному контуру. Рассмотрим наибольшую по размеру сильно связную компоненту G_m. С остальным графом эта компонента соединена дугами. Какие-то дуги ведут в G_m, какие-то из G_m. Если все дуги ведут в одном направлении, то больше ничего не делаем. Если же это не так, то для дуг, которые ведут из $u \in G_m$ в v добавим в граф дугу, которая ведет из v в u. Это увеличит индекс $W^0(G_n)$, поскольку в графе появляются новые пути, однако не происходит сокращения длин каких-то путей, так как из v в u до проведения дуги не существовало путей, иначе бы v входила в $V(G_m)$. После этой операции к G_m добавятся вершины. Будем пополнять G_m пока все дуги не будут вести в одном направлении в оставшуюся часть графа, которую мы будем обозначать G_{n-m}.

В том случае если между какими-то $u, v \in V(G_{n-m})$ не существует пути, то добавим дугу, ведущую из u в v. Это добавит пути в графе G, но не уменьшит какие-то из существовавших путей, поскольку для этого u и v должны лежать на каком-то из них. В результате проведенное действие увеличивает индекс $W^0(G_n)$. Будем продолжать такое добавление дуг пока это возможно.

В результате граф будет представлять собой две сильно связные компоненты, при этом дуги будут вести из одной компоненты в другую в одном направлении. Если таких дуг несколько, то мы можем удалить все, кроме одной. Это не разрушает пути в графе, поскольку эти дуги не влияют на пути внутри сильно связных компонент, а между ними остается путь через оставшуюся дугу. Это может увеличить какие-то из путей между сильно связными компонентами, что увеличивает индекс $W^0(G_n)$, который теперь состоит из двух сильно связных компонент, соединенных дугой. Обозначим их через G_k и G_{n-k}, и дуга пусть ведет из G_k в G_{n-k}. Тогда

$$W^0(G_n) = W^0(G_k) + W^0(G_{n-k}) + \sum_{\substack{u \in V(G_k) \\ v \in V(G_{n-k})}} d^0(u, v).$$

Можно заметить, что $W^0(G_k) \leqslant W^0(\vec{C}_k), W^0(G_{n-k}) \leqslant W^0(\vec{C}_{n-k})$ и для всех $u \in V(G_k), v \in V(G_{n-k})$:

$$d^0(u, v) = d^0(u, x) + 1 + d^0(y, v),$$

где x и y — это концы дуги, соединяющей компоненты связности. При этом набор расстояний $d^0(u, x)$ и $d^0(y, v)$ максимален в том случае, когда $G_k = \vec{C}_k$ и $G_{n-k} = \vec{C}_{n-k}$. Отсюда следует, что

$$\begin{aligned}
W^0(G_n) \leqslant & W^0(\vec{C}_k) + W^0(\vec{C}_{n-k}) + \sum_{j=0}^{k-1} \sum_{i=0}^{n-k-1} (j + 1 + i) = \\
= & \frac{k^2(k-1)}{2} + \frac{(n-k)^2(n-k-1)}{2} + \sum_{j=0}^{k-1} \left(j(n-k) + (n-k) + \frac{(n-k-1)(n-k)}{2} \right) = \\
= & \frac{k^2(k-1)}{2} + \frac{(n-k)^2(n-k-1)}{2} + \frac{(n-k)(k-1)k}{2} + (n-k)k + \frac{(n-k-1)(n-k)k}{2} = \\
= & \frac{k^2(k-1)}{2} + \frac{(n-k)^2(n-k-1)}{2} + \frac{(k-1+2+n-k-1)(n-k)k}{2} = \\
= & \frac{k^2(k-1)}{2} + \frac{(n-k)^2(n-k-1)}{2} + \frac{nk(n-k)}{2} = (n-1)k^2 - (n^2-n)k + \frac{n^3-n^2}{2} \to \max_{1 \leqslant k < n}.
\end{aligned}$$

Функция $f(k) = (n-1)k^2 - (n^2-n)k + \frac{n^3-n^2}{2}$ является параболой с ветвями вверх. Тогда $f'(k) = 2(n-1)k - n(n-1)$, приравняем к нулю и получим, что $k = \frac{n}{2}$. То есть для описанного класса графов минимум достигается в середине исследуемого промежутка, соответственно максимум достигается

на краях, т. е. когда одна из компонент связности является вершиной, и туда либо ведет дуга, либо оттуда исходит дуга. □

Перейдем теперь к рассмотрению оценок для индекса $W^N(G)$.

Утверждение 4. *Минимальное значение индекса $W^N(G)$ для слабо связных ориентированных графов на n вершинах достигается для графов, у которых все вершины попарно соединены дугами.*

Доказательство. По определению, для любой пары вершин u, v справедливо, что $1 \leqslant d^N(u,v) \leqslant n$. Следовательно, в графах, где все вершины попарно соединены дугами, достигается минимальное значение индекса $W^N(G)$. Максимальное количество дуг в таком графе равно $n(n-1)$, следовательно для любого графа $n(n-1) \leqslant W^N(G)$. □

Можно заметить, что минимальное значение индекса $W^N(G)$ достигается для слабо связного ориентированного графа, который является сильно связным. Если мы будем рассматривать графы, не обладающие свойством сильной связности, то справедливо следующее утверждение.

Утверждение 5. *Минимальное значение индекса $W^N(G)$ для слабо связных ориентированных графов на n вершинах, не обладающих свойством сильной связности, достигается для графов, где есть одна вершина, связанная одинаково ориентированными по отношению к ней дугами, инцидентными подграфу на $n-1$ вершине, которые попарно соединены дугами.*

Доказательство. Для того чтобы не выполнялось свойство сильной связности необходимо чтобы существовала пара вершин в графе, между которыми отсутствует путь. Следовательно граф состоит из множества сильно связных подграфов, которые связывают дуги, причем эти дуги не должны принадлежать ни одному контуру.

Пусть исходный граф G_n на n вершинах состоит из двух сильно связных компонент, между которыми есть произвольно направленные дуги. Пусть первая компонента G_k имеет размер k, тогда вторая компонента G_{n-k} имеет размер $n-k$. Тогда справедливо, что

$$W^N(G_n) = W^N(G_k) + W^N(G_{n-k}) + \sum_{\substack{u \in V(G_k) \\ v \in V(G_{n-k})}} \left(d^N(u,v) + d^N(v,u) \right).$$

Из утверждения 4 следует, что $W^N(G_k) \geqslant k(k-1)$, а $W^N(G_{n-k}) \geqslant (n-k)(n-k-1)$. Для любых $u \in V(G_k), v \in V(G_{n-k}$ справедливо, что $d^N(u,v)+d^N(v,u) \geqslant 1+n$, поскольку в какую-то сторону у нас путь отсутствует, и тогда расстояние между рассматриваемой парой вершин равно n (для определенности возьмем, что путь отсутствует из G_{n-k} в G_k), а в другую сторону он минимально может быть равен единице. Исходя из этого справедливо следующее:

$$W^N(G_n) \geqslant k(k-1) + (n-k)(n-k-1) + n(n-k)k + 1(n-k)k = (1-n)k^2 + (n^2-n)k + n^2 - n.$$

Это выражение минимально при $k = 1$, либо $k = n-1$, что в обоих случаях приводит нас к искомому графу.

В том случае, когда исходный граф G_n состоит из m сильно связных компонент, то можно заметить, что в совокупности $m-1$ компонента имеет значение индекса меньше, чем граф из утверждения 4, соответственно мы легко сводим этот случай к рассмотренному. □

Из леммы 2 следует, что для идентификации графов, для которых достигается максимум $W^N(G)$ нам необходимо рассматривать слабо связные графы, основной граф которых является деревом.

Если рассмотреть произвольные, не обязательно слабо связные графы на n вершинах, то с очевидностью максимальное значение индекса $W^N(G)$ достигается для графов, у которых нет дуг.

В этом случае для каждой вершины u расстояние до любой другой вершины v равно n и фактически набор расстояний от u состоит из $n-1$ элемента, равного n. В результате мы получаем, что индекс $W^N(G)$ для таких графов равен $n^2(n-1)$.

Утверждение 6. *Максимальное значение индекса $W^N(G)$ для слабо связных ориентированных графов на n вершинах достигается для звезд, у которых все дуги либо исходят из одной вершины, либо заходят в одну вершину.*

Доказательство. Согласно лемме 1 добавление дуги в граф G уменьшает индекс $W^N(G)$, причем если проводится дуга от вершины u к вершине v, то $d^N(u,v)$ становится равным единице и, возможно, уменьшаются некоторые другие расстояния между вершинами графа.

Рассмотрим ориентированный граф G, в котором отсутствуют дуги. Соответственно, для максимизации индекса $W^N(G)$ слабо связного графа нам необходимо добавить в граф G $n-1$ дугу таким образом, чтобы только одно расстояние становилось равным единице, а другие расстояния не изменялись. Выберем произвольную вершину u в графе и начнем добавлять исходящие дуги из этой вершины до остальных $n-1$ вершины. Пусть на очередном шаге у нас появилась дуга из u в v. После этого расстояние $d^N(u,v)$ стало равно единице, однако расстояния до других вершин из вершины v не изменились, поскольку из v по построению не будет исходить ни одной дуги.

Построение слабо связного графа было проведено таким образом, чтобы на каждом шаге минимально уменьшать значение индекса $W^N(G)$. Следовательно, после $n-1$ шага мы получим граф, имеющий максимальное значение индекса $W^N(G)$. Для таких графов $W^N(G) = n(n-1)^2 + (n-1)$.

Аналогичные рассуждения можно провести при построении графа, где все дуги заходят в одну вершину. □

Итак, для произвольного слабо связного графа G на n вершинах выполняется

$$n(n-1) \leqslant W^N(G) \leqslant n(n-1)^2 + (n-1).$$

Из вышесказанного можно сделать вывод, что индекс $W^N(G)$ характеризует слабо связные графы несколько отличным от индекса $W^0(G)$ образом.

С одной стороны мы наблюдаем двойственность индексов: для одного и того же класса графов на n вершинах индекс $W^0(G)$ является минимальным, в то время как индекс $W^N(G)$ является максимальным. Следующей интересной точкой наблюдения являются ориентированные по кругу контуры $\vec{C}_n$. Это минимальные по количеству дуг сильно связные графы, и здесь индекс $W^0(G)$ достигает максимума. Для сильно связных графов индексы $W^0(G)$ и $W^N(G)$ совпадают. Если добавлять дуги в сильно связный граф, то по лемме 1 индекс $W^N(G)$ будет уменьшаться, а вместе с ним и индекс $W^0(G)$. В результате мы приходим к классу графов, где все вершины попарно соединены дугами, для которого индекс $W^N(G)$ принимает глобальный минимум, а индекс $W^0(G)$ — локальный.

4. Экспериментальные исследования изменения индексов $W^0(G)$ и $W^N(G)$

Мы провели экспериментальные исследования для демонстрации как изменяются индексы $W^0(G)$ и $W^N(G)$ по мере случайного добавления ребер в слабо связный граф. Как следует из предыдущего раздела, эти индексы различны для слабо связных графов, которые не являются сильно связными. Поэтому интересным является то, как будет изменяться первый или второй индекс по мере добавления дуг в граф. Общая схема экспериментов заключается в следующем:

1. Случайным образом генерируется слабо связный граф.
2. В этот граф случайно добавляются дуги до тех пор, пока граф не станет сильно связным.

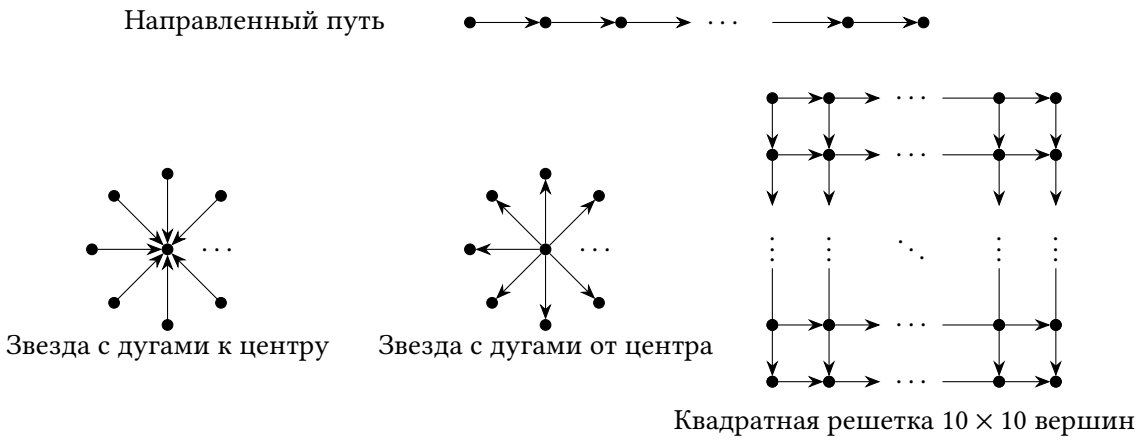

Fig. 1. Starting graphs with fixed structure

Рис. 1. Фиксированные стартовые графы

Генерирование случайных графов является сложной задачей (см. например [11]), требующей определения класса графов, которые будут генерироваться, а также процедуры, которая позволяет возвращать граф из этого класса с заданным распределением вероятностей. Мы отклоняемся от этой схемы и генерируем граф следующим образом:

1. Рассмотрим графы с количеством вершин $n = 100$. В начале в графе отсутствуют дуги.
2. Сгенерируем список, состоящий из всех A_n^2 дуг в лексикографическом порядке.
3. При помощи алгоритма [12, 13] получаем случайную перестановку списка дуг. Каждая перестановка возвращается равновероятным образом.
4. Добавляем в граф из получившейся перестановки дуги по порядку до тех пор, пока он не станет слабо связным. Назовем этот граф стартовым, поскольку с него начинается расчет индексов $W^0(G)$ и $W^N(G)$.
5. Продолжаем процесс добавления дуг до тех пор, пока не получим сильно связный граф. Получающиеся в рамках этого процесса графы будем называть промежуточными.

При проведении эксперимента мы не проверяем изоморфизм получившихся стартовых графов, поэтому вполне возможно, что какие-то из проанализированных графов являются изоморфными. Стоит также отметить, что из-за большого количества сгенерированных графов слишком трудоемко хранить матрицы смежности каждого графа для последующей проверки уникальности или изоморфизма. Вместо этого каждый граф идентифицируется по хэшу SHA-512 от строкового представления матрицы смежности графа.

В настоящей работе мы рассматриваем некоторые классы стартовых графов:

- случайный слабо связный граф представляет результат работы представленного алгоритма в чистом виде;
- направленный путь $\vec{P}_n$, а также рассматриваемые виды звезд и квадратная решетка изображены на рис. 1 и не предполагают случайного формирования стартового графа. Выбор в качестве одного из стартовых графов квадратной решетки обусловлен статьей [14], где рассматривается задача поиска максимальной ориентации для неориентированной квадратной решетки;
- случайная звезда получается путем случайной ориентации дуг звезды;
- случайная ориентация дерева представляет собой результат работы представленного алгоритма, с учетом модификации шага 4, которая при рассмотрении очередной дуги не добавляет в граф те из них, которые приводят к возникновению циклов в основном графе;
- при генерации случайной ориентации графа P_n мы добавляем дугу $(v_i v_{i+1})$ или $(v_{i+1} v_i)$ при условии отсутствия дуги, следующей в обратном направлении;

Table 1. Summary data on the studied graphs within the framework of the conducted experiments

Таблица 1. Сводные данные по исследованным графам в рамках проведенных экспериментов

Стартовый граф	Кол-во уникальных стартовых графов	Кол-во повторных стартовых графов	Кол-во исследованных слабо связных графов	Минимальное кол-во сгенерированных промежуточных графов для стартового графа	Среднее кол-во сгенерированных промежуточных графов для стартового графа	Максимальное кол-во сгенерированных промежуточных графов для стартового графа
Направленный путь	1	11 199	1 628 790	2	150	969
Случайная ориентация графа P_n	11 100	0	4 884 260	25	438	1 375
Случайная ориентация графа C_n	11 100	0	4 903 134	4	440	1 155
Звезда с дугами к центру	1	11 099	5 542 988	6	497	1 389
Звезда с дугами от центра	1	11 099	5 513 473	7	495	1 506
Случайная звезда	11 100	0	5 524 721	12	493	1 330
Случайная ориентация дерева	11 100	0	5 227 851	11	469	1 309
Квадратная решетка	1	11 149	1 641 593	1	153	1 203
Случайный слабо связный граф	11 150	0	3 495 362	1	312	1 226

- случайная ориентация графа C_n получается по тому же принципу, что и в предыдущем пункте, с учетом необходимости выбрать ориентацию дуги между первой и последней вершинами.

Статистика по исследованным графам приведена в таблице 1, а также на рис. 2. Мы видим, что распределение количества графов, которые необходимо построить до момента получения связного графа зависит от стартового графа.

На рис. 3 показаны результаты изменения индекса Винера по мере добавления дуг к стартовому графу в рамках проведенных экспериментов. На изображениях синим изображен индекс $W^0(G)$, а красным — индекс $W^N(G)$. На графиках изображены данные по всем экспериментам, чем более интенсивным цветом окрашена область на графике, тем в большем количестве экспериментов достигались значения из этой области. Для исследуемых графов из $n = 100$ вершин нижняя и верхняя границы индекса $W^0(G)$ равны 99 и 495 000 соответственно и показаны синим цветом. Верхняя граница индекса $W^0(G)$ для слабо связных графов, не являющихся сильно связными, равна 490 099 (она близка к верхней границе для произвольных слабо связных графов и на графиках не различима). Для индекса $W^N(G)$ нижняя и верхняя границы равны 9 900 и 980 199 соответственно и показаны красным цветом. Нижняя граница индекса $W^N(G)$ для слабо связных графов, не являющихся сильно связными, равна 19 701 и отмечена линией из точек.

Приведенные графики позволяют сделать следующие выводы. Изменение индексов $W^0(G)$, и $W^N(G)$ по мере добавления дуг в граф не является полиномиальным и зависит от стартового графа. В целом на графиках видно что сначала происходит медленное убывание индекса $W^N(G)$, за которым следует резкое снижение значения этого индекса, а также резкий рост индекса $W^0(G)$, за которым следует его медленное убывание.

Есть стартовые графы, где изменения происходят похожим образом, например, случайная ориентация графа P_n, случайная ориентация графа C_n. Не слишком отличается характер изменений при рассмотрении случайной ориентации дерева (небольшие различия заметны в самом начале эксперимента, начальное значение индекса $W^0(G)$ несколько выше, чем у предыдущих графов). Действительно, эти графы либо принадлежат одному классу рассматриваемых графов, как ори-

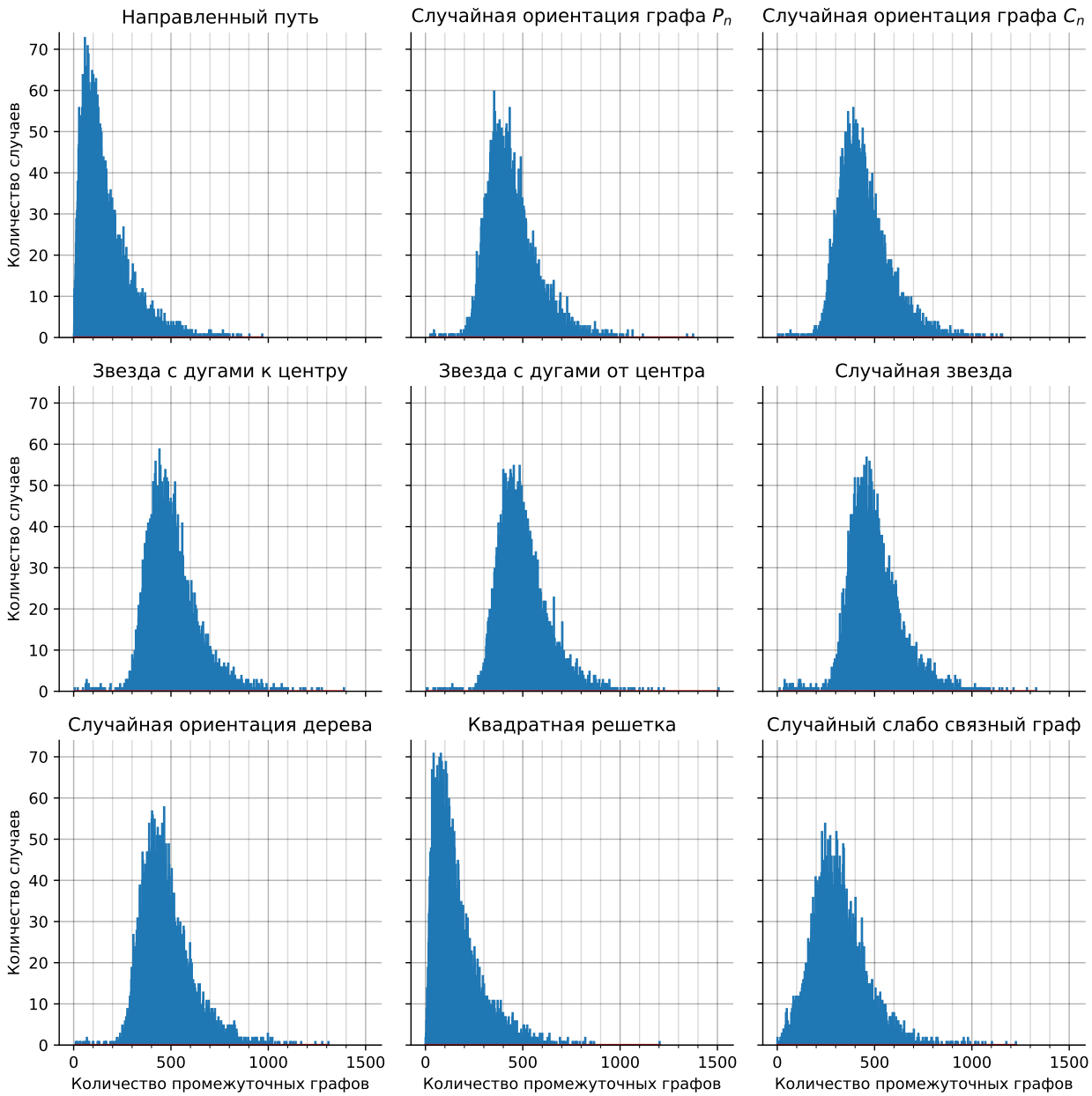

Fig. 2. Distributions of the number of constructed intermediate graphs from the starting graph to the strongly connected graph

Рис. 2. Распределения количества построенных промежуточных графов от стартового до сильно связного графа

ентация P_n и случайная ориентация дерева, либо не слишком сильно отличаются друг от друга по структуре, как в случае ориентации графов P_n и C_n.

Похожи также изменения при добавлении дуг к тем звездам, которые в нашем эксперименте имеют фиксированную структуру. В том случае, если ориентация дуг графа выбирается случайным образом, это тоже может вести к изменениям. Это ярко видно на примере, когда ориентация дуг звезды выбирается случайным образом. Аналогичные выводы мы можем сделать, когда происходит переход от фиксированной ориентации графа P_n к случайной.

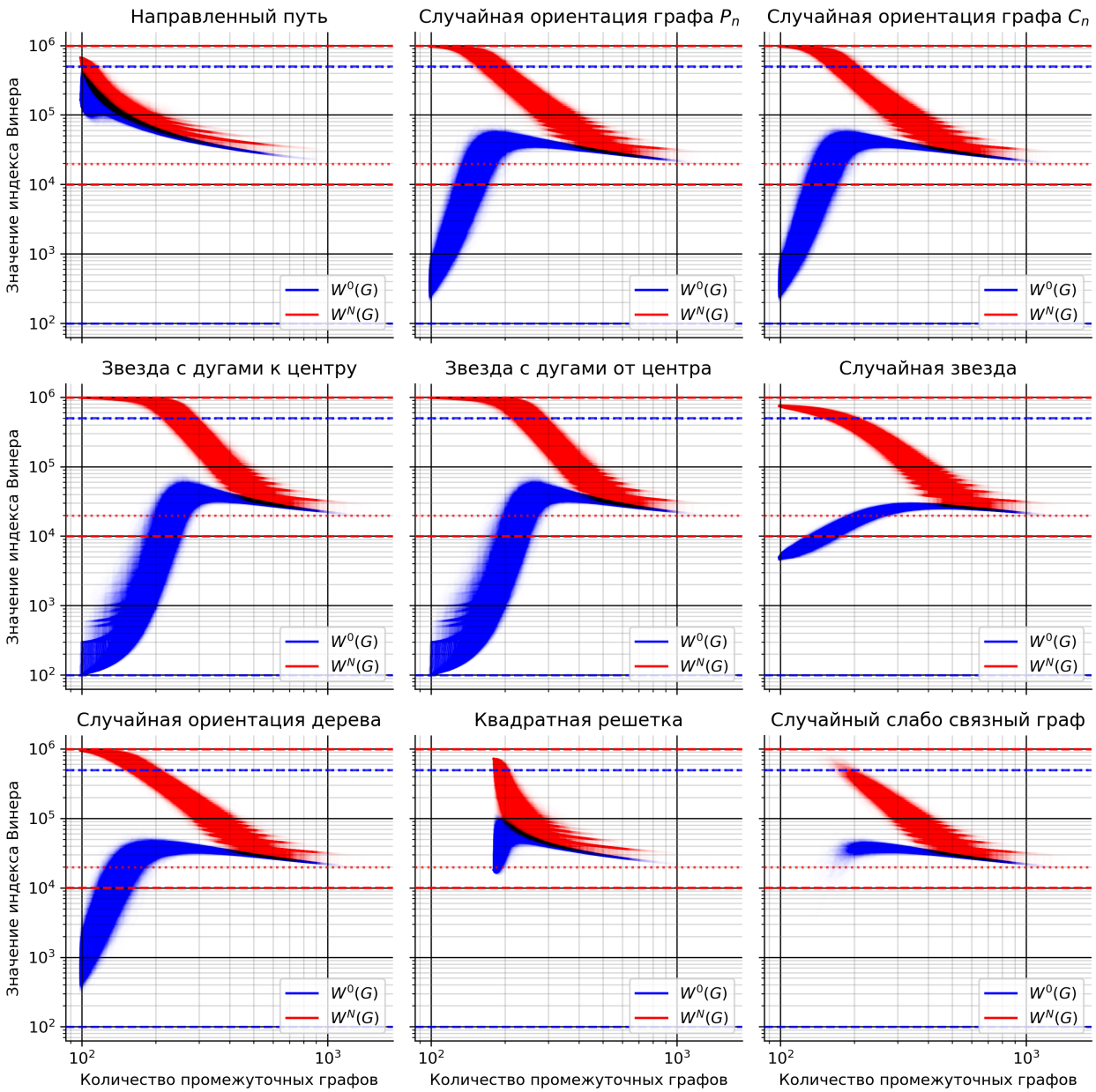

Fig. 3. Changes of the indicies $W^0(G)$ and $W^N(G)$ for the considered weakly connected directed graphs

Рис. 3. Изменения индексов $W^0(G)$ и $W^N(G)$ для рассматриваемых слабо ориентированных графов

На всех правых частях графиков можно видеть «полосы» в визуализации изменения индекса $W^N(G)$. Это означает, что при развитии процесса добавления дуг мы приходим к разным подклассам графов с похожими значениями индекса $W^N(G)$, в которых при добавлении дуги лишь сокращаются длины существующих путей, однако добавление нескольких дуг делает граф сильно связным.

Далее можно видеть, что в своей эволюции некоторые графы с точки зрения индексов $W^0(G)$ и $W^N(G)$ становятся слабо отличимыми от случайного слабо связного графа. Правые части многих

графиков похожи на то как развивается процесс, если стартовый граф является случайным слабо связным графом.

Некоторые стартовые графы имеют свои особенности. Это наблюдается и в форме графиков для таких фиксированных стартовых графов как направленный путь и квадратная решетка. Некоторые особенности видны и в случае фиксированных звезд. В самом начале изменения индекса $W^0(G)$ мы видим несколько скачкообразных увеличений индекса, каждое из которых в дальнейшем постепенно увеличивается.

Заключение

В работе рассматривается индекс Винера для слабо ориентированных графов с уточнением расстояния между вершинами из [9], где $d(u, v) = |V(G)|$ при отсутствии пути из u в v. Показано, что такое уточнение наделяет индекс Винера другими свойствами, отличными от классического $d(u, v) = 0$. В статье приводится доказательство того, для каких графов индекс $W^N(G)$ принимает свои наибольшее и наименьшее значения. Интересным является тот факт, что индекс $W^N(G)$, в отличие от индекса $W^0(G)$, является монотонным.

Приводятся также результаты экспериментального анализа изменения индексов $W^N(G)$ и $W^0(G)$ для слабо связных ориентированных графов при случайном добавлении дуг в граф. Для этого выбраны несколько фиксированных, а также случайных графов, с которых начинается процесс добавления дуг, который приводит в конце концов к сильно связному графу, где выполняется $W^0(G) = W^N(G)$. В результате экспериментов установлено, что имеют место отличия в характере изменения индексов. В целом видно, что индексы изменяются неполиномиально и для некоторых графов характер изменений похож и имеет определенные особенности. Из всего этого можно сделать вывод, что изучение индекса Винера $W^N(G)$ для исследования топологических свойств графовых моделей может иметь практическую ценность.

References

[1] H. Wiener, “Structural determination of paraffin boiling points”, *Journal of American Chemical Society*, vol. 69, no. 1, pp. 17–20, 1947.

[2] A. Dobrynin, R. Entringer, and I. Gutman, “Wiener index of trees: Theory and applications”, *Acta Applicandae Mathematicae*, vol. 66, pp. 211–249, 2001. DOI: 10.1023/a:1010767517079.

[3] F. Harary, “Status and contrastatus”, *Sociometry*, vol. 22, no. 1, pp. 23–43, 1959.

[4] C. Ng and H. Teh, “On finite graphs of diameter 2”, *Nanta Math*, vol. 1, pp. 72–75, 1966/67.

[5] M. Knor, R. Škrekovski, and A. Tepeh, “Some remarks on Wiener index of oriented graphs”, *Applied Mathematics and Computation*, vol. 273, pp. 631–636, 2016. DOI: 10.1016/j.amc.2015.10.033.

[6] M. Knor, R. Škrekovski, and A. Tepeh, “Orientations of graphs with maximum Wiener index”, *Discrete Applied Mathematics*, vol. 211, pp. 121–129, 2016. DOI: 10.1016/j.dam.2016.04.015.

[7] P. Dankelmann, “On the Wiener index of orientations of graphs”, *Discrete Applied Mathematics*, vol. 336, pp. 125–131, 2023. DOI: 10.1016/j.dam.2023.04.004.

[8] M. Knor, R. Škrekovski, and T. A., “Digraphs with large maximum Wiener index”, *Applied Mathematics and Computation*, vol. 284, pp. 260–267, 2016. DOI: 10.1016/j.amc.2016.03.007.

[9] R. Botafogo, E. Rivlin, and B. Shneiderman, “Structural analysis of hypertexts: Identifying hierarchies and useful metrics”, *ACM Transactions on Information Systems*, vol. 10, no. 2, pp. 142–180, 1992. DOI: 10.1145/146802.146826.

[10] D. West, *Introduction to Graph Theory*, Second Edition. Pearson Education, 2001.

[11] C. Greenhill, "Generating graphs randomly", in (London Mathematical Society Lecture Note Series), K. Dabrowski, M. Gadouleau, N. Georgiou, M. Johnson, G. Mertzios, and D. Paulusma, Eds., London Mathematical Society Lecture Note Series. Cambridge University Press, 2021, pp. 133–186.

[12] R. Durstenfeld, "Algorithm 235: Random permutation", *Communications of ACM*, vol. 7, no. 7, p. 420, 1964. DOI: 10.1145/364520.364540.

[13] D. Knuth, *The Art of Computer Programming: Seminumerical Algorithms*, 3rd ed. Addison-Wesley, 1998, vol. 2.

[14] M. Knor and R. Skrekovski, "On maximum Wiener index of directed grids", *The Art of Discrete and Applied Mathematics*, vol. 6, no. 3, #P3.02, 2023. DOI: 10.26493/2590-9770.1526.2b3.

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Dominant Sets with Neighborhood for Trees

M. A. Iordanski[1,2]

DOI: 10.18255/1818-1015-2025-1-32-41

[1]Minin Nizhny Novgorod State Pedagogical University, Nizhny Novgorod, Russia
[2]Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod, Russia

MSC2020: 05C69
Research article
Full text in Russian

Received December 18, 2024
Revised February 2, 2025
Accepted February 12, 2025

The subset $V' \subset V(G)$ forms a dominant set of vertices of the graph G with a neighborhood ε if for any vertex $v \in V \setminus V'$ there is a vertex $u \in V'$ such that the length of the shortest chain connecting these vertices $d(v,u) \leqslant \varepsilon$; $\delta_\varepsilon(G)$ is the number of vertices in the minimal ε-dominating set; $\delta_\varepsilon(G) = 1$ for $r(G) \leqslant \varepsilon \leqslant d(G)$; for $\varepsilon < r(G)$ the numbers $\delta_\varepsilon(G) > 1$, but the calculation of $\delta_1(G) = \delta(G)$ is an NP-complete problem. The paper considers class of trees t_d^ρ of diameter d whose degrees of all internal vertices are equal to ρ. Constructive descriptions of trees $t \in t_d^\rho$ are given. Procedures for calculating the values $\delta_\varepsilon(t)$ in the range $1 \leqslant \varepsilon < r(t)$ have been developed. Asymptotic estimates for $\delta_\varepsilon(t)$ and their share of the total number of vertices $t \in t_d^\rho$ are set at $d \to \infty$. Computational examples are given.

Keywords: trees; diameter; radius; dominating set with neighborhood; dominance number; gluing and cloning operations

INFORMATION ABOUT THE AUTHORS

Iordanski, Mikhail A. (corresponding author) | ORCID iD: 0000-0001-6625-1572. E-mail: iordanski@mail.ru
Dr. Sc., Professor

For citation: M. A. Iordanski, "Dominant sets with neighborhood for trees", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 32–41, 2025. DOI: 10.18255/1818-1015-2025-1-32-41.

© Iordanski M. A., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

DISCRETE MATHEMATICS IN RELATION TO COMPUTER SCIENCE

Доминирующие множества с окрестностью для деревьев

М. А. Иорданский[1,2]

DOI: 10.18255/1818-1015-2025-1-32-41

[1]Нижегородский государственный педагогический университет им. К. Минина, Нижний Новгород, Россия
[2]Нижегородский государственный университет им. Н.И. Лобачевского, Нижний Новгород, Россия

УДК 519.17
Научная статья
Полный текст на русском языке

Получена 18 декабря 2024 г.
После доработки 2 февраля 2025 г.
Принята к публикации 12 февраля 2025 г.

Подмножество $V' \subset V(G)$ образует ε-доминирующее множество графа G, если для любой вершины $v \in V \backslash V'$ найдется вершина $u \in V'$ такая, что длина кратчайшей цепи, соединяющей эти вершины $d(v,u) \leqslant \varepsilon$; $\delta_\varepsilon(G)$ — число вершин в минимальном ε-доминирующем множестве; $\delta_\varepsilon(G) = 1$ при $r(G) \leqslant \varepsilon \leqslant d(G)$; для $\varepsilon < r(G)$ числа $\delta_\varepsilon(G) > 1$, вычисление $\delta_1(G) = \delta(G)$ является NP-полной задачей. В работе рассматривается класс деревьев t_d^ρ диаметра d, степени внутренних вершин которых равны ρ. Приводятся конструктивные описания деревьев $t \in t_d^\rho$. Разработаны процедуры вычисления значений $\delta_\varepsilon(t)$ в диапазоне $1 \leqslant \varepsilon < r(t)$. Установлены асимптотические оценки для $\delta_\varepsilon(t)$ и их доли от общего числа вершин деревьев $t \in t_d^\rho$ при $d \to \infty$. Приводятся вычислительные примеры.

Ключевые слова: деревья; диаметр; радиус; доминирующее множество с окрестностью; число доминирования; операции склейки и клонирования

ИНФОРМАЦИЯ ОБ АВТОРАХ

Иорданский, Михаил Анатольевич (автор для корреспонденции) | ORCID iD: 0000-0001-6625-1572. E-mail: iordanski@mail.ru
Доктор физ.-мат. наук, профессор

Для цитирования: M. A. Iordanski, “Dominant sets with neighborhood for trees”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 32–41, 2025. DOI: 10.18255/1818-1015-2025-1-32-41.

© Иорданский М. А., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

Рассматриваются неориентированные связные графы без петель и кратных ребер. Используются следующие обозначения и определения: $V(G)$ — множество вершин графа $G(V, E)$; $G' = G(V')$ — подграф графа G, порожденный подмножеством вершин $V' \subset V(G)$; $d(v, u)$ — длина кратчайшей цепи, соединяющей вершины $v, u \in V(G)$; $\max_{u \in V(G)} d(v, u) = e(v)$ — *эксцентриситет* вершины v в графе G; $\max_{v \in V(G)} e(v) = d(G)$ — *диаметр* графа G; $\min_{v \in V(G)} e(v) = r(G)$ — *радиус* графа G, если $e(v) = r(G)$, то v *центральная* вершина графа G.

Используется конструктивный подход к представлению графов, предложенный автором, при котором графы рассматриваются как результаты некоторых процессов их построения [1]. К графам применяются теоретико-множественные операции объединения с пересечением, называемые операциями *склейки*. При выполнении операций склейки в графах-операндах G_1 и G_2 выделяются изоморфные подграфы $G'_1 \subseteq G_1$, $G'_2 \subseteq G_2$ и производится их отождествление.

Подграф результирующего графа, полученный путем отождествления подграфов G'_1 и G'_2, называется *подграфом склейки*. Если $G'_1 = G_1$ или (и) $G'_2 = G_2$, то операция склейки является *тривиальной* — её результирующий граф изоморфен хотя бы одному из графов-операндов.

Нетривиальные операции склейки, в которых один из графов-операндов изоморфен подграфу другого графа-операнда называются операциями *клонирования*. При их выполнении к исходному графу G добавляется подграф G'', изоморфный собственному связному подграфу $G' \subset G$. Подграф G'' называемый *клоном* подграфа G'. Вершины подграфа $G(V \backslash V')$, смежные с вершинами из V', называются *опорными*; V'_0 — множество всех опорных вершин. Опорные вершины соединяются с вершинами клона так, чтобы подграфы результирующего графа, порожденные множествами вершин $V' \cup V'_0$ и $V'' \cup V'_0$, были изоморфны. Таким образом, опорные вершины являются вершинами подграфа склейки.

Задавая ограничения на операции склейки, можно сохранять при масштабировании графов — увеличении числа вершин и ребер, различные свойства: планарность, эйлеровость, гамильтоновость и другие. Обзор этих работ можно найти в [2] а также в англоязычной версии [3]. В данной работе операции склейки используются для оценки метрических свойств масштабируемых графов.

Подмножество вершин $V' \subset V(G)$ графа G образует его *доминирующее множество с окрестностью* ε, если для любой вершины $v \in V \backslash V'$ найдется хотя бы одна вершина $u \in V'$ такая, что $d(v, u) \leqslant \varepsilon$. Будем называть такие множества *ε-доминирующими* множествами. Число вершин графа G в минимальном ε-доминирующем множестве обозначается через $\delta_\varepsilon(G)$ и называется *числом ε-доминирования*. При $\varepsilon = 1$ получаем обычное доминирующее множество, число вершин в котором называется *числом доминирования* $\delta(G)$ [4]. Учитывая определения радиуса и диаметра графа, $\delta_\varepsilon(G) = 1$ для $r(G) \leqslant \varepsilon \leqslant d(G)$. Если $\varepsilon < r(G)$, то $\delta_\varepsilon(G) > 1$. При этом задача установления чисел ε-доминирования становится достаточно сложной. Для произвольного графа G нахождение чисел доминирования $\delta(G) = \delta_1(G)$ является NP-полной задачей [5].

Доминирующие множества с окрестностью имеют многочисленные приложения, связанные с проектированием различных коммуникационных сетей. Имеется немало работ, в которых рассматриваются такие множества. Достаточно полный обзор этих публикаций содержится в [6].

В работе 1975 года [7] на основе совместного рассмотрения понятий расстояния и доминирования была сформулирована концепция доминирования на расстоянии. В последующих работах были получены многочисленные оценки для $\delta_\varepsilon(G)$ через различные параметры графов: число вершин, максимальные и минимальные степени вершин, диаметр, радиус и обхват графа. При получении оценок использовались такие факты: если H — остовный подграф графа G, то $\delta_\varepsilon(G) \leqslant \delta_\varepsilon(H)$, причем в G всегда найдется остовное дерево T, такое, что $\delta_\varepsilon(G) = \delta_\varepsilon(T)$ [8]. Верхние оценки $\delta_\varepsilon(G)$ для различных классов графов можно найти в [7, 9—11]. Нижние оценки рассматривались в [8, 12].

Для суперпозиций графов было установлено [8], что для прямого произведения графов справедливо соотношение $\delta_\varepsilon(G\times H)\geqslant\delta_\varepsilon(G)+\delta_\varepsilon(H)-1$. Для $G_{m,n}$ — решетки $m\times n$ (декартова произведения простых цепей с m и n вершинами) в [13] получены формулы для вычисления $\delta_\varepsilon(G_{m,n})$, $m\leqslant 3, n\geqslant m$, а при неограниченном росте m и n установлена асимптотика $\delta_\varepsilon(G_{m,n})/mn = 1/(2\varepsilon^2+2\varepsilon+1)$. В [14] получена верхняя оценка $\delta_\varepsilon(G_{m,n})$ при $n\geqslant m\geqslant 2$.

В настоящее время актуальны вопросы повышения производительности суперкомпьютерных вычислений, зависящей от структур используемых коммуникационных сетей. При масштабировании сетей предлагались различные технологические решения. Первое из них — множественность соединений между собой пар коммуникационных узлов, увеличивало объем данных, которые можно передавать по сети одновременно. На этом пути были разработаны соответствующие универсальные вычислительные сети, так называемые «толстые» деревья [15]. Второе направление связано с обеспечением топологической компактности коммуникационной сети за счет ограничения диаметра графа масштабируемой сети. Знание диаметра графа коммуникационной сети позволяет оценивать сверху время соединений произвольной пары узлов сети в ходе суперкомпьютерных вычислений [16, 17].

Операции склейки и, в частности, операции клонирования эффективны при решении задач масштабировании графов в обоих указанных выше направлениях. Технологичность процесса сборки графов с помощью операций клонирования следует из того, что при этом один из графов-операндов изоморфен собственному подграфу другого графа-операнда, то есть происходит тиражирование уже построенных подграфов [18, 19]. Примеры построения с помощью операций клонирования толстых деревьев а также масштабирования некоторых классов графов с ограничением диаметра приведены в [20, 21].

При переходе к операциям склейки по подграфам, вершины которых являются минимальными ε-доминирующими множествами увеличивается *коэффициент масштабирования* графов — отношение числа добавляемых вершин к числу вершин подграфа склейки, и тем самым уменьшается число необходимых склеек пар вершин при сохранении номенклатуры и количества используемых графов-операндов [21].

В работе продолжаются исследования начатые в [21]. Рассматривается класс деревьев t_d^ρ диаметра d, степени всех внутренних вершин которых равны ρ. Получены конструктивные описания деревьев $t\in t_d^\rho$, разработаны процедуры вычисления значений $\delta_\varepsilon(t)$ в диапазоне $1\leqslant\varepsilon<r(t)$. Найдена асимптотика для чисел доминирования $\delta(t)$ и их доли от общего числа вершин деревьев $t\in t_d^\rho$ при $d\to\infty$. Установлены асимптотические верхние оценки для $\delta_\varepsilon(t), \varepsilon\geqslant 2$ и их доли от общего числа вершин деревьев $t\in t_d^\rho$ при $d\to\infty$. На основе этих результатов получена асимптотика для коэффициента масштабирования графов при склейке с деревьями $t\in t_d^3$ по $\delta(t)$ вершинам и асимптотические нижние оценки для коэффициента масштабирования графов при склейке с деревьями $t\in t_d^\rho, \rho\geqslant 4$ по $\delta_\varepsilon(t)$ вершинам при $d\to\infty$. Приводятся вычислительные примеры.

1. Конструктивные описания деревьев класса t_d^ρ

Рассматривается класс деревьев, обладающих максимальным числом вершин при заданных диаметре $d\geqslant 4$ и максимальной степени внутренних вершин $\rho\geqslant 3$. Деревья с одной центральной вершиной (корнем), называются *равновесными*, если все вершины, равноудаленные от корня, имеют одинаковые степени. Длина цепей, соединяющих корень с листьями (висячими вершинами), определяет глубину h равновесного дерева. Для одноцентрального равновесного дерева диаметр и глубина связаны следующим образом $h=d/2$. При наличии двух смежных центральных вершин каждая из них рассматривается как корень равновесного поддерева глубины $h=\lfloor d/2\rfloor$, в котором все вершины, равноудаленные от корня поддерева, имеют одинаковые степени. Равновесное дерево, все внутренние вершины которого имеют одинаковые степени равные ρ, называется *однородным.*

Обозначим класс равновесных однородных деревьев диаметра d со степенью внутренних вершин ρ через t_d^ρ.

При построении произвольного дерева $t \in t_d^\rho$ в качестве исходного графа рассматривается цепь σ, содержащая $\lfloor d/2 \rfloor$ ребер. Занумеруем последовательно вершины цепи σ, начиная с вершины смежной с одной из её концевых вершин и кончая другой концевой вершиной, числами от 1 до $\lfloor d/2 \rfloor$. Эти вершины используются в качестве опорных при выполнении операций клонирования в процессе построения дерева $t \in t_d^\rho$.

При четном диаметре $d = 2k, k \geqslant 2$ операция клонирования для каждой опорной вершины повторяется до тех пор, пока ее степень не станет равной ρ. Для первой опорной вершиной в качестве клонируемого подграфа выступает смежная с ней концевая вершина цепи σ (равновесное поддерево глубины $h = 0$). При этом получаем равновесное поддерево глубины $h = 1$. Это поддерево является клонируемым подграфом для второй опорной вершины. При этом получаем равновесное поддерево глубины $h = 2$. Процесс построения равновесных поддеревьев увеличивающейся глубины продолжается до последней опорной вершины с номером $\lfloor d/2 \rfloor$, клонируемым подграфом для которой является равновесное поддерево глубины $h = k - 1$. В результате получаем равновесное однородное дерево степени ρ и глубины $h = k$, диаметр которого $d = 2k, k \geqslant 2$.

При нечетном диаметре $d = 2k + 1, k \geqslant 2$ строятся два поддерева глубины $h = k$ аналогично случаю четного диаметра. Отличие состоит лишь в том, что степени последних опорных вершин с номерами равными $\lfloor d/2 \rfloor$ доводятся с помощью операций клонирования до величины равной $\rho - 1$. После соединения ребром этих вершин получаем равновесное однородное дерево степени ρ, диаметр которого $d = 2k + 1, k \geqslant 2$. На рис. 1 приведены для примера деревья t_4^3 и t_5^3.

Замечание. В работе [21] показано, как деревья из класса t_d^ρ можно построить с помощью операций клонирования с одной опорной вершиной из цепи, содержащей одно ребро. Там же приведены оценки для минимального числа необходимых для этого операций клонирования.

2. Вычисление чисел ε-доминирования

Из структуры деревьев класса t_d^ρ следует, что их радиус $r = \lceil d/2 \rceil$. Поэтому значения чисел ε-доминирования $\delta_\varepsilon(t), t \in t_d^\rho$ будут рассматриваться в диапазоне $1 \leqslant \varepsilon < d/2$ для четных d и $1 \leqslant \varepsilon < \lceil d/2 \rceil$ — для нечетных d, полагая $\rho \geqslant 3$ и $d \geqslant 4$.

Учитывая способ построения деревьев класса t_d^ρ, вычислим вначале числа ε-доминирования для исходных графов — цепей σ, содержащих $\lfloor d/2 \rfloor$ ребер. Справедлива следующая лемма.

Лемма 1. *Числа ε-доминирования для цепей σ, содержащих $\lfloor d/2 \rfloor$ ребер, можно вычислить следующим образом:*

1. По заданным d и ε найти целочисленное $k \geqslant 1$, удовлетворяющее уравнению

$$\varepsilon + 1 + (k-1)(2\varepsilon + 1) + l = \lfloor d/2 \rfloor + 1, \tag{1}$$

при условии, что $l \in \overline{0, 2\varepsilon}$.

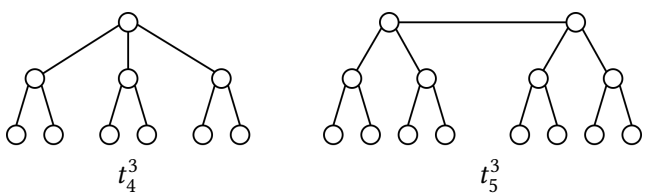

Fig. 1. Structure of t_4^3 and t_5^3 trees

Рис. 1. Структура деревьев t_4^3 и t_5^3

2. По найденному k число ε-доминирования определяется так:

$$\delta_\varepsilon(\sigma) = \begin{cases} k, & \text{если } l \in \overline{0,\varepsilon}, \\ k+1, & \text{если } l \in \overline{\varepsilon+1, 2\varepsilon}. \end{cases}$$

Доказательство. Для цепи σ, содержащей $\lfloor d/2 \rfloor$ ребер, ее ε-доминирующее множество будет, очевидно, минимально, если включить в него вершину, удаленную от одной из концевых вершин цепи σ на расстояние равное ε, и далее в направлении к другой концевой вершине цепи σ последовательно добавить все вершины, отстоящие друг от друга на расстояние равное $2\varepsilon+1$. При этом за последней (k-ой) такой вершиной до конца цепи могут располагаться ещё $l \in \overline{0, 2\varepsilon}$ вершин. Для вычисления k ищется решение уравнения (1), отражающего структуру вышеописаннного размещения вершин ε-доминирующего множества вдоль цепи σ при условии, что $l \in \overline{0, 2\varepsilon}$. Если $l \leqslant \varepsilon$, то $\delta_\varepsilon(\sigma) = k$. Если $l \in \overline{\varepsilon+1, 2\varepsilon}$, то в ε-доминирующее множество к k вершинам добавляется ещё одна вершина, принадлежащая подмножеству из l последних вершин цепи σ. □

Теорема 1. *Числа ε-доминирования $\delta_\varepsilon(t)$ для деревьев $t \in t_d^\rho$ можно вычислить по формулам:*

$$\delta_\varepsilon(t) = \begin{cases} \delta_\varepsilon^1(t) \text{ или } 2\delta_\varepsilon^2(t), & \text{если } l \in \overline{0,\varepsilon} \\ \delta_\varepsilon^1(t)+1 \text{или } 2\delta_\varepsilon^2(t)+1, & \text{если } l \in \overline{\varepsilon+1, 2\varepsilon}, \end{cases} \tag{2}$$

где

$$\delta_\varepsilon^1(t) = \frac{\rho(\rho-1)^{d/2-1}}{(\rho-1)^\varepsilon}\left(1 + \frac{1}{(\rho-1)^{2\varepsilon+1}} + \ldots + \frac{1}{(\rho-1)^{(k-1)(2\varepsilon+1)}}\right) \tag{3}$$

при четном d,

$$\delta_\varepsilon^2(t) = \frac{(\rho-1)^{\lfloor d/2 \rfloor}}{(\rho-1)^\varepsilon}\left(1 + \frac{1}{(\rho-1)^{2\varepsilon+1}} + \ldots + \frac{1}{(\rho-1)^{(k-1)(2\varepsilon+1)}}\right) \tag{4}$$

при нечетном d, в которых число слагаемых k определяется в соответствии с процедурой леммы 1.

Доказательство. Разобьем вершины дерева $t \in t_d^\rho$ на *слои* — подмножества вершин равноудаленных от листьев. Пусть d четно, тогда число вершин во всех слоях дерева $t \in t_d^\rho$, начиная от корня, равно

$$1, \rho, \rho(\rho-1), \rho(\rho-1)^2, \ldots, \rho(\rho-1)^{d/2-1}. \tag{5}$$

Пусть d нечетно. В этом случае дерево $t \in t_d^\rho$ строится из двух равновесных поддеревьев $t \in t_{d-1}^\rho$, корни которых соединены ребром. Число вершин в слоях этих поддеревьев, начиная от корней, равны

$$1, (\rho-1), (\rho-1)^2, \ldots, (\rho-1)^{\lfloor d/2 \rfloor}. \tag{6}$$

Так как при выполнении операций клонирования в ходе построения дерева $t \in t_d^\rho$ на основе исходной цепи σ, содержащей $\lfloor d/2 \rfloor$ ребер, сохраняется положение вершин из минимальных ε-доминирующих множеств на всех цепях, соединяющих листья с корнем дерева $t \in t_d^\rho$ при четном d (с корнем поддерева $t \in t_{d-1}^\rho$ при нечетном d), то в ε-доминирующее множество дерева $t \in t_d^\rho(t_{d-1}^\rho)$ будут входить некоторые из выделенных выше слоев вершин. Первый из этих слоёв, содержащий наибольшее число вершин, удален от листьев на расстояние равное ε. Все дальнейшие слои вершин ε-доминирующего множества, если $k \geqslant 2$, располагаются друг от друга на расстоянии равном $2\varepsilon+1$.

Число слоев k определяется в соответствии с процедурой леммы 1. Количество вершин в указанных слоях можно подсчитать по формуле (3) для четного d и по формуле (4) для нечетного d.

Если $l \in \overline{0,\varepsilon}$, то $\delta_\varepsilon(t) = \delta^1_\varepsilon(t)$ при четном d и $\delta_\varepsilon(t) = 2\delta^2_\varepsilon(t)$ при нечетном d. Если $l \in \overline{\varepsilon+1, 2\varepsilon}$, то необходимо добавить $(k+1)$-й слой, вершины которого получены в ходе построения дерева дерева $t \in t^\rho_d(t^\rho_{d-1})$ в результате клонирования некоторой вершины, принадлежащей подмножеству из l последних вершин цепи σ. Конкретный выбор этой вершины не влияет на значение числа ε-доминирования для цепи σ, однако, для сокращения числа вершин ε-доминирующего множества дерева $t \in t^\rho_d(t^\rho_{d-1})$ имеет смысл выбрать из множества l вершину, являющуюся концевой для цепи σ. Эта вершина не клонируется при построении дерева $t \in t^\rho_d(t^\rho_{d-1})$ и $(k+1)$-й слой будет состоять из одной этой вершины. При четном d этой вершиной является корень дерева $t \in t^\rho_d$. При этом $\delta_\varepsilon(t) = \delta^1_\varepsilon(t) + 1$. При нечетном d этой вершиной является корень поддерева $t \in t^\rho_{d-1}$. Так как корни этих поддеревьев смежны, то только один из них включается в минимальное ε-доминирующее множество, при этом $\delta_\varepsilon(t) = 2\delta^2_\varepsilon(t) + 1$. □

Пример 1. На рис. 2 фрагментарно представлены соответственно деревья t^3_{12} и t^3_{13} с указанием числа вершин в слоях. Общее число вершин в этих деревьях равно $n(t^3_{12}) = 190$, $n(t^3_{13}) = 254$. Для каждого дерева темным цветом помечены вершины из $\delta_1(t)$.

Результаты вычислений чисел ε-доминирования по формулам (2), (3) и (4) приведены в таблице 1.

Проиллюстрируем заполнение таблицы для $\varepsilon = 1$. Уравнение (1) для обоих деревьев примет следующий вид $3(k-1)+l = 5$. Его решение $k = 2$ и $l = 2$ при $l \in \overline{0,2}$. По формулам (3) и (4) получаем $\delta^1_1(t^3_{12}) = 3 * 2^4(1 + 1/2^3) = 54$, $\delta^2_1(t^3_{13}) = 2^5(1 + 1/2^3) = 36$.

Так как $l = \varepsilon + 1$, то по формуле (2) имеем $\delta_1(t^3_{12}) = \delta^1_1(t^3_{12}) + 1 = 55$ и $\delta_1(t^3_{13}) = 2\delta^2_1(t^3_{13}) + 1 = 73$.

3. Оценки для чисел ε-доминирования

Рассмотрим асимптотические оценки для чисел $\delta_\varepsilon(t) \leqslant \delta(t)$. Справедлива следующая лемма.

Лемма 2. *Для деревьев $t \in t^\rho_d$ при неограниченном росте d имеем:*

$$\delta(t) \sim \frac{\rho(\rho-1)}{(\rho-1)^3 - 1}(\rho-1)^{d/2} \tag{7}$$

на четной подпоследовательности d,

$$\delta(t) \sim \frac{2(\rho-1)}{(\rho-1)^3 - 1}(\rho-1)^{d/2} \tag{8}$$

на нечетной подпоследовательности d.

Доказательство. Так как k пропорционально d, то при неограниченном росте d формулы (3) и (4) превращаются при $\varepsilon = 1$ в суммы убывающих членов бесконечных геометрических прогрессий со знаменателем $q = 1/(\rho-1)^3$ и первыми членами соответственно $\frac{\rho(\rho-1)^{d/2-1}}{(\rho-1)}$ и $\frac{(\rho-1)^{\lfloor d/2 \rfloor}}{(\rho-1)}$. Пределы

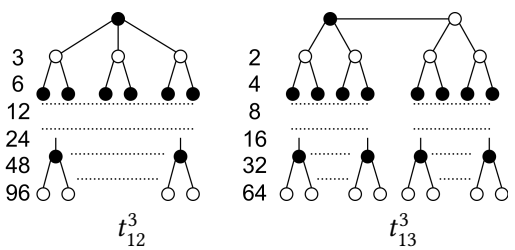

Fig. 2. Structure of trees t^3_{12} and t^3_{13}

Рис. 2. Структура деревьев t^3_{12} и t^3_{13}

Table 1. The values of the numbers of ε-dominance, $\varepsilon \in \overline{1,6}$ for trees t^3_{12} and t^3_{13}

Таблица 1. Значения чисел ε-доминирования, $\varepsilon \in \overline{1,6}$ для деревьев t^3_{12} и t^3_{13}

ε	$\delta_\varepsilon(t^3_{12})$	$\delta_\varepsilon(t^3_{13})$
1	55	73
2	25	33
3	12	16
4	6	8
5	3	4
6	1	2

этих сумм приведены в (7) и (8). В формулу (8) добавлен множитель 2, поскольку при нечетном d дерево $t \in t^\rho_d$ содержит два поддерева $t \in t^\rho_{d-1}$. □

Следствие 1. *Числа ε-доминирования $\delta_\varepsilon(t), t \in t^\rho_d$ асимптотически не превосходят значений (7) и (8) соответственно на четной и нечетной подпоследовательностях d.*

Обозначим через G_t класс графов, в каждом из которых можно выделить остовное дерево $t \in t^\rho_d$. Так как при добавлении ребер к любому графу числа ε-доминирования не могут увеличиваться, то отсюда получаем следующее.

Следствие 2. *Числа ε-доминирования $\delta_\varepsilon(G), G \in G_t$ асимптотически не превосходят значений (7) и (8) на четной и нечетной подпоследовательностях d.*

Теорема 2. *При неограниченном росте d числа доминирования $\delta(t), t \in t^\rho_d$ составляют от $n(t)$ — числа вершин дерева t, следующую долю:*

$$\frac{\delta(t)}{n(t)} \sim \frac{(\rho-1)(\rho-2)}{(\rho-1)^3-1}. \tag{9}$$

Доказательство. Вычислим $n(t)$ — число вершин в дереве $t \in t^\rho_d$. Число вершин в слоях (5), кроме первой единицы, соответствует членам геометрической прогрессии со знаменателем $(\rho-1)$ с начальным членом ρ и последним членом $\rho(\rho-1)^{\frac{d}{2}-1}$. Число вершин в слоях (6) соответствует членам геометрической прогрессии со знаменателем $(\rho-1)$ с начальным членом $\rho-1$ и последним членом $(\rho-1)^{\lfloor d/2 \rfloor}$. На основе сумм членов этих прогрессий получаем для четного и нечетного d соответственно

$$n(t) = 1 + \rho\frac{(\rho-1)^{d/2}-1}{\rho-2}, \tag{10}$$

$$n(t) = 2\frac{(\rho-1)^{\lceil d/2 \rceil}-1}{\rho-2}. \tag{11}$$

При неограниченном росте d имеем

$$n(t) \sim \frac{\rho}{\rho-2}(\rho-1)^{d/2}$$

для четного d,

$$n(t) \sim \frac{2}{\rho-2}(\rho-1)^{d/2}$$

для нечетного d.

Учитывая (7) и (8), отсюда получаем асимптотику (9). □

Следствие 3. *Для деревьев $t \in t_d^3$ при $d \to \infty$*

$$\frac{\delta(t)}{n(t)} \sim \frac{2}{7} \approx 0,2857.$$

Пример 2. Используя формулы (10) и (11), а также данные из примера 1, имеем для четного и нечетного d соответственно:

$$\frac{\delta(t_{12}^3)}{n(t_{12}^3)} = \frac{55}{190} \approx 0,2895; \quad \frac{\delta(t_{14}^3)}{n(t_{14}^3)} = \frac{109}{382} \approx 0,2853.$$

$$\frac{\delta(t_{13}^3)}{n(t_{13}^3)} = \frac{73}{254} \approx 0,2874; \quad \frac{\delta(t_{15}^3)}{n(t_{15}^3)} = \frac{145}{510} \approx 0,2843.$$

Поскольку значение выражения (9) уменьшается с ростом ρ, то учитывая следствие 3, получаем следующее.

Следствие 4. *Числа ε-доминирования $\delta_\varepsilon(t)$ составляют от $n(t)$ — общего числа вершин деревьев $t \in t_d^\rho$ долю, асимптотически не превосходящую $2/7$ при неограниченном росте d.*

Как указывалось во введении, использование операций склейки графов по вершинам их минимальных ε-доминирующих множеств позволяет увеличивать коэффициент масштабирования графов — отношение числа добавляемых вершин к числу вершин подграфа склейки. Если число вершин в подграфе склейки равно $\delta(t)$, а $(n(t) - \delta(t))$ — число добавляемых вершин, то $\frac{n(t)}{\delta(t)} - 1$ задает значение коэффициента масштабирования. При этом из следствия 3 получаем следующее.

Следствие 5. *Коэффициент масштабирования графов при склейке с деревьями $t \in t_d^3$ по $\delta(t)$ вершинам асимптотически равен $2,5$ при $d \to \infty$.*

Пример 3. Используя данные из примера 2, получаем следующие значения коэффициентов масштабирования графов при их склейке с деревьями $t \in t_d^3$ по $\delta(t)$ вершинам для четного и нечетного d соответственно:

$$\frac{n(t_{12}^3)}{\delta(t_{12}^3)} - 1 = \frac{190}{55} - 1 \approx 2,4545; \quad \frac{n(t_{14}^3)}{\delta(t_{14}^3)} - 1 = \frac{382}{109} - 1 \approx 2,5046.$$

$$\frac{n(t_{13}^3)}{\delta(t_{13}^3)} - 1 = \frac{254}{73} - 1 \approx 2,4794; \quad \frac{n(t_{15}^3)}{\delta(t_{15}^3)} - 1 = \frac{510}{145} - 1 \approx 2,5172.$$

Из следствий 3–5 получаем следующее.

Следствие 6. *Коэффициент масштабирования графов при склейке с деревьями $t \in t_d^\rho$ по $\delta_\varepsilon(t)$ вершинам асимптотически не меньше $2,5$ при $d \to \infty$.*

Заключение

Для чисел ε-доминирования $\delta_\varepsilon(G), 1 \leqslant \varepsilon < r(G)$ были разработаны процедуры вычисления значений $\delta_\varepsilon(t), 1 \leqslant \varepsilon < \lceil d/2 \rceil$ для деревьев $t \in t_d^\rho$.

Получена асимптотика для чисел доминирования $\delta(t), t \in t_d^\rho$ при неограниченном росте диаметра d, на основе которой установлены асимптотические верхние оценки чисел ε-доминирования для деревьев $t \in t_d^\rho$ и для графов, в которых можно выделить остовное дерево $t \in t_d^\rho$.

Найдены асимптотические значения и оценки для доли чисел ε-доминирования $\delta_\varepsilon(t)$ от $n(t)$ — числа вершин деревьев $t \in t_d^\rho$ и для коэффициентов масштабирования графов при склейке с деревьями $t \in t_d^\rho$ по $\delta_\varepsilon(t)$ вершинам при неограниченном росте d.

References

[1] M. A. Iordanski, "Constructive descriptions of graphs", *Discrete Analysis and Operations Research*, vol. 3, no. 4, pp. 35–63, 1996, in Russian.

[2] M. A. Iordanski, *Constructive graph theory and its applications*. Cyrillic, 2016, 172 pp., in Russian.

[3] M. A. Iordanski, *Constructive graph theory: Generation methods, structure and dynamic characterization of closed classes of graphs – a survey*, 2020. arXiv: 2011.10984 [math.CO].

[4] O. Ore, *Theory of graphs*. Nayka, Moscow, 1968, 352 pp., in Russian.

[5] M. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, 1979, 416 pp., in Russian.

[6] T. Haynes, S. Hedetniemi, and M. Henning, *Topics in Domination in Graphs. Developments in Mathematics*. Springer, 2020, vol. 64, 545 pp.

[7] A. Meir and J. Moon, "Relations between packing and covering number of a tree", *Pacific Journal of Mathematics*, vol. 61, no. 1, pp. 225–233, 1975.

[8] R. Davila, C. Fast, M. Henning, and F. Kenter, "Lower bounds on the distance domination number of a graph", *Contributions to Discrete Mathematics*, vol. 12, no. 2, pp. 11–21, 2017.

[9] F. Tian and J. Xu, "A note on distanse domination numbers of graphs", *The Australasian Journal of Combinatorics*, vol. 43, pp. 181–190, 2009.

[10] M. Henning and N. Lichiardopol, "Distance domination in graphs with given minimum and maximum degree", *Journal of Combinatorial Optimization*, vol. 34, pp. 545–553, 2017.

[11] P. Dankelmann and D. Erwin, "Distance domination and generalized eccentricity in graphs with given minimum degree", *Jouranl of Graph Theory*, vol. 94, no. 1, pp. 5–19, 2020. DOI: 10.1002/jgt.22503.

[12] J. Cyman, M. Lemanska, and J. Raczek, "Lower bound on the distance k-domination number of a tree", *Mathematica Slovaca*, vol. 56, no. 2, pp. 235–243, 2006.

[13] A. Klobucar, "On the k-dominating number of Cartesian products of two paths", *Mathematica Slovaca*, vol. 55, no. 2, pp. 141–154, 2005.

[14] E. Fata, S. Smith, and S. Sundaram, "Distributed dominating sets on grids", in *Proceedings of the American Control Conference*, 2013, pp. 211–216.

[15] C. Leiserson, "Fat-trees: Universal networks for hardware-efficient supercomputing", *IEEE Transactions on Computers*, vol. C-34, pp. 892–901, 1985.

[16] A. M. Rappoport, "Metric characteristics of communication network graphs", *Proceedings of the Institute of System Analysis of the Russian Academy Sciences*, vol. 14, pp. 141–147, 2005, in Russian.

[17] V. A. Melentiev and V. I. Shubin, "On scalability of computing systems with compact topology", *Theoretical Applied Science*, vol. 11, no. 43, pp. 164–169, 2016, in Russian.

[18] M. A. Iordanski, "Cloning of graphs", in *Proceedings of the XVIII International Conference on Problems of theoretical Cybernetics*, in Russian, 2017, pp. 108–110.

[19] M. A. Iordanski, "On the complexity of graph synthesis using cloning operations", in *Proceedings of the XIII International seminar on Discrete mathematics and its applications*, in Russian, 2019, pp. 220–223.

[20] M. A. Iordanski, "Cloning operations and the diameter of graphs", *Discrete Mathematics*, vol. 34, no. 2, pp. 26–31, 2022, in Russian.

[21] M. A. Iordanski, "Scaling graphs with constraint diameter", *Discrete Mathematics*, vol. 35, no. 4, pp. 46–57, 2023, in Russian.

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

ARTIFICIAL INTELLIGENCE

A Survey of Models for Automatic Assessment of Similarity of Student's Answer to the Reference Answer

N. S. Lagutina[1], K. V. Lagutina[1]

DOI: 10.18255/1818-1015-2025-1-42-65

[1]P.G. Demidov Yaroslavl State University, Yaroslavl, Russia

MSC2020: 68T50
Research article
Full text in Russian

Received February 10, 2025
Revised February 20, 2025
Accepted February 26, 2025

The development of automatic assessment systems is a relevant task designed to simplify the routine work of a teacher and speed up feedback for a student. The survey is devoted to research in the field of automatic assessment of student answers based on a teacher's reference answer. The authors of the work analyzed text models used for the tasks of automatic assessment of short answers (ASAG) and automated essay assessment (AES). Several approaches were also taken into account for the task of determining the text similarity, since it is a close task, and the methods for solving it can also be useful for analyzing student answers.

Text models can be divided into several large categories. The first is linguistic models based on various stylometric features, both simple ones like a bag of words and n-grams, and complex ones like syntactic and semantic ones. The authors attributed neural network models based on various embeddings to the second category. It highlights large language models as universal, popular and high-quality modeling methods. The third category includes combined models that unite both linguistic features and neural network embeddings. A comparison of modern studies on models, methods and quality metrics showed that the trends in the subject area coincide with the trends in computational linguistics in general. A large number of authors choose large language models to solve their problems, but standard features remain in demand. It is impossible to single out a universal approach; each subtask requires a separate choice of method and adjustment of its parameters. Combined and ensemble approaches allow achieving higher quality than other methods. The vast majority of studies examine texts in English. However, successful results for national languages are also found. It can be concluded that the development and adaptation of methods for assessing students' answers in national languages is a relevant and promising task.

Keywords: natural language processing; text similarity; text classification; neural network language models; assessing students' answers; artificial intelligence in education

INFORMATION ABOUT THE AUTHORS

Lagutina, Nadezhda S.	ORCID iD: 0000-0002-6137-8643. E-mail: lagutinans@gmail.com PhD, associate professor
Lagutina, Ksenia V. (corresponding author)	ORCID iD: 0000-0002-1742-3240. E-mail: lagutinakv@mail.ru PhD, associate professor

Funding: This work was supported by a grant from the Russian Science Foundation (Project no. 25-21-00196).

For citation: N. S. Lagutina and K. V. Lagutina, "A survey of models for automatic assessment of similarity of student's answer to the reference answer", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 42–65, 2025. DOI: 10.18255/1818-1015-2025-1-42-65.

© Lagutina N. S., Lagutina K. V., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

Обзор моделей автоматической оценки сходства ответа учащегося с эталонным ответом

Н. С. Лагутина[1], К. В. Лагутина[1]

DOI: 10.18255/1818-1015-2025-1-42-65

[1]Ярославский государственный университет им. П.Г. Демидова, Ярославль, Россия

УДК 004.912
Научная статья
Полный текст на русском языке

Получена 10 февраля 2025 г.
После доработки 20 февраля 2025 г.
Принята к публикации 26 февраля 2025 г.

Разработка систем автоматического оценивания является актуальной задачей, призванной упростить рутинный труд учителя и ускорить обратную связь для учащегося. Обзор посвящён исследованиям в области автоматической оценки ответов учащихся на основе эталонного ответа учителя. Авторы работы проанализировали модели текстов, применяемые для задач автоматической оценки коротких ответов (ASAG) и автоматизированной оценки эссе (AES). Также принималось во внимание несколько подходов для задачи определения близости текстов, так как она является аналогичной задачей, и методы её решения могут быть полезны и для анализа ответов студентов.

Модели текста можно разделить на несколько больших категорий. Первая — это лингвистические модели, основанные на разнообразных стилометрических характеристиках, как простых вроде мешка слов и n-грамм, так и сложных вроде синтаксических и семантических. Ко второй категории авторы отнесли нейросетевые модели, основанные на разнообразных эмбеддингах. В ней выделяются большие языковые модели как универсальные, популярные и качественные методы моделирования. Третья категория включает в себя комбинированные модели, которые объединяют в себе как лингвистические характеристики, так и нейросетевые эмбеддинги. Сравнение современных исследований по моделям, методам и метрикам качества показало, что тренды в предметной области совпадают с трендами в компьютерной лингвистике в целом. Большое количество авторов выбирают для решения своих задач большие языковые модели, но и стандартные характеристики остаются востребованными. Универсальный подход выделить нельзя, каждая подзадача требует отдельного выбора метода и настройки его параметров. Комбинированные и ансамблевые подходы позволяют достичь более высокого качества, чем остальные методы. В подавляющем большинстве работ исследуются тексты на английском языке. Однако успешные результаты для национальных языков также встречаются. Можно сделать вывод, что разработка и адаптация методов оценки ответов студентов на национальных языках является актуальной и перспективной задачей.

Ключевые слова: обработка естественного языка; сходство текстов; классификация текстов; нейросетевые языковые модели; оценка ответов учащихся; искусственный интеллект в образовании

ИНФОРМАЦИЯ ОБ АВТОРАХ

Лагутина, Надежда Станиславовна | ORCID iD: 0000-0002-6137-8643. E-mail: lagutinans@gmail.com
Канд. физ.-мат. наук, доцент

Лагутина, Ксения Владимировна (автор для корреспонденции) | ORCID iD: 0000-0002-1742-3240. E-mail: lagutinakv@mail.ru
Канд. тех. наук, доцент

Финансирование: Исследование выполнено за счет гранта Российского научного фонда № 25-21-00196.

Для цитирования: N. S. Lagutina and K. V. Lagutina, “A survey of models for automatic assessment of similarity of student’s answer to the reference answer”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 42–65, 2025. DOI: 10.18255/1818-1015-2025-1-42-65.

© Лагутина Н. С., Лагутина К. В., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

Определение и оценка сходства ответа учащегося с эталонным ответом учителя является основой эффективного контроля знаний. Классический подход к проверке преподавателем выполненных заданий с развернутым ответом очень трудоёмкий, утомительный и субъективный [1]. Автоматическая оценка работ позволяет упростить и ускорить этот процесс, основываясь на объективных критериях, что способствует более справедливому отношению к учащимся и повышению мотивации к обучению [2].

В области применения методов искусственного интеллекта к оценке развернутых ответов на естественном языке можно условно обозначить два направления: автоматическая оценка коротких ответов (automatic short answer grading, ASAG) и автоматизированная оценка эссе (automated essay scoring, AES). Типичным критерием разделения является длина текста. В обоих случаях оценка сосредоточена на семантике, а не на синтаксисе ответа [3, 4]. Конкретные результаты работы систем автоматической оценки ответов могут сильно отличаться в зависимости от постановки задачи или цели системы. Для направления ASAG характерно прогнозирование бинарной оценки правильности ответа или оценки по заданной шкале, AES также может иметь целью получить определённые баллы, но часто предлагаются качественные критерии, вычисляемые на основе автоматической обработки текста [5]. Во многих случаях основную роль играет сравнение с эталонным ответом или ответами.

В исторической ретроспективе методы оценки развернутых ответов развиваются в соответствии с методами обработки естественного языка (natural language processing, NLP). Ранние системы автоматической оценки письма использовали простейшие характеристики текста, такие как частоты символов и слов, средняя длина предложения и т. п. С появлением более мощных компьютеров и методов NLP, анализирующих структуру слов и предложений, появились более сложные характеристики, отражающие контекст употребления слов и семантику текста [6]. Машинное обучение и большие языковые модели в последнее десятилетие позволили вычислять и отбирать релевантные признаки текста для прогнозирования оценки и показали высокое качество при обучении на больших корпусах данных, качественно размеченных экспертами [7].

Прорыв в развитии языковых моделей, позволяющих более эффективно находить смысловую составляющую текстов, вызвал существенное увеличение количества исследований по оценке открытых ответов и необходимость систематизации результатов. Поэтому авторы данной работы поставили целью анализ современных достижений в этой области. Основной задачей была выбрана автоматическая оценка ответов учащихся на основе близости к эталону. Формальное понятие близости или сходства определяется через расстояние между числовыми характеристическими векторами текстов или как результат классификации таких векторов в соответствии с выбранной шкалой оценок. Все исследования рассматривались с точки зрения используемых моделей.

Когда автоматическая оценка ответов учащихся ставится как задача классификации, это может быть либо бинарная классификация (правильный/неправильный ответ), либо мультиклассификация в двух вариантах: классификация по категории правильности (правильный/частично правильный/неправильный ответ/ответ не по теме и т. д.) и классификация по оценке, когда за ответ выставляется балл, например, по пятибалльной шкале. Последняя задача также может решаться как задача регрессии, но, так как баллы обычно ставятся либо целые, либо с долей 0.5, набор итоговых баллов строго ограничен, и задача сводится к классификационной.

В качестве исходных данных для автоматической оценки ответа учащегося выступают сам текстовый ответ, эталонный ответ и заданный вопрос. Ответ учащегося — это основной моделируемый текст. Эталонный ответ применяется в первую очередь для сравнения с ответом учащегося, но в некоторых случаях он может быть использован как источник для моделирования предметной области в целом, например, создания тезауруса. Заданный вопрос чаще всего не используется

совсем, но в некоторых исследованиях его текст применяется для определения темы текстов или моделирования предметной области.

Качество работы методов для автоматической оценки ответа учащегося определяется при помощи стандартных метрик для классификации и регрессии. К первым относятся доля правильных ответов (accuracy), точность, полнота и F-мера. Ко вторым относится среднеквадратичная ошибка (RMSE), средняя абсолютная ошибка (MAE) и квадратично-взвешенная каппа (QWK). В некоторых работах также встречаются каппа Коэна и коэффициент корреляции Пирсона. Часть исследователей ограничиваются экспертными оценками, когда не проводят масштабных экспериментов.

Обзор структурирован следующим образом. В разделе 1 рассматриваются модели текста на основе лингвистических характеристик, как стандартных стилометрических, так и сложных структурных и семантических. В разделе 2 обсуждаются нейросетевые модели текста. Раздел 3 посвящён моделям, комбинирующим нейросетевые эмбеддинги и лингвистические характеристики. В заключении делается вывод о состоянии предметной области.

1. Лингвистические модели

1.1. Модели уровня слов

Лингвистические характеристики входят в состав самых первых моделей текста и не теряют актуальности сейчас. Числовые характеристики уровня слов без учёта их порядка (модель «мешок слов»), хотя и почти не учитывают контекст и синтаксис естественного языка, тем не менее несут важную информацию о содержании текста, которую необходимо принимать во внимание при оценке ответов учащихся.

Hsu и др. [8] разработали автоматизированную систему оценки кратких ответов. Авторы работали со студентами своего университета, собрав собственный англоязычный корпус ответов, представляющих собой текстовые описания программ. Метод оценки использовал логистическую регрессию для биграмм и числовых характеристик на основе мешка слов, показав долю правильных ответов около 90 %. Исследователи отмечают, что в их системе возникло существенное количество ложноотрицательных срабатываний, что пришлось компенсировать механизмом ручных апелляций от студентов.

Suzen и др. [9] оценивали англоязычные короткие тексты студентов из университета Северного Техаса. Корпус состоял из 29 ответов, оценки за ответы учащихся выставлялись по шкале от 0 до 5. Распределение по классам (оценкам за ответ) оказалось несбалансированным: большинство ответов получили оценку 5, второй по популярности была оценка 2. Авторы сравнивали ответ студента с эталонным, базируясь на количестве общих слов. До эталонного ответа вычислялось расстояние Хэмминга, которое затем подставлялось в авторскую формулу расстояния. Полученные ответы сопоставлялись с оценками двух преподавателей, которые были получены вручную. Среднеквадратичное отклонение между результатом работы автоматического метода и средним значением двух преподавательских оценок составило 0.17, доля правильных ответов — 17 %. Авторы отмечают, что их модель следует усложнить, например, с применением синонимов.

Классический вариант моделирования текста в рамках подхода «мешок слов» применён в работе [10]. Векторизация осуществлялась на основе частот слов и биграмм и метрики TF-IDF. Для классификации по пяти категориям (полностью правильный или аналогичный эталонному ответ; частично правильный ответ; противоречащий эталонному ответ; неуместный ответ в рамках предметной области; ответ, не относящийся к предметной области и демонстрирующий отсутствие знаний) использовались методы опорных векторов, случайного леса, k-ближайших соседей и искусственной нейронной сети. Для экспериментов был взят англоязычный набор ответов на открытые вопросы SciEntsBank. В данном корпусе около 40 % правильных ответов, около 27 % частично правильных,

около 22 % неуместных, около 10 % противоречащих и 1 % ответов не из предметной области. Лучший результат по доле правильных ответов оказался 70 %.

Poguda и Tape [11] разработали систему автоматического оценивания англоязычных ответов учащихся по пятибалльной системе. Основой оценки явился расчет схожести между студенческим ответом и эталонным текстом с использованием алгоритма Джаро — Левенштейна, учитывающего как порядок символов, так и их совпадения. Дополнительно оценивались оригинальность, полнота и корректность текстов. Для окончательной оценки использовались алгоритмы машинного обучения. К сожалению, нет описания корпуса, на котором проводились эксперименты, и нет статистических оценок качества. Среди положительных качеств системы является возможность обнаружения плагиата в студенческих работах.

Pribadi и др. [12] оценивали короткие ответы из корпуса университета Северного Техаса. Авторы дополнили в нем набор эталонных ответов с помощью алгоритма суммаризации максимальной предельной релевантности на основе студенческих ответов и имеющихся эталонных ответов. Затем сгенерированные тексты были доработаны вручную. Такой метод сделал эталонные ответы более разнообразными. Измерение близости между студенческими и эталонными ответами производилось по методу GAN-LCS, предложенному авторами. Данная формула подсчета близости двух текстов опирается на самую длинную общую подпоследовательность символов между двумя текстами и на длины этих текстов. Для каждого студенческого ответа считалась метрика GAN-LCS до каждого эталонного ответа, затем бралось максимальное значение. Среднее значение среднеквадратичной ошибки составило 0.88, шкала оценок за ответы учащихся была от 0 до 5.

В работе [13] автоматически оценивались короткие описания программного кода, данные студентами. Авторы вычисляли простые стилометрические характеристики текстов: длину, долю самостоятельных частей речи, метрику читабельности. Также авторы применяли стандартные методы измерения близости текстов: ChrF и METEOR, основанные на n-граммах, и BERTScore, основанную на языковой модели BERT. Авторы экспериментировали с корпусом из 3019 пар ответ — эталон, подсчитывали метрики близости, но не вычисляли общие метрики качества. Эксперименты с анализом статистики у полученных характеристик и метрик близости показали, что поверхностные стилометрические характеристики текстов не позволяют отличить правильные ответы от неправильных. Только метрика METEOR позволила отделить правильные ответы от неправильных по близости к эталону.

Meccawy и др. [14] использовали для моделирования коротких ответов из корпуса AR-ASAG «мешок слов» и синонимы из арабского WordNet, а для вычисления близости текстов — косинусную метрику. Оценки учащимся выставлялись по шкале от 0 до 5. Значение среднеквадратичной ошибки, равное 1.12, достаточно хорошее, но использование нейросетевой языковой модели позволило авторам улучшить результат, как показано в следующем разделе.

В работе [15] для слов из коротких ответов искались синонимы в арабском WordNet. Слова из ответа и их синонимы сопоставлялись со словами из эталонного ответа. Для измерения близости ответа к эталону использовался алгоритм LCS, основанный на самой длинной общей подпоследовательности символов. Оценка за ответ выставлялась по шкале от 0 до 10. Эксперименты на собственном корпусе из 330 ответов студентов показали среднеквадратичную ошибку, равную 0.81. F-мера варьировалась от 60 % до 96 % для разных вопросов.

Buditjahjanto и др. [16] использовали «мешок слов», TF-IDF и косинусную метрику для определения близости эссе к эталонному. Далее для определения итоговой оценки применялась нейронная сеть с обратным распространением ошибки. Эксперименты с собственным корпусом из эссе на индонезийском языке показали MAE, равную 0.06, и F-меру, равную 88.57 %.

Непосредственное отношение к анализу ответов учащихся имеет задача определения сходства текстов. В этом случае под сходством текстов понимается смысловая (семантическая) близость

свободно сконструированного ответа учащегося с заведомо правильным, эталонным ответом преподавателя [17].

Авторы работы [18] проанализировали наборы лексических характеристик уровня символов, слов и структуры для определения близости текстов на английском языке. Текст на естественном языке преобразовывался в вектор чисел на основе стилометрической модели, для пар векторов текстов рассчитывались метрики близости пяти видов: косинусное сходство, коэффициент корреляции Пирсона, метрика Чебышева, евклидово расстояние, метрика Минковского. Если метрика близости оказывалась выше заданного порога, то тексты классифицировались как близкие, иначе нет. Ответы метода сравнивались с эталонными с помощью стандартных метрик качества. Для экспериментальной апробации векторных моделей были выбраны два корпуса англоязычных текстов: Text Similarity[1], содержащий 1613 пар близких и 529 пар неблизких текстов; собственный корпус из ответов студентов и эталонных, содержащий 618 пар близких и 1195 пар неблизких текстов. На корпусе Text Similarity лучшее качество показала модель на основе характеристик уровня слов: F-мера 86 %. Для собственного корпуса векторное представление уровня символов показало лучшее значение F-меры, равное 80 %.

Крюкова А. В. [19] решала проблему оценки семантической близости текстов на русском языке с помощью открытого программного инструмента DKPro Similarity [20]. В нём реализованы 15 различных строковых метрик близости. Эксперименты были проведены с текстами из четырёх наборов: аннотации научных статей, новости СПбГУ, переводы романа Набокова, заголовки новостных статей. Из каждой группы случайным образом было выбрано несколько текстов: пять аннотаций; по пять сообщений из двух частей новостного корпуса; пять соответствующих друг другу отрывков из трех переводов; пять пар заголовков. Эталонное сходство текстов определялось экспертами по шкале «0–1–2», где «0» — небольшая степень схожести, «1» — средняя степень, «2» — сильная степень схожести. Для задач классификации эксперименты проводились с несколькими моделями: логистическая регрессия, гребневый классификатор, классификатор SGD, пассивно-агрессивный классификатор, перцептрон. Объём обучающей выборки составлял примерно две трети (23 текста), а тестовой, соответственно, треть (12 объектов). Лучшую F-меру 63–100 % для разных групп текстов показали мера включения n-грамм, косинусная метрика и классификатор на основе логистической регрессии.

Все подходы в рассмотренных исследованиях объединяет простота реализации методов и возможность применения для корпусов текста ограниченного размера. Однако заметно, что оценка качества сильно колеблется даже в рамках одной научной работы. Анализ результатов показывает большое количество ложных результатов. Для преодоления недостатков часто используется работа экспертов, а также предлагается использование более сложных методов.

1.2. Модели на основе лингвистических правил и синтаксиса

Более сложные лингвистические характеристики связаны в первую очередь с синтаксисом фраз и предложений естественного языка. С одной стороны, современные инструменты NLP уже содержат алгоритмы определения лемм, морфологических характеристик, ключевых слов, параметров структуры предложений и дают возможность быстро реализовывать сложные правила проверки, с другой стороны, остаются трудоёмкими.

Nandini и Uma Maheswari [21] разработали систему, которая использует метод извлечения признаков на основе синтаксических отношений для автоматической оценки ответов описательного типа. Для моделирования ответов применялись тип ответа, определяемый из текста вопроса, n-граммы, ключевые слова, отфильтрованные по алгоритму разрешения лексической многозначности Lesk, а также общие слова. Для определения близости к эталону использовалась комплексная

[1] https://www.kaggle.com/datasets/rishisankineni/text-similarity

метрика, учитывающая косинусную метрику, коэффициент Жаккара и количество общих слов. Корпус состоял из 50 вопросов, на которые дали ответы 10 студентов. Качество работы алгоритма достигло 95 % точности и 94 % полноты.

Авторы исследования [22] ставят задачу разработать систему автоматизированной проверки ответов учащихся на русском языке на открытые вопросы. Предложения в ответе студента могут быть любой сложности. Эталонный ответ представляет собой набор коротких односложных предложений, смысл которых должен присутствовать в ответе. Разработанная система анализирует ответы и на основе сравнения с эталонным и выставляет предварительную оценку. При сравнении учитываются полностью совпадающие слова, синонимы, антонимы, различные формы слов, написания дат и имен собственных, а также использование местоимений. Метод решения задачи комбинирует сравнение деревьев, построенных на основе графа синтаксического разбора, и извлечение фактов и формирует числовую метрику текста ответа. Система, руководствуясь заданными преподавателем порогами, относит ответ учащегося к одному из трех классов: правильный, неправильный или неопределённый, требующий дополнительной ручной проверки преподавателем. Эксперимент с 456 текстами показал F-меру 70 %. Интересно, что отсутствовали ошибки первого и второго рода, но 57 ответов, оценённых преподавателем как верные, и 77, оценённых как неверные, попали в класс неопределённых.

Зафиевский и др. [23] рассматривают модель, предназначенную для автоматической оценки делового письма на заданную тему на английском языке. Параметры оценки сформулированы и формализованы в виде 14 критериев при помощи экспертов в области обучения английскому языку. Критерии включают анализ лексики, особенности предметной области, тематики текста, стиля и формата письма, наличие средств логической связи предложений. Алгоритмы определения критериев основаны на анализе состава и структуры предложений, используют специализированные словари и регулярные выражения. Проведён эксперимент по анализу результатов работы этой системы на корпусе из 20 текстов. Автоматическая оценка и оценка экспертов сравнивались с помощью тепловых карт и технологии двумерного представления векторов UMAP, применённой к характеристическим векторам текстов. В большинстве случаев не было выявлено значимых различий между этими оценками.

Программный модуль для оценки открытых ответов на русском языке построен на основе системы формальных грамматик и правил и описан в работе [24]. Автор вводит индивидуальные концептуальные грамматики для синтаксического анализа текста в условиях, определённых контекстом вопроса. В рамках грамматик формулируются правила использования лексем, отношений между ними, полноты ответа. В тексте выделяются смысловые сегменты, которые анализируются на соответствие правилам, процесс продолжается рекурсивно до завершения текста ответа. В результате вычисляются семь числовых характеристик, объединённых в вектор ситуации. Эксперимент проведён для 20 ответов, в ходе которого вектора этих текстов сравнивались с векторами эталонных. В описании работы приведён анализ ошибочных ситуаций и сделан вывод о возможной коррекции правил.

Анализ синтаксических структур предложений лежит в основе определения сходства русскоязычных текстов в исследовании [25]. Авторы предлагают пять числовых критериев, агрегация которых показывает долю схожести текстов: доля покрытия предложения-эталона предложением сопоставляемого текста на основе метрики TF-IDF; оценка информационной значимости слов предложения-эталона в предложении сравниваемого текста; оценка сходства предложения-эталона и предложения сравниваемого текста на основе совпадения синтаксических структур; доля совпавших семантических значений у словоупотреблений в предложении эталона и в предложении сопоставляемого текста; оценка совпадающих семантических связей на основе роли слов в предложении. Однако в работе отсутствует описание экспериментов с корпусом текстов.

Относительно небольшое количество исследований, применяющих сложные лингвистические характеристики, можно объяснить несоответствием между приростом качества решения задачи и трудоёмкости методов. Однако, этот подход может оказаться эффективным для отражения специфики предметных областей, особенно в случае малого размера обучающего корпуса текстов. Кроме того, рассматриваемые характеристики текста могут быть частью комплексных методов определения сходства текстов и оценки ответов.

1.3. Модели на основе семантики

Самой сложной задачей любой области NLP является выявление смысловой составляющей текста. Семантическую близость понятий часто определяет отношение синонимии. Некоторые подходы предоставляют методы извлечения информации, тезаурусы и онтологии. Высококачественное определение семантики текста еще впереди, однако и существующие подходы успешно используются для оценки ответов.

Ouahrani и Bennouar [26] собрали корпус арабских текстов AR-ASAG для измерения близости короткого ответа к эталону. Для демонстрации работы с корпусом был поставлен эксперимент по определению близости текстов косинусной метрикой. Ответы моделировались параметрами «мешка слов», причём слова выбирались из семантического пространства, предварительно составленного для корпуса методом COALS. Характеристикой слова для вектора текста выступала NTFlog. Среднеквадратичная ошибка данного метода составила в среднем 0.80.

Хорошие результаты показала система автоматического оценивания ответов на открытые вопросы на русском языке на правильные и неправильные, описанная в статье [27]. Из ответа учащегося с помощью Томита-парсера извлекается набор фактов, которые сравниваются с эталонным ответом. Грамматические правила, позволяющие извлекать факты, строятся отдельно для каждого конкретного вопроса, кроме того, вводятся весовые коэффициенты значимости. Такой подход приводит к высокой F-мере 90 %, но не позволяет легко применять метод для аналогичных задач из других предметных областей.

В работе [28] предлагается решение задачи определения семантической близости команды, выданной на естественном языке, с эталонным текстом. Сравниваемые тексты относятся к одной предметной области и содержат термины из определённого набора. Моделирование текста происходит на основе тезаурусного графа основных и второстепенных терминов предметной области со связями трёх видов: «включает в себя», «обеспечивает результат» и «соответствует». Степень семантической близости определяется по количеству совпавших понятий, а также расстоянию между соответствующими вершинами в графе. К сожалению, описания корпуса текстов и экспериментов нет.

Maharjan и Rus [29] предложили оценивать ответы студентов с помощью концептуальной карты для предметной области. Концептуальная карта представляет собой базу знаний, сформированную экспертами. В процессе оценивания ответов авторы сопоставляли слова и другие фрагменты ответов с данной картой, измеряя близость фрагментов к концептам из карты. Полученные вектора характеристик для студенческих ответов и для эталонных ответов конкатенировались и использовались для классификации студенческого ответа как корректного/некорректного/неполного/спорного. Бинарная классификация на корректный/некорректный осуществлялась с помощью косинусной метрики с эмпирически установленным порогом, F-мера на собственном корпусе достигла 80.2 %. Для мультиклассификации использовался классификатор LSTM, F-мера на корпусе DT-Grade достигла 62.0 %.

Автор работы [30] обращает внимание на то, что проблема получения новых эффективных алгоритмов для вычисления семантической близости предложений остается актуальной, поскольку современные методы часто не дают достоверных результатов. Он предлагает алгоритм определения схожести предложений на основе вычисления семантической близости биграмм и триграмм. На основе корпуса текстов предметной области строятся бинарные деревья понятий с иерархиче-

скими связями. Семантическая близость двух понятий определяется длиной пути наследования, сложность которого вычисляется разностью бинарных иерархических чисел [31]. Эксперимент был проведён с семью новостными предложениями на политическую тематику на русском и английском языках. Программная реализация алгоритма позволила построить матрицу попарного сходства, которая была проанализирована экспертом, сделавшим качественный вывод о пригодности предложенного подхода к решению поставленной задачи.

1.4. Сравнение лингвистических моделей

Исследования, опирающиеся на лингвистические модели текстов, приведены в таблице 1. Для исследований указаны тип задачи, язык текстов, модель текста, метод сравнения или классификации, лучшее качество результата или диапазон лучших значений, если корпусов было несколько.

Типы задач по анализу студенческих текстов указаны англоязычными аббревиатурами (ASAG — automatic short answer grading — автоматическая оценка кратких ответов, AES — automated essay scoring — автоматизированная оценка эссе), отдельно указаны исследования по определению близости других текстов. Наиболее популярной задачей является автоматическая оценка кратких ответов, она встречается чаще всего. Автоматизированная оценка эссе также распространена, но когда исследователям нужно сравнить эссе с эталонным, они прибегают не к лингвистическим моделям, что будет видно в следующих разделах.

Наиболее хорошо и часто изучаются тексты на английском языке. Однако следует отметить, что для русского и арабского языков область активно развивается. Для арабского языка это отчасти связано с появлением арабской версии WordNet и открытого корпуса AR-ASAG. Других открытых корпусов для разных языков существует немного, но это не является существенным препятствием: очень многие авторы собирают собственные корпуса текстов.

Среди лингвистических моделей наиболее часто используются стандартные стилометрические характеристики: «мешок слов», n-граммы, синонимы из WordNet. Однако сами по себе они редко дают высокое качество результата, поэтому авторы их применяют в комплексе с синтаксическими и семантическими характеристиками текстов, что повышает статистические метрики оценки решения.

Метод для сравнения и бинарной классификации векторных моделей — это в большинстве случаев косинусная метрика с пороговым значением, подобранным вручную. Для мультиклассовой классификации, то есть выставления оценки студенческому ответу, обычно используются или регрессионные алгоритмы, или нейронные сети.

Общая оценка качества анализа студенческих текстов часто дается с помощью стандартных для классификации метрик: доли правильных ответов (accuracy в таблице), точности, полноты и F-меры. В таблице приводится в основном F-мера, как наиболее сбалансированная и показательная метрика или доля правильных ответов, если F-меру исследователи не вычисляли. Помимо данных метрик качества, в задачах сравнения близости текстов часто измеряется среднеквадратичная ошибка, которая должна быть как можно меньше. Вместе со значением среднеквадратичной ошибки приводится шкала, по которой выставлялась оценка учащимся, чтобы сделать поправку на диапазон значений: чем больше шкала оценок, тем большее значение среднеквадратичной ошибки можно считать хорошим. Значения метрик качества сильно варьируются: от 62 до 100 % F-меры, от 17 до 90 % доли правильных ответов. Только у среднеквадратичной ошибки можно указать достаточно часто встречающийся уровень около 0.8. Некоторые авторы вообще не приводят данных об общих метриках качества, что означает, что модель и метод решения подбираются под конкретные условия и оцениваются вручную. Таким образом, среди лингвистических моделей текста нет универсального подхода, дающего стабильно высокие результаты.

Table 1. Research where the tasks of analASAG: Automatic Short Answer Gradingyzing a text answer are solved using linguistic models

Таблица 1. Исследования, где задачи анализа текстового ответа решаются с помощью лингвистических моделей

Авторы	Задача	Язык	Модель	Метод	Качество
Hsu и др. [8]	ASAG	Английский	Мешок слов	Логистическая регрессия	Accuracy 90 %
Suzen и др. [9]	ASAG	Английский	Мешок слов	Расстояние Хэмминга	Accuracy 17 %
Кербенева [10]	ASAG	Английский	Мешок слов	Нейронная сеть	Accuracy 70 %
Poguda и Tape [11]	ASAG	Английский	Мешок слов	Алгоритм Джаро-Левенштейна	-
Pribadi и др. [12]	ASAG	Английский	GAN-LCS	GAN-LCS	RMSE 0.88 шкала 0–5
Lekshmi-Narayanan и др. [13]	ASAG	Английский	n-граммы	METEOR Экспертная оценка	-
Meccawy и др. [14]	ASAG	Арабский	WordNet	Косинусная метрика	RMSE 1.12 шкала 0–5
Abdeljaber и др. [15]	ASAG	Арабский	WordNet	LCS	F-мера 60–96 % RMSE 0.81 шкала 0–10
Buditjahjanto и др. [16]	AES	Индонезийский	Мешок слов	Косинусная метрика BPNN	MAE 0.06 F-мера 89 %
Лагутина и др. [18]	Близость текстов	Английский	Мешок слов	Корелляция Пирсона	F-мера 80–86 %
Крюкова А. В. [19]	Близость текстов	Русский	Косинусная метрика, мера включения n-грамм	Логистическая регрессия	F-мера 63–100 %
Nandini и др. [21]	ASAG	Английский	Тип ответа n-граммы Ключевые слова Общие слова	Косинусная метрика Коэффициент Жаккара	Точность 95 % Полнота 94 %
Леонов и др. [22]	ASAG	Русский	Граф синтаксических связей	Косинусная метрика	F-мера 70 %
Зафиевский и др. [23]	AES	Английский	Правила	Косинусная метрика	-
Прокопьев [24]	Близость текстов	Русский	Грамматики и правила	Экспертная оценка	-
Хорошилов и др. [25]	Близость текстов	Русский	Синтаксические характеристики	Экспертная оценка	-
Ouahrani и Bennouar [26]	ASAG	Арабский	Мешок слов	Косинусная метрика	RMSE 0.80 шкала 0–5
Кожевников и др. [27]	ASAG	Русский	Грамматические правила	Доля правильности	F-мера 90 %
Гадасин и др. [28]	ASAG	Русский	Тезаурус	Экспертная оценка	-
Maharjan и Rus [29]	ASAG	Английский	Концептуальная карта	Косинусная метрика LSTM	F-мера 80 % F-мера 62 %
Каширин [30]	Близость текстов	Русский Английский	Бинарные иерархические числа	Экспертная оценка	-

ASAG — automatic short answer grading, автоматическая оценка кратких ответов.
AES — automated essay scoring, автоматизированная оценка эссе.
RMSE — root mean square error, среднеквадратичная ошибка.

2. Нейросетевые модели

2.1. Большие языковые модели

Огромное количество методов, так или иначе связанных с нейронными сетями, отражает современное состояние NLP. С точки зрения моделирования текста, следует выделить уже ставшие

классическими эмбеддинги и их комбинации с другими характеристиками. При моделировании текстов только с помощью эмбеддингов значительная часть исследователей напрямую использует числовые параметры больших языковых моделей для каждого текста. Отдельно хотелось бы выделить подход, агрегирующий эмбеддинги ответов и эталона, он будет рассмотрен в следующем разделе. Этот раздел посвящен большим языковым моделям, построенными на основе нейронных сетей, которые учитывают многие особенности устройства естественного языка. Эмбеддинги становятся характеристическими векторами текстов, а сравнение с эталоном трактуется как определение близости соответствующих векторов.

В работе [14], описанной ранее, авторы использовали для моделирования коротких ответов из корпуса AR-ASAG не только синонимы из WordNet, но и эмбеддинги word2vec и AraBERT. Вектора эталонных и проверяемых ответов сравнивались по косинусной метрике. Языковая модель AraBERT позволила достичь лучших результатов, чем модель на основе лингвистических характеристик: среднеквадратичная ошибка 1.00 против 1.12.

В работе [32] рассматривалось применение векторных моделей для анализа ответов студентов, сформулированных в свободной форме на русском языке. В качестве моделей представления слов и документов были выбраны word2vec, doc2vec, BERT. Сравнение ответа, данного обучающимся, и эталонного с помощью косинусной меры показало, что более качественно модели выявляют неверные ответы. Такой же вывод можно сделать из экспериментов бинарной классификации на правильные и неправильные ответы, где использовались эмбеддинги word2vec и классификатор Случайный лес. F-мера оказалась 74 % для определения класса неверных ответов и 55 % для верных. Для повышения качества авторы предлагают проводить дополнительные этапы проверки, такие как анализ ключевых слов и экспертная проверка.

В продолжении исследований [33] предложен метод оценивания ответов через нахождение косинусного сходства полученных векторов и уточнение оценок проверкой ключевых слов. Векторное представление текста строилось на основе модели BERT. Отдельно оценивались ответы на каждый из нескольких вопросов по теме «Компьютерная графика». Максимальная доля автоматического определения правильности и неправильности в соответствии с экспертным мнением составила 90 %, средняя: 77 %.

Ndukwe и др. [34] применили Sentence-BERT, чтобы поставить 228 ответам студентов оценку от 0 до 5. Эталонные ответы преподавателей получили оценку 5 и использовались для обучения нейронной сети. Квадратично-взвешенная каппа достигла 0.70. Авторы очень коротко описали постановку и результаты экспериментов и значения других метрик в работе не привели.

Fateen и Mine [35] оценивали короткие ответы на арабском языке с помощью нескольких подходов. В первом подходе моделировались вопрос, эталонный ответ и ответ студента с помощью языковой модели AraBERT, далее нейронная сеть выступала классификатором этих троек, определяя оценку ответа студента. Во втором подходе моделировались эталонный ответ и ответ студента также с AraBERT, а вместо классификатора использовалась сиамская нейронная сеть, которая оценивала близость ответа к эталону с помощью косинусной метрики. QWK для разных вопросов варьировалась очень сильно: от 0.02 до 0.80, так что оба подхода нельзя назвать стабильными.

Gaddipati и др. [36] применили предобученные языковые модели ELMo, BERT, GPT и GPT-2 для автоматической оценки кратких ответов. Ответы разделялись на слова, слова токенизировались, сумма эмбеддингов слов выступала эмбеддингом ответа. Для пар эмбеддингов эталонного и студенческого ответа вычислялась косинусная метрика близости, порог решения о том, близки тексты или нет, определялся на тренировочной выборке с помощью изотонической, линейной и гребневой регрессии. Эксперименты проводились на корпусе Mohler, состоящем из 2273 ответов, оцениваемых по шкале от 0 до 5. Среднеквадратичная ошибка составила 1.00 для ELMo и греб-

невой регрессии. Авторы провели сравнение с предыдущими работами, где экспериментировали с корпусом Mohler, и показали, что их результаты выше на 0.11.

Schneider и др. [37] оценивают корректность коротких ответов по близости к эталону: корректный ответ/некорректный ответ/не определено. Корпус вопросов-ответов состоит из 10 миллионов ответов на 990000 вопросов на 14 языках, собранных с сайта Classtime. Языки включают в себя в основном языки европейских стран, в том числе русский, а также турецкий и тайский. В корпусе больше правильных ответов, чем неправильных: 58 %, но для экспериментов авторы искусственно уравняли размеры классов, добавляя ответы на другие вопросы в качестве неправильных. Тексты на всех языках моделировались одинаково с помощью мультиязыковых моделей BERT и LaBSE, причём эмбеддинги ответа строились на основе текстов и ответа, и соответствующего вопроса. Близость между эмбеддингами ответов оценивалась с помощью формулы, основанной на косинусной метрике. Доля правильных ответов метода достигла 87 % для модели LaBSE, для BERT качество оказалось немногим ниже. Авторы отмечают, что для полноценной замены работы преподавателя автоматическим оценщиком данного уровня качества недостаточно, так как алгоритм совершает слишком много ошибок.

Hendre и др. [38] оценивают качество эссе с помощью эмбеддингов GSE, ELMo и GloVe. Близость ответа студента к эталонному считается с помощью косинусной метрики. В корпусе ASAP, использовавшемся для экспериментов, нет эталонных ответов, поэтому авторы выбрали в качестве эталонных эссе с наибольшей оценкой для каждого вопроса. Если эссе с высшей оценкой для вопроса было несколько, то выбиралось эссе с большей длиной. В качестве метрики для сравнения результатов метода с оценками двух преподавателей считалась корреляция Пирсона. Она достигла 0.52–0.69 для различных вопросов, лучшие результаты были достигнуты моделью GSE.

Doewes и др. [39] сравнивали эссе с их перефразированными версиями и измеряли близость между ними. Моделирование производилось несколькими способами: с помощью «мешка слов», TF-IDF, эмбеддингов Sentence-BERT и LASER. Метрикой близости выступала косинусная метрика, итоговая оценка за эссе выставлялась при помощи алгоритмов регрессии: гребневой, случайного леса или повышения градиента. На экспериментах с собственным корпусом QWK для эмбеддингов превысила QWK для стилометрических моделей и составило 0.72–0.77.

В работе [40] сравнивались 50 эссе студентов, изучающих английский, с образцовыми эссе от экспертов. В качестве моделей использовались эмбеддинги из открытых программных библиотек: InferSent, spaCy, ADW, DKPro, SEMILAR, LSA, скомбинированные с косинусной метрикой для измерения близости. Корреляция Пирсона с оценками экспертов составила 0.82 для эмбеддингов InferSent.

Dhini и др. [41] оценивали 1028 эссе на индонезийском языке, собранные из LMS университета Terbuka. Эссе моделировались как эмбеддинги предложений с помощью Sentence-BERT и как эмбеддинги ключевых слов с помощью KeyBERT. Близость между образцовыми текстами и эссе студентов считалась по косинусной метрике между данными эмбеддингами. Значение MAE достигло 0.71.

Работа [42] посвящена применению генеративных моделей, взаимодействующих с пользователем посредством чат-ботов ChatGPT и PerplexityAI на базе модели OpenAI GPT 3.5, для оценки студенческих эссе, написанных в формате стандартизированного экзамена по английскому языку. Эссе были проверены преподавателем по 9-балльной шкале IELTS, баллы были целочисленными или кратными 0.5. Оценивались тексты в целом и аспекты: решение коммуникативной задачи, связанность, лексический ресурс, грамматические навыки. Для экспериментов было использовано 19 эссе. Выставленные баллы были сопоставлены с оценкой преподавателя и двух чат-ботов путем вычисления коэффициентов согласованности (альфа Кронбаха) и межэкспертного согласия (каппы Коэна и Флейса). Доля оценок, отличавшихся не более чем на балл, была высока и варьировалась от 79 до 100 %, доля полностью идентичных оценок была мала, наибольшее согласие прослеживалось

у двух чат-ботов, что вполне объясняется общей базовой моделью языка. В результате качественного анализа выявлены особенности обратной связи от чат-ботов, такие как периодическое игнорирование инструкций в запросе, тенденция к нахождению несуществующих ошибок, выставление разных баллов одной и той же работе при последовательных запросах.

Преподаватель английского языка для русскоязычных студентов описывает опыт применения чат-бота Mistral AI для оценки эссе на заданную тему длиной не более 250 слов [43]. В статье приведён результат анализа одного эссе, который содержит баллы по отдельным критериям. Автор обращает внимание, что качество результата зависит от формулировки запроса к чат-боту.

В работе [44] рассматриваются результаты применения трех языковых моделей на основе BERT и GPT для задачи определения семантического сходства текста ответа учащегося и эталонного ответа учителя. Эксперимент проводился на собственном корпусе текстов, состоящем из 1812 пар схожих фраз и словосочетаний. Тексты классифицировались на похожие и не похожие на эталонный ответ на основе шести метрик сходства. Наиболее высокие результаты оказались для косинусной метрики и коэффициента корреляции Пирсона. Лучшая F-мера 55 % оказалась у модели sembeddings/model_gpt_trained. Лучшую точность 47 % показала модель bert-base-cased. Модель bert-large-cased обнаружила лучшую полноту 85 %.

В уже упоминавшейся работе [18] по определению сходных текстов сравнивались различные варианты больших языковых моделей BERT, GPT и Mamba. На корпусе Text Similarity все модели добились примерно одинакового качества результата: 85–87 % F-меры. Среди BERT-моделей лучшими оказались bert-base-cased и bert-base-uncased. Эмбеддинги GPT-моделей и Mamba сработали аналогично. Интересно, что стилометрические модели показали то же качество. На собственном корпусе эмбеддинги языковых моделей показали низкое качество: 50–56 % F-меры, среди них можно отметить только sembeddings/gptops_finetuned_mpnet_ gpu_v1. Их существенно опередили стилометрические характеристики, позволившие выбрать близкие тексты с качеством около 80 %. Вероятно, такой результат связан с меньшим размером собственного корпуса.

Таким образом, на текущий момент большие языковые модели становятся очень популярными для автоматизации оценок текстов как на английском, так и на других языках. Однако, требуется большое количество экспериментов для объективной оценки качества их работы с разных сторон: аспектного анализа языковых компетенций, особенностей предметных областей, анализа ошибок.

Можно заметить, что прямое использование эмбеддингов больших языковых моделей лучше большинства лингвистических характеристик, но недостаточно для построения по-настоящему эффективных систем оценок. Повысить качество удаётся дообучением или дополнением моделей.

2.2. Модели, использующие эталонный ответ для последующей классификации

В этом разделе рассматривается вариант решения задачи, когда характеристический вектор ответа формируется как модификация эмбеддингов исходного текста и эталона. После построения модели прогнозирование оценки ответа решается как мультиклассификационная задача.

Gomaa и Fahmy [45] доработали стандартную модель word2vec и представили Ans2vec, адаптированную для проверки коротких ответов. Разность векторов эталонного и студенческого ответа представляла собой итоговый вектор ответа. Оценка ответа вычислялась при помощи логистической регрессии. Лучшие результаты на корпусе университета Северного Техаса достигли среднеквадратичной ошибки 0.91, на корпусе Cairo — 0.92, на корпусе SCIENTSBANK — 0.46. Данные результаты близки к результатам других исследователей на этих же корпусах, но не превышают их, как показывают авторы при анализе результатов.

Weegar и Idestam-Almquis [46] оценивали 230 коротких ответов студентов, написанных на шведском и английском языках. Корпус с помощью алгоритмов кластеризации k-средних и GMM делился на тренировочную и тестовую выборки. Тексты моделировались как эмбеддинги BERT, также в вектор ответа добавлялось значение косинусной метрики близости между ответом студента и эталоном.

Классификаторами служили метод случайного леса, SVM и наивный байесовский классификатор, первый позволил достичь лучших результатов. F-мера оказалась 57 %, доля правильных ответов — 84 %.

Рассматриваемый подход к классификации ответов является стандартным с точки зрения методов машинного обучения. Невысокая F-мера, скорее всего, обусловлена слабым учётом особенностей предметной области.

2.3. Модели на основе ансамблей и собственных архитектур

Одним из способов учёта особенностей предметной области и повышения качества решения задачи является построение ансамблевых моделей и собственных оригинальных решений.

Li и др. [47] предложили семантическую сеть SFRN, основанную на связях из базы знаний о вопросах, справочных ответов или рубрик и маркированных ответов студентов. SFRN применялась для автоматической оценки кратких ответов. С её помощью для каждого ответа строились три вектора: для текста самого ответа, эталонного ответа и вопроса. Векторизация текстов производилась с помощью моделей LSTM и BERT. Три вектора объединялись в один посредством семантической нейронной сети. Корпусом для экспериментов выступал SemEval-2013, автоматически увеличенный с помощью переводов ответов студентов с английского на французский или китайский и обратно. Задача определения близости ставилась как классификационная задача: ответ правильный, неправильный, правильный частично, противоречивый, нерелевантный, не по теме. Классификатором выступил многослойный персептрон. Качество работы алгоритма оценивалось с помощью QWK и составило 0.79.

Tan и др. [48] оценивали короткие ответы студентов с помощью графовой свёрточной сети (GCN). Граф содержал два типа вершин. Первый тип — вершины на уровне предложений, соответствующие эталонным ответам, ответам, оценённым человеком, и ответам студентов, которые должны быть оценены. Второй тип — вершины на уровне слов/биграмм, соответствующие словам и биграммам из эталонных и студенческих. Вершины были соединены рёбрами в соответствии с их отношениями включения или совместной встречаемости. Вес ребра соответствовал TF-IDF или PMI (точечной взаимной информации). Подобный граф моделирует весь корпус, что обуславливает его способность решать классификационную задачу по оценке ответа студента. Эксперименты на корпусе SemEval-2013 позволили достичь F-меры 73 %. Эксперименты с китайским языком и двумя собственными тематическими корпусами текстов показали F-меру, равную 77 %, для корпуса по математической теме (17248 ответов) и 65 % для литературной темы (10104 ответов). Можно отметить, что алгоритм сработал хуже для корпуса меньшего размера.

Xie и др. [49] применяют для оценки эссе собственную модель NPCR. Она основывается на принципе того, что для двух заданных эссе расстояние между похожими эссе из одной и той же категории должно быть небольшим, в то время как расстояние между разнородными эссе должно быть большим, и семантическая связь может быть отражена путем измерения расстояния. В качестве эмбеддингов для входных данных NPCR брался BERT, который обеспечил QWK 0.82 на корпусе ASAP.

Garg и др. [50] составили ансамбль из двух методов оценивания короткого ответа. Первая часть ансамбля считала косинусную метрику близости между эмбеддингами ответа и эталона. Для подсчета эмбеддингов бралась версия BERT, предназначенная для вопросно-ответных систем. Вторая часть ансамбля объединяла ответ и эталон в общий текст и строила для них один BERT-эмбеддинг. Далее с помощью BERT-регрессора вычислялась оценка за ответ. Оценки из обеих частей ансамбля отправлялись в финальный нейросетевой классификатор, который предсказывал итоговый ответ. Качество работы на корпусе Mohler достигло среднеквадратичной ошибки, равной 0.73.

Комбинация эмбеддингов USE и BERT для инструмента визуализации поиска похожих текстов используется в работе [51]. Эмбеддинги вычисляются параллельно и конкатенируются, близость

Table 2. Research where text answers analysis tasks are solved using neural network models

Таблица 2. Исследования, где задачи анализа текстового ответа решаются с помощью нейросетевых моделей

Авторы	Задача	Язык	Модель	Метод	Качество
Meccawy и др. [14]	ASAG	Арабский	word2vec AraBERT	Косинусная метрика	RMSE 1.00 шкала 0–5
Миннегалиева и др. [32]	ASAG	Русский	word2vec	Косинусная метрика	F-мера 55–74 %
Миннегалиева и др. [33]	ASAG	Русский	BERT Косинусная метрика	Случайный лес	Accuracy 77 %
Ndukwe и др. [34]	ASAG	Английский	SBERT	Нейронная сеть	QWK 0.70
Fateen и Mine [35]	ASAG	Арабский	AraBERT	Siamese NN Косинусная метрика Нейронная сеть	QWK 0.02–0.80
Gaddipati и др. [36]	ASAG	Английский	ELMo, BERT, GPT, GPT-2	Косинусная метрика Гребневая регрессия	RMSE 1.00 шкала 0–5
Schneider и др. [37]	ASAG	14 языков	BERT, LaBSE	Косинусная метрика	Accuracy 87 %
Hendre и др. [38]	AES	Английский	GSE, ELMo GloVe	Косинусная метрика	Пирсон 0.52–0.69
Doewes и др. [39]	AES	Английский	SBERT, LASER	Косинусная метрика Повышение градиента	QWK 0.72–0.77
Wang [40]	AES	Английский	InferSent, spaCy, ADW, DKPro, SEMILAR, LSA	Косинусная метрика	Пирсон 0.82
Dhini и др. [41]	AES	Индонезий-ский	SBERT KeyBERT	Косинусная метрика	MAE 0.71
Боголерова и др. [42]	AES	Английский	ChatGPT PerplexityAI	Экспертная оценка	Accuracy 79–100 %
Евстигнеев [43]	AES	Английский	Mistral AI	Экспертная оценка	-
Копнин и др. [44]	Близость текстов	Английский	BERT, GPT	Косинусная метрика	F-мера 55 %
Лагутина и др. [18]	Близость текстов	Английский	Mamba BERT, GPT	Косинусная метрика	F-мера 85–87 %
Gomaa и Fahmy [45]	ASAG	Английский	Ans2vec	Линейный классификатор	RMSE 0.46–0.91 шкала 0–5
Weegar и др. [46]	ASAG	Шведский Английский	BERT Косинусная метрика	Случайный лес	F-мера 57 %
Li и др. [47]	ASAG	Английский	SFRN	Нейронная сеть	QWK 0.79
Tan и др. [48]	ASAG	Английский Китайский	GCN	GCN	F-мера 72.5 % F-мера 65–77 %
Xie и др. [49]	AES	Английский	NPCR, BERT	Нейронная сеть	QWK 0.82
Garg и др. [50]	ASAG	Английский	BERT	BERT-регрессор Косинусная метрика Нейронная сеть	RMSE 0.73 шкала 0–5
Witschard и др. [51]	Близость текстов	Английский	USE, BERT	Косинусная метрика	F-мера 55 %

ASAG — automatic short answer grading, автоматическая оценка кратких ответов.
AES — automated essay scoring, автоматизированная оценка эссе.
RMSE — root mean square error, среднеквадратичная ошибка.
QWK — quadratic weighted kappa, квадратично-взвешенная каппа.
MAE — mean absolute error, средняя абсолютная ошибка.

текстов измеряется по косинусной метрике. F-мера на корпусе IEEE VIS достигает не очень высокого значения 55 %.

2.4. Сравнение нейросетевых моделей

Исследования, опирающиеся на нейросетевые модели текстов, приведены в таблице 2. Таблица по структуре аналогична таблице из раздела 1.4.

Наиболее популярной задачей снова является автоматическая оценка кратких ответов, однако автоматизированная оценка эссе тоже часто решается при помощи нейросетей.

Мультиязычные языковые модели и языковые модели для конкретных языков позволили исследователям достигать хороших результатов не только для английского, но и многих других национальных языков, а также оценивать мультиязычные корпуса текстов. Тем не менее наиболее часто изучаются англоязычные тексты, особенно из открытого корпуса Mohler. Однако авторы собирают собственные корпуса и нередко проводят большое количество экспериментов и на собственном корпусе, и на Mohler в рамках одного исследования.

Тексты моделируются в основном с помощью языковых моделей, популярных и в других задачах компьютерной лингвистики, — BERT и GPT, а также их вариаций. Именно они обеспечивают или лучшие результаты, или хотя бы близкие к лучшим. Прочие эмбеддинги таких стабильных результатов не показывают.

Метод для сравнения и классификации векторных моделей — это или косинусная метрика с пороговым значением, подобранным вручную, или простая нейронная сеть. В нескольких случаях используются сиамские нейронные сети, которые предназначены для сравнения пар текстов.

Общая оценка качества анализа студенческих текстов дается с помощью достаточно разнообразных метрик. Помимо доли правильных ответов (accuracy в таблице), F-меры и среднеквадратичной ошибки, исследователи вычисляют QWK — квадратично-взвешенную каппу и коэффициент корреляции Пирсона. Значения метрик качества нельзя назвать высокими и стабильными: в каждом исследовании оказывается свой уровень или диапазон значений. Тем не менее большинство авторов успешно решают или свою основную задачу, или одну из подзадач. Они достигают высоких значений F-меры, доли правильных ответов и QWK и низких значений метрик, описывающих ошибки, либо на корпусе текстов в целом, либо на его части. Таким образом, для каждого предложенного метода находится своя область, где он оказывается полезен.

3. Комбинированные модели

Прямое применение больших языковых моделей обычно не даёт желаемого качества решения задачи, так как мало учитывает предметную область, а для дообучения модели недостаточно материалов, поскольку экспертная разметка текстов требует значительных временных и человеческих ресурсов. Один из путей решения этих проблем находится в комбинировании эмбеддингов и лингвистических и других характеристик, отражающих особенности проверки правильности ответов.

Tulu и др. [52] используют в методе оценки коротких ответов синонимы из WordNet, переведённые в эмбеддинги методом SemSpace, и модель MaLSTM, основанную на Манхэттенском расстоянии и позволяющую работать с парами эмбеддингов эталонных ответов и студенческих ответов. Эксперименты с корпусом Mohler из 87 ответов показали среднеквадратичную ошибку 0.04 и среднюю абсолютную ошибку (MAE) 0.23. Эксперименты с собственным корпусом CU-NLP из 171 ответа показали MAE, равную 0.02. Следует отметить, что хорошие результаты метод показал на авторском корпусе.

Zhang и др. [53] скомбинировали для оценки коротких ответов модели правильных ответов, ответа студента, вопроса и уровня знаний студента. Для моделирования пар «правильный ответ» — «студенческий ответ» использовался мешок слов, IDF, LSA, косинусная метрика, разница в количестве слов, метрика близости с учетом предметной области, общая метрика близости. Для модели-

рования вопроса определялась его тема и уровень сложности. Для моделирования уровня знаний студента применялась скрытая марковская модель с двумя состояниями, базирующаяся на предварительном тестировании студента по темам, которым посвящены вопросы. В качестве классификатора правильный/неправильный ответ использовалась нейронная сеть DBN. F-мера достигла 83 % на корпусе данных из системы Cordillera, состоящем из ответов 158 студентов на 482 вопроса.

Lubis и др. [54] скомбинировали эмбеддинги word2vec, обученные на индонезийской Википедии, и лингвистические характеристики, основанные на частях речи и синтаксической структуре предложения. На основе синтаксических связей между словами в коротком ответе эмбеддинги слов word2vec объединялись в эмбеддинг текста. Близость ответа к эталону измерялась с помощью косинусной метрики. Эксперименты на собственном корпусе индонезийских коротких ответов показали MAE, равную 0.70 .

Shashavali и др. [55] разбивали предложения на n-граммы и вычисляли для них эмбеддинги FastText. Эмбеддинги ответов сравнивались с эталонными по косинусной метрике. Эксперименты на собственном корпусе предложений различной длины показали F-меру, равную 93 % .

Prabhudesai и Duong [56] оценивают короткие ответы студентов при помощи комбинации эмбеддингов GloVe и стилометрических характеристик: количества слов, длины ответа, количества уникальных слов. Вектора характеристик строятся для студенческого и эталонного ответов и подаются на вход сиамской нейронной сети на основе LSTM. Эксперименты на корпусе Mohler позволили достичь MAE, равной 0.62, и среднеквадратичной ошибки, равной 0.89.

Lakshmi и др. [57] моделировали каждый короткий ответ как набор метрик близости между ним и эталоном. Метрики близости включали в себя статистическую близость по длинам текстов, количествам всех слов и уникальных слов, близость по общим словам, близость по ключевым, уникальным и лемматизированным словам, близость по мешку слов и TF-IDF, близость LSA, семантическую близость по эмбеддингам Infersent, близость по сгенерированным аннотациям ответов. Вектора из метрик близости классифицировались по оценкам с помощью стандартных методов машинного обучения: KNN, байесовский классификатор, SVM, дерево решений, случайный лес, XGBoost. Эксперименты проводились на корпусе текстов, составленном из известных открытых корпусов для оценки коротких текстов. Качество классификации достигло F-меры в среднем 75 % для метода случайного леса.

Сравнение текстов через ансамбль четырёх мер сходства осуществляют Hassan и др. [58]. Ансамбль включает в себя две характеристики близости на основе мешка слов и синонимов из BabelNet, эмбеддинги DSM, учитывающие контекст слов и характеристику близости по расстоянию Левенштейна. Эксперименты на собственном корпусе и на корпусах SemEval показывают коэффициент корреляции Пирсона 0.70–0.85.

Del Gobbo и др. [59] моделировали короткие ответы с помощью TF-IDF и эмбеддингов BERT. В процессе работы метод использует для измерения близости эталонных ответов и ответов студентов регрессор на основе метода опорных векторов (SVR). Для экспериментов были объединены открытые корпуса ASAP, SciEntBank и Cu-NLP, для корпуса ASAP ответы с лучшими оценками были размечены как эталонные, ответы оценивались по шкале о 0 до 5. Качество работы метода достигло среднеквадратичной ошибки в среднем 0.73 среди 83 вопросов.

Zhao и др. [60] построили модель анализа семантической целостности студенческих эссе на основе BERT. Авторы скомбинировали эту модель с характеристиками на основе ключевых моментов эссе: особенности стиля, содержание абзацев и темы абзацев. В качестве классификатора применялся ансамбль из сиамской нейронной сети и ESIM, который оценивал пары студенческое эссе — эталонное эссе. Эксперименты на собственном корпусе показали сильный разброс коэффициента каппы Коэна от 0.50 до 0.85 для 8 различных вопросов для эссе.

Table 3. Research where text answer analysis tasks are solved using combined models

Таблица 3. Исследования, где задачи анализа текстового ответа решаются с помощью комбинированных моделей

Авторы	Задача	Язык	Модель	Метод	Качество
Tulu и др. [52]	ASAG	Английский	WordNet, SemSpace MaLSTM	MaLSTM	RMSE 0.04 шкала 0–5 MAE 0.02–0.23
Zhang и др. [53]	ASAG	Английский	Стилометрия Метрики близости Скрытая марковская модель	DBN	F-мера 83 %
Lubis и др. [54]	ASAG	Индонезийский	word2vec Синтаксические характеристики	Косинусная метрика	MAE 0.70
Shashavali и др. [55]	Близость текстов	Английский	*n*-граммы FastText	Косинусная метрика	F-мера 93 %
Prabhudesai и др. [56]	ASAG	Английский	GloVe Стилометрия	Siamese LSTM	MAE 0.62 RMSE 0.89 шкала 0–5
Lakshmi и др. [57]	ASAG	Английский	Метрики близости	Случайный лес	F-мера 75 %
Hassan и др. [58]	Близость текстов	Английский Арабский	Эмбеддинги DSM Мешок слов BabelNet	Косинусная метрика	Пирсон 0.70–0.85
Del Gobbo и др. [59]	ASAG	Английский	TF-IDF, BERT	SVR	RMSE 0.73 шкала 0–5
Zhao и др. [60]	AES	Английский	BERT Стилометрия	Siamese NN ESIM	Каппа 0.50–0.85
Faseeh и др. [61]	AES	Английский	RoBERTa Лингвистические характеристики	Нейронная сеть	QWK 0.93 RMSE 0.18 шкала 0–5
Gagliardi и др. [62]	Близость текстов	Английский Французский	BERT, GPT WordNet	Косинусная метрика	F-мера 74 %
Beseiso и др. [63]	AES	Английский	BERT, word2vec Стилометрия	Нейронная сеть	Accuracy 75 % Каппа 0.77

ASAG — automatic short answer grading, автоматическая оценка кратких ответов.
AES — automated essay scoring, автоматизированная оценка эссе.
RMSE — root mean square error, среднеквадратичная ошибка.
QWK — quadratic weighted kappa, квадратично-взвешенная каппа.
MAE — mean absolute error, средняя абсолютная ошибка.

Faseeh и др. [61] применили гибридный подход для автоматической оценки эссе, скомбинировав языковую модель RoBERTa и лингвистические характеристики. Лингвистические характеристики включают в себя грамматические ошибки, близость между эталонным и студенческим эссе, рассчитанную как косинусную метрику между BERT-эмбеддингами, длины абзацев, тональность, частоту встречаемости частей речи, появление вспомогательных слов и словосочетаний, маркирующих структуру эссе, читабельность, богатство словаря. Эксперименты на корпусе ASAP показали QWK 0.93 и среднеквадратичную ошибку 0.18.

Авторы исследования [62] сопоставляют ключевые слова и фразы с использованием ансамблей эмбеддингов BERT, GPT, параметров связей на основе WordNet, алгоритма сравнения строк Джаро-Винклера. Эксперименты на мультиязычных корпусах с текстами на английском и французском языках показывают максимальное значение F-меры 74 %.

Beseiso и Alzahrani [63] оценивали эссе из корпуса ASAP. Они сконкатенировали три векторные репрезентации текстов: word2vec, BERT и стилометрические характеристики. Последние включали в себя количество именованных сущностей, слов, знаков пунктуации, частей речи, предложений,

а также значения метрик близости до 8 эссе, выбранных авторами вручную как образцовые. Доля правильных ответов достигла 75 %, каппа Коэна — 0.77.

Исследования, опирающиеся на комбинированные модели текстов, приведены в таблице 3. Таблица по структуре аналогична таблице из раздела 1.4.

Комбинированные модели встречаются существенно реже, чем только нейросетевые или только лингвистические. В основном они предназначены для английского языка.

В качестве эмбеддингов для моделирования текстов чаще берутся BERT и word2vec. Они конкатенируются с разнообразными лингвистическими моделями как на основе стандартных характеристик уровня слов, так и с более сложными характеристиками уровней структуры предложения или семантики.

Методы для сравнения и классификации комбинированных моделей совпадают с аналогичными методами, применяемыми для нейросетевых моделей.

Значения метрик качества достаточно хорошие. F-мера часто оказывается выше 74 %, среднеквадратичная ошибка и MAE в отдельных случаях близки к нулю.

Заключение

Анализ исследований в области оценки ответов учащихся на естественном языке показывает, что применяемые модели в целом соответствуют общей тенденции развития методов NLP. Среди стандартных лингвистических параметров популярны n-граммы, метрика TF-IDF, ключевые слова, синонимы. Эмбеддинги больших языковых моделей занимают главное место в работах последнего десятилетия. Однако не существует универсального подхода, обеспечивающего хорошее качество прогнозирования оценки ответа, даже в случаях бинарной классификации на правильные и неправильные. Явное преимущество показывают ансамблевые и комбинированные модели. Их комплексный подход позволяет учесть особенности поставленной задачи, но неизбежно ведёт к специализации разрабатываемого метода.

Можно отметить практически полное отсутствие исследований зависимости проверки текстов ответов от тематики задаваемых вопросов и их типов. В первую очередь это связано с отсутствием корпусов размеченных данных. Подавляющее большинство учёных формирует собственные корпуса, отсутствующие в открытом доступе. Такая ситуация сильно ограничивает объективность результатов. Интересно, что это характерно даже для английского языка.

Работы с английским языком сильно преобладают, несмотря на то, что хорошие языковые модели существуют для большинства естественных языков. Это открывает широкое поле для исследователей. Результаты автоматической проверки англоязычных ответов показывают перспективность разработки подобных национальных систем.

References

[1] R. Gao, H. E. Merzdorf, S. Anwar, M. C. Hipwell, and A. R. Srinivasa, “Automatic assessment of text-based responses in post-secondary education: A systematic review”, *Computers and Education: Artificial Intelligence*, vol. 6, p. 100 206, 2024. DOI: 10.1016/j.caeai.2024.100206..

[2] N. A. Medvedeva, N. G. Malkov, and M. L. Prozorova, “Professional and public accreditation as an assessment of agricultural educational program quality in Russia”, *Asian Journal of University Education*, vol. 17, no. 1, pp. 100–111, 2021. DOI: 10.24191/ajue.v17i1.12611.

[3] S. Bonthu, S. Rama Sree, and M. Krishna Prasad, “Automated short answer grading using deep learning: A survey”, in *Proceedings of the 5th International Cross-Domain Conference Machine Learning and Knowledge Extraction*, Springer, 2021, pp. 61–78. DOI: 10.1007/978-3-030-84060-0_5.

[4] D. Ramesh and S. K. Sanampudi, "An automated essay scoring systems: A systematic literature review", *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2495–2527, 2022. DOI: 10.1007/s10462-021-10068-2.

[5] S. Burrows, I. Gurevych, and B. Stein, "The eras and trends of automatic short answer grading", *International journal of artificial intelligence in education*, vol. 25, pp. 60–117, 2015. DOI: 10.1007/s40593-014-0026-8.

[6] L. Parra G and X. Calero S, "Automated writing evaluation tools in the improvement of the writing skill.", *International Journal of Instruction*, vol. 12, no. 2, pp. 209–226, 2019. DOI: 10.29333/iji.2019.12214a.

[7] A. Mizumoto and M. Eguchi, "Exploring the potential of using an ai language model for automated essay scoring", *Research Methods in Applied Linguistics*, vol. 2, no. 2, p. 100 050, 2023. DOI: 10.1016/j.rmal.2023.100050.

[8] S. Hsu, T. W. Li, Z. Zhang, M. Fowler, C. Zilles, and K. Karahalios, "Attitudes surrounding an imperfect AI autograder", in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–15. DOI: 10.1145/3411764.3445424.

[9] N. Süzen, A. N. Gorban, J. Levesley, and E. M. Mirkes, "Automatic short answer grading and feedback using text mining methods", *Procedia Computer Science*, vol. 169, pp. 726–743, 2020. DOI: 10.1016/j.procs.2020.02.171.

[10] A. Y. Kerbeneva, "Razrabotka procedur intellektual'noj ocenki znanij na osnove semanticheskoj obrabotki otvetov pol'zovatelej na estestvennom yazyke", in *Informacionnye sistemy i tekhnologii IST-2021*, in Russian, 2021, pp. 187–191.

[11] A. A. Poguda and J. Tape, "Development of an algorithm and module for automatic evaluation of student papers based on semantic analysis of text", *Open Education*, vol. 28, no. 3, pp. 46–55, 2024, in Russian. DOI: 10.21686/1818-4243-2024-3-46-55.

[12] F. S. Pribadi, A. E. Permanasari, and T. B. Adji, "Short answer scoring system using automatic reference answer generation and geometric average normalized-longest common subsequence (GAN-LCS)", *Education and Information Technologies*, vol. 23, pp. 2855–2866, 2018. DOI: 10.1007/s10639-018-9745-z.

[13] A.-B. Lekshmi-Narayanan and P. Brusilvosky, "Evaluating correctness of student code explanations: Challenges and solutions", in *Proceedings of 8th Educational Data Mining in Computer Science Education Workshop*, 2024, p. 1079.

[14] M. Meccawy, A. A. Bayazed, B. Al-Abdullah, and H. Algamdi, "Automatic essay scoring for Arabic short answer questions using text mining techniques", *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, 2023.

[15] H. A. Abdeljaber, "Automatic Arabic short answers scoring using longest common subsequence and Arabic WordNet", *IEEE Access*, vol. 9, pp. 76 433–76 445, 2021. DOI: 10.1109/ACCESS.2021.3082408.

[16] I. Buditjahjanto, M. Idhom, M. Munoto, and M. Samani, "An automated essay scoring based on neural networks to predict and classify competence of examinees in community academy.", *TEM Journal*, vol. 11, no. 4, 2022. DOI: 10.18421/TEM114-34.

[17] O. B. Mishunin, A. P. Savinov, and D. I. Firstov, "Sostoyanie i uroven' razrabotok sistem avtomaticheskoj ocenki svobodnyh otvetov na estestvennom yazyke", *Modern High Technologies*, no. 1, pp. 38–44, 2016, in Russian.

[18] N. S. Lagutina, K. V. Lagutina, and V. N. Kopnin, "Automatic determination of semantic similarity of student answers with the standard one using modern models", *Modeling and Analysis of Information Systems*, vol. 31, no. 2, pp. 194–205, 2024, in Russian. DOI: 10.18255/1818-1015-2024-2-194-205.

[19] A. V. Kryukova, "Computing semantic similarity of russian texts by means of DKPro similarity tool", in *Trudy ob"edinyonnoj nauchnoj konferencii ¡¡Internet i sovremennoe obshchestvo¿¿*, 2017, pp. 87–97. DOI: 10.17586/2541-9781-2017-1-87-97.

[20] D. Bär, T. Zesch, and I. Gurevych, "Dkpro similarity: An open source framework for text similarity", in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2013, pp. 121–126.

[21] V. Nandini and P. Uma Maheswari, "Automatic assessment of descriptive answers in online examination system using semantic relational features", *The Journal of Supercomputing*, vol. 76, no. 6, pp. 4430–4448, 2020. DOI: 10.1007/s11227-018-2381-y.

[22] A. G. Leonov, N. S. Martynov, K. A. Mashchenko, A. A. Kholkina, and A. V. Shlyakhov, "Automation of semantic analysis for textual responses of students in a digital educational platform", *Software & Systems*, vol. 37, no. 3, pp. 440–452, 2024, in Russian. DOI: 10.15827/0236-235X.142.440-452.

[23] D. D. Zafievsky, N. S. Lagutina, O. A. Melnikova, and A. Y. Poletaev, "Text model for the automatic scoring of business letter writing", *Automatic Control and Computer Sciences*, vol. 57, no. 7, pp. 828–840, 2023. DOI: 10.3103/S0146411623070167.

[24] N. A. Prokopyev, "Automatic grading of answers in knowledge control for «definition» and «description» question types", *Uchenye Zapiski Kazanskogo Universiteta. Seriya Fiziko-Matematicheskie Nauki*, vol. 166, no. 4, pp. 580–593, 2024, in Russian. DOI: 10.26907/2541-7746.2024.4.580-593.

[25] A. Khoroshilov, A. Kan, E. Evdokimova, and S. Pitskhelauri, "Establishing similarities between text documents", *Modelling and Data Analysis*, vol. 13, no. 4, pp. 45–58, 2023, in Russian. DOI: 10.17759/mda.2023130403.

[26] L. Ouahrani and D. Bennouar, "AR-ASAG an Arabic dataset for automatic short answer grading evaluation", in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2020, pp. 2634–2643.

[27] V. A. Kozhevnikov and O. Y. Sabinin, "System of automatic verification of answers to open questions in Russian", *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunications and Control Systems*, vol. 11, no. 3, pp. 57–72, 2018, in Russian. DOI: 10.18721/JCSTCS.11306.

[28] D. V. Gadasin, A. V. Shvedov, and I. S. Vakurin, "Opredelenie semanticheskoj blizosti tekstov s ispol'zovaniem algoritma sravneniya sushchnosti grafov", *REDS: Telekommunikacionnye ustrojstva i sistemy*, vol. 12, no. 4, pp. 11–19, 2022, in Russian.

[29] N. Maharjan and V. Rus, "A concept map based assessment of free student answers in tutorial dialogues", in *Artificial Intelligence in Education: 20th International Conference, AIED*, Springer, 2019, pp. 244–257. DOI: 10.1007/978-3-030-23204-7_21.

[30] I. Y. Kashirin, "Binarnye ierarhicheskie chisla dlya vychisleniya semanticheskoj blizosti predlozhenij estestvennogo yazyka", *Vestnik of RSREU*, no. 86, pp. 110–120, 2023, in Russian. DOI: 10.21667/1995-4565-2023-86-110-121.

[31] I. Y. Kashirin, "Ierarhicheskie chisla dlya proektirovaniya ICF-taksonomij iskusstvennogo intellekta", *Vestnik of RSREU*, no. 71, pp. 71–82, 2020, in Russian. DOI: 10.21667/1995-4565-2020-71-71-82.

[32] C. Minnegalieva, G. Sabitova, and A. Gayaliev, "Method of pre-assessment of students' answers based on the vector model of documents", *Russian Digital Libraries Journal*, vol. 26, no. 3, pp. 324–339, 2023, in Russian. DOI: 10.26907/1562-5419-2023-26-3-324-339.

[33] C. Minnegalieva, I. Kashapov, and O. Morozova, "Automated students' short answers grading using language models", *Russian Digital Libraries Journal*, vol. 27, no. 3, pp. 278–293, 2024, in Russian. DOI: 10.26907/1562-5419-2024-27-3-278-293.

[34] I. G. Ndukwe, C. E. Amadi, L. M. Nkomo, and B. K. Daniel, "Automatic grading system using Sentence-BERT network", in *Artificial Intelligence in Education*, 2020, pp. 224–227.

[35] M. Fateen and T. Mine, "In-context meta-learning vs. semantic score-based similarity: A comparative study in Arabic short answer grading", in *Proceedings of ArabicNLP 2023*, 2023, pp. 350–358. DOI: 10.18653/v1/2023.arabicnlp-1.28.

[36] S. K. Gaddipati, D. Nair, and P. G. Plöger, *Comparative evaluation of pretrained transfer learning models on automatic short answer grading*, 2020. arXiv: 2009.01303 [cs.CL].

[37] J. Schneider, R. Richner, and M. Riser, "Towards trustworthy autograding of short, multi-lingual, multi-type answers", *International Journal of Artificial Intelligence in Education*, vol. 33, no. 1, pp. 88–118, 2023.

[38] M. Hendre, P. Mukherjee, R. Preet, and M. Godse, "Efficacy of deep neural embeddings based semantic similarity in automatic essay evaluation", *International Journal of Computing and Digital Systems*, vol. 9, pp. 1–11, 2020.

[39] A. Doewes, A. Saxena, Y. Pei, and M. Pechenizkiy, "Individual fairness evaluation for automated essay scoring system", in *Proceedings of the 15th International Conference on Educational Data Mining*, 2022, pp. 206–216. DOI: 10.5281/zenodo.685315.

[40] Q. Wang, "The use of semantic similarity tools in automated content scoring of fact-based essays written by EFL learners", *Education and Information Technologies*, vol. 27, no. 9, pp. 13 021–13 049, 2022.

[41] B. F. Dhini, A. S. Girsang, U. U. Sufandi, and H. Kurniawati, "Automatic essay scoring for discussion forum in online learning based on semantic and keyword similarities", *Asian Association of Open Universities Journal*, vol. 18, no. 3, pp. 262–278, 2023. DOI: 10.1108/aaouj-02-2023-0027.

[42] S. V. Bogolepova and M. G. Zharkova, "Researching the potential of generative language models for essay evaluation and feedback provision", *Domestic and Foreign Pedagogy*, vol. 1, no. 5, pp. 123–137, 2024, in Russian. DOI: 10.24412/2224fi??0772fi??2024fi??101fi??123fi??137.

[43] M. N. Evstigneev, "Thematic control and criteria-based assessment of foreign language writing skills using artificial intelligence technologies", *Tambov University Review. Series: Humanities*, vol. 29, no. 4, pp. 913–926, 2024, in Russian. DOI: 10.20310/1810-0201-2024-29-4-913-926.

[44] V. N. Kopnin and N. S. Lagutina, "Opredelenie semanticheskogo skhodstva tekstov s ispol'zovaniem yazykovyh modelej na osnove transformerov", in *Matematicheskoe i informacionnoe modelirovanie : materialy Vserossijskoj konferencii molodyh uchenyh, Tyumen'*, in Russian, Tyumen' : TyumGU-Press, 2024, pp. 143–146.

[45] W. H. Gomaa and A. A. Fahmy, "Ans2vec: A scoring system for short answers", in *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019) 4*, Springer, 2020, pp. 586–595.

[46] R. Weegar and P. Idestam-Almquist, "Reducing workload in short answer grading using machine learning", *International Journal of Artificial Intelligence in Education*, vol. 34, no. 2, pp. 247–273, 2024. DOI: 10.1007/s40593-022-00322-1.

[47] Z. Li, Y. Tomar, and R. J. Passonneau, "A semantic feature-wise transformation relation network for automatic short answer grading", in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 6030–6040. DOI: 10.18653/v1/2021.emnlp-main.487.

[48] H. Tan, C. Wang, Q. Duan, Y. Lu, H. Zhang, and R. Li, "Automatic short answer grading by encoding student responses via a graph convolutional network", *Interactive Learning Environments*, vol. 31, no. 3, pp. 1636–1650, 2023. DOI: 10.1080/10494820.2020.1855207.

[49] J. Xie, K. Cai, L. Kong, J. Zhou, and W. Qu, "Automated essay scoring via pairwise contrastive regression", in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 2724–2733.

[50] J. Garg, J. Papreja, K. Apurva, and G. Jain, "Domain-specific hybrid BERT based system for automatic short answer grading", in *2022 2nd International Conference on Intelligent Technologies (CONIT)*, IEEE, 2022, pp. 1–6. DOI: 10.1109/CONIT55038.2022.9847754.

[51] D. Witschard, I. Jusufi, R. M. Martins, K. Kucher, and A. Kerren, "Interactive optimization of embedding-based text similarity calculations", *Information Visualization*, vol. 21, no. 4, pp. 335–353, 2022. DOI: 10.1177/14738716221114372.

[52] C. N. Tulu, O. Ozkaya, and U. Orhan, "Automatic short answer grading with SemSpace sense vectors and MaLSTM", *IEEE Access*, vol. 9, pp. 19 270–19 280, 2021.

[53] Y. Zhang, C. Lin, and M. Chi, "Going deeper: Automatic short-answer grading by combining student and question models", *User modeling and user-adapted interaction*, vol. 30, no. 1, pp. 51–80, 2020.

[54] F. F. Lubis, A. Putri, D. Waskita, T. Sulistyaningtyas, A. A. Arman, Y. Rosmansyah, *et al.*, "Automated short-answer grading using semantic similarity based on word embedding", *International Journal of Technology*, vol. 12, no. 3, pp. 571–581, 2021. DOI: 10.14716/ijtech.v12i3.4651.

[55] D. Shashavali *et al.*, "Sentence similarity techniques for short vs variable length text using word embeddings", *Computación y Sistemas*, vol. 23, no. 3, pp. 999–1004, 2019. DOI: 10.13053/cys-23-3-3273.

[56] A. Prabhudesai and T. N. Duong, "Automatic short answer grading using Siamese bidirectional LSTM based regression", in *Proceedings of the IEEE International Conference on Engineering, Technology and Education (TALE)*, IEEE, 2019, pp. 1–6. DOI: 10.1109/TALE48000.2019.9226026.

[57] P. S. Lakshmi, J. Simha, and R. Ranjan, "Empowering educators: Automated short answer grading with inconsistency check and feedback integration using machine learning", *SN Computer Science*, vol. 5, no. 5, p. 653, 2024. DOI: 10.1007/s42979-024-02954-7.

[58] B. Hassan, S. E. Abdelrahman, R. Bahgat, and I. Farag, "UESTS: An unsupervised ensemble semantic textual similarity method", *IEEE Access*, vol. 7, pp. 85 462–85 482, 2019. DOI: 10.1109/ACCESS.2019.2925006.

[59] E. Del Gobbo, A. Guarino, B. Cafarelli, and L. Grilli, "GradeAid: A framework for automatic short answers grading in educational contexts—design, implementation and evaluation", *Knowledge and Information Systems*, vol. 65, no. 10, pp. 4295–4334, 2023.

[60] J. Zhao, Y. Li, and W. Feng, "Investigating the validity and reliability of a comprehensive essay evaluation model of integrating manual feedback and intelligent assistance", *International Journal of Emerging Technologies in Learning*, vol. 18, no. 4, 2023. DOI: 10.3991/ijet.v18i04.38241.

[61] M. Faseeh *et al.*, "Hybrid approach to automated essay scoring: Integrating deep learning embeddings with handcrafted linguistic features for improved accuracy", *Mathematics*, vol. 12, no. 21, p. 3416, 2024. DOI: 10.3390/math12213416.

[62] I. Gagliardi and M. T. Artese, "Ensemble-based short text similarity: An easy approach for multilingual datasets using transformers and WordNet in real-world scenarios", *Big Data and Cognitive Computing*, vol. 7, no. 4, p. 158, 2023. DOI: 10.3390/bdcc7040158.

[63] M. Beseiso and S. Alzahrani, "An empirical analysis of BERT embedding for automated essay scoring", *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 10, 2020. DOI: 10.14569/ijacsa.2020.0111027.

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

ARTIFICIAL INTELLIGENCE

Comparison of Pre-trained Models for Domain-specific Entity Extraction from Student Report Documents

A. V. Melnikova[1], M. S. Vorobeva[1], A. V. Glazkova[1]

DOI: 10.18255/1818-1015-2025-1-66-79

[1]University of Tyumen, Tyumen, Russia

MSC2020: 68T50, 97B40
Research article
Full text in Russian

Received February 11, 2025
Revised February 21, 2025
Accepted February 26, 2025

The authors propose a methodology for extracting domain-specific entities from student report documents in Russian language using pre-trained transformer-based language models. Extracting domain-specific entities from student report documents is a relevant task since the obtained data can be used for various purposes, ranging from the formation of project teams to the personalization of learning pathways. Additionally, automating the document processing workflow reduces the labor costs associated with manual processing. As training material for training models, expert-annotated student report documents were used. These documents were created by students in information technology programs between 2019 and 2022 for project-based, practical disciplines, and theses. The domain-specific entity extraction task is approached as two subtasks: named entity recognition (NER) and annotated text generation. A comparative analysis was conducted among NER encoder-only models (ruBERT, ruRoBERTa), encoder-decoder models (ruT5, mBART), and decoder-only models (ruGPT, T-lite) for text generation. The effectiveness of the models was evaluated using the F1-score, along with an analysis of common errors. The highest F1-score on the test set was achieved by mBART (93.55 %). This model also showed the lowest error rate in domain-specific entity identification during text generation and annotation. The NER models demonstrated a lower tendency for errors but tended to extract domain-specific entities in a fragmented manner. The obtained results indicate the applicability of the examined models for solving the stated tasks, considering the specific requirements of the problem.

Keywords: domain-specific entities; digital footprint; information extraction; natural language processing; pre-trained language models

INFORMATION ABOUT THE AUTHORS

Melnikova, Antonina V.	ORCID iD: 0009-0006-1011-5225. E-mail: a.v.melnikova@utmn.ru Senior Lecturer of the Department of Software of the School of Computer Science
Vorobeva, Marina S.	ORCID iD: 0000-0002-1508-4089. E-mail: m.s.vorobeva@utmn.ru Head of the Department of Software of the School of Computer Science, Associate Professor, PhD
Glazkova, Anna V. (corresponding author)	ORCID iD: 0000-0001-8409-6457. E-mail: a.v.glazkova@utmn.ru Associate Professor of the Department of Software of the School of Computer Science, PhD

Funding: This study was supported by the Ministry of Science and Higher Education of the Russian Federation within the framework of a State assignment (FEWZ-2024-0052).

For citation: A. V. Melnikova, M. S. Vorobeva, and A. V. Glazkova, "Comparison of pre-trained models for domain-specific entity extraction from student report documents", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 66–79, 2025. DOI: 10.18255/1818-1015-2025-1-66-79.

© Melnikova A. V., Vorobeva M. S., Glazkova A. V., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

Сравнение предварительно обученных моделей для извлечения предметно-ориентированных сущностей из студенческих отчетных документов

А. В. Мельникова[1], М. С. Воробьева[1], А. В. Глазкова[1]

DOI: 10.18255/1818-1015-2025-1-66-79

[1]Тюменский государственный университет, Тюмень, Россия

УДК 004.912+378.1
Научная статья
Полный текст на русском языке

Получена 11 февраля 2025 г.
После доработки 21 февраля 2025 г.
Принята к публикации 26 февраля 2025 г.

Авторы предлагают методику извлечения предметно-ориентированных сущностей (ПОС) из русскоязычных текстов студенческих отчетных документов с использованием предварительно обученных языковых моделей на основе трансформеров. Извлечение ПОС из студенческих работ представляет собой актуальную задачу, так как полученные данные могут использоваться для различных целей — начиная от формирования проектных групп и заканчивая персонализацией учебных маршрутов, а также автоматизация процесса обработки документов снижает затраты труда на ручную обработку. В качестве материала для дообучения исследуемых моделей использовались размеченные экспертами отчетные документы студентов, обучающихся по направлениям информационных технологий и поступивших в период с 2019 по 2022 год, по проектным, практическим дисциплинам и выпускным квалификационным работам. Задача извлечения ПОС рассматривается как две задачи: идентификация именованных сущностей и генерация размеченного текста. Сравнительный анализ проводился между моделями, основанными исключительно на энкодерах (ruBERT, ruRoBERTa), предназначенными для извлечения именованных сущностей, и моделями, использующими как энкодеры, так и декодеры (ruT5, mBART), а также моделями, базирующимися только на декодерах (ruGPT, T-lite), применяемыми для генерации текста. Для оценки эффективности сравниваемых моделей использовалась F-мера, а также проведен анализ типичных ошибок. Наиболее высокие показатели по F-мере на тестовом наборе данных продемонстрировала модель mBART (93.55 %). Эта же модель показала наименьший уровень ошибок при идентификации ПОС во время генерации текста и разметки. Модели для извлечения именованных сущностей проявляют меньшую склонность к ошибкам, однако имеют тенденцию к фрагментарному выделению ПОС. Полученные результаты свидетельствуют о применимости рассматриваемых моделей для решения поставленных задач с учетом специфики предъявляемых требований.

Ключевые слова: предметно-ориентированные сущности; цифровой след; извлечение информации; обработка естественного языка; предварительно обученные языковые модели

ИНФОРМАЦИЯ ОБ АВТОРАХ

Мельникова, Антонина Владимировна	ORCID iD: 0009-0006-1011-5225. E-mail: a.v.melnikova@utmn.ru Старший преподаватель кафедры программного обеспечения Школы компьютерных наук
Воробьева, Марина Сергеевна	ORCID iD: 0000-0002-1508-4089. E-mail: m.s.vorobeva@utmn.ru Заведующий кафедрой программного обеспечения Школы компьютерных наук, доцент, канд. тех. наук
Глазкова, Анна Валерьевна (автор для корреспонденции)	ORCID iD: 0000-0001-8409-6457. E-mail: a.v.glazkova@utmn.ru Доцент кафедры программного обеспечения Школы компьютерных наук, канд. тех. наук

Финансирование: Исследование выполнено при поддержке Министерства науки и высшего образования Российской Федерации в рамках государственного задания (FEWZ-2024-0052).

Для цитирования: A. V. Melnikova, M. S. Vorobeva, and A. V. Glazkova, “Comparison of pre-trained models for domain-specific entity extraction from student report documents”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 66–79, 2025. DOI: 10.18255/1818-1015-2025-1-66-79.

© Мельникова А. В., Воробьева М. С., Глазкова А. В., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

При работе с текстовыми документами специалисту важно оценить степень погруженности исполнителя в конкретную область. В зависимости от специфики неструктурированных или полуструктурированных текстовых данных это могут быть элементы предметной области (термины, навыки, специализированные слова, ключевые фразы), встречающиеся в отчетах, вакансиях, описаниях проектов, портфолио, заявках на гранты и других документах [1].

Если рассматривать образовательные данные, на текущий момент в вузах о работах обучающихся накоплен большой объем информации, который все время увеличивается. В рамках учебной деятельности студенты в процессе выполнения заданий формируют свой цифровой след, включающий отчетные документы проектных практикумов, письменных практических заданий, промежуточных результатов по дисциплинам, практических подготовок, выпускных квалификационных работ [2]. Например, из студенческих работ можно получить базу знаний о том, что изучали и использовали студенты ИТ-направлений: технологии, языки программирования, библиотеки, фреймворки, алгоритмы, методы, подходы и др.

Систематизация таких данных очень важна для решения широкого спектра задач — от формирования проектных команд до персонализации образовательных траекторий. Анализ цифрового следа студентов может служить основой для рекомендаций по улучшению учебных планов и курсов, анализа востребованности навыков и знаний, мониторинга успеваемости и выявления направлений развития в соответствии актуальными требованиями индустрии.

При постоянном увеличении объема текстовой информации и работы с ней ручной сбор требует от сотрудников учебных заведений значительных ресурсов, автоматическое извлечение нужных данных становится все более востребованной задачей [3].

Инструменты обработки естественного языка позволяют автоматизировать процесс извлечения, исключая участие специалистов, работающих с текстовыми документами, к тому же можно проводить более глубокий анализ информации благодаря контексту. Большинство существующих методов в узкоспециализированных задачах, вроде извлечения из текстов навыков и компетенций, было проверено на англоязычных текстах, для анализа русскоязычного контента доступен лишь ограниченный набор инструментов и моделей.

В своем исследовании мы будем использовать термин «предметно-ориентированная сущность» (ПОС), который охватывает различные компоненты, связанные с конкретной областью знаний или исследований, отражая ее тематический или предметный контекст, и который включен в текст с учетом практического применения. Эта сущность может включать разнообразные элементы, такие как ключевые слова, фразы, именованные сущности, навыки и термины, которые способствуют более точной передаче информации и всестороннему описанию выбранных тем в корпусе документов.

Цель работы — проведение сравнительного анализа предварительно обученных моделей на основе трансформеров для задачи извлечения используемых предметно-ориентированных сущностей из студенческих отчетов. В частности, проведение дообучения и сравнение методов, основанных на энкодерах и предназначенных для распознавания именованных сущностей, и методов, основанных на энкодерах и декодерах или только декодерах, для генерации размеченных текстов на русском языке.

Для проведения исследования экспертами был размечен набор данных, полученный из 300 отчетных документов студентов, для сравнения качества и анализа ошибок выбранных моделей.

1. Обзор работ по тематике

В статье [4] дан небольшой обзор существующих методов извлечения терминов из текстов. Существуют несколько групп методов, основанных на правилах, на статистических признаках, на глубоком обучении.

Методы, основанные на правилах, используют регулярные выражения или лексико-грамматические шаблоны для выделения терминов. Правила составляются исследователями отдельно для специализированных корпусов [5]. Вторая группа методов использует статистические признаки для отбора терминов. К таковым методам относится, например, rutermextract, который вычисляет частотность использования отдельных слов или словосочетаний в текстах и на их основе определяет специфические для текстов слова, которые являются кандидатами в термины. Комбинированные методы, использующие для отбора терминов и шаблоны, и статистические показатели, показывают более высокий результат для извлечения терминов [6]. Методы глубокого обучения для извлечения терминов из русскоязычных текстов были применены в работе [7], где авторами предложена нейросетевая архитектура, использующая векторные представления текстов из мультиязычной модели BERT (mBERT) [8]. В работе [9]проводилось сравнение дообученных моделей, основанных на архитектуре Transformer, ruBERT [10] и mBERT.

Задача извлечения терминов является частным случаем задачи извлечения именованных сущностей [11, 12]. Для этой задачи зачастую используются языковые модели типа Transformer [13]. В последних работах сравниваются модели для извлечения именованных сущностей из русских текстов. В [14] лучший результат показала модель ruBERT (F-мера — 81.3 % на уровне сущностей). В работе [15] F-мера составила 68.82 % на уровне сущностей, данный результат был также получен моделью ruBERT, дополнительно обученной на текстах предметной области.

В последние годы также был опубликован ряд работ по извлечению упоминаний навыков из русскоязычных текстов вакансий. В статье [16] исследуется эффективность методов извлечения ключевых слов и модели BERT, дообученной для задачи распознавания именованных сущностей (named-entity recognition, NER). Авторами показано, что BERT значительно превосходит методы извлечения ключевых слов, однако демонстрирует достаточно низкое качество при извлечении навыков, отсутствующих в обучающей выборке. Работа [17] описывает подход на основе модели BERT и построения синтаксических деревьев. Автор подчеркивает, что в настоящее время перспективным инструментарием для решения рассматриваемой задачи являются предварительно обученные модели, основанные на архитектуре Transformer [18]. В работе [19] навыки извлекаются с помощью инструмента SkillNER. Поскольку данная модель предназначена для анализа англоязычных текстов, авторы предложили использовать инструменты машинного перевода русских текстов. Статья [20] исследует эффективность больших языковых моделей для извлечения навыков из русскоязычных текстов. Авторы сравнивают традиционные модели NER, основанные на энкодерах, с большими языковыми моделями, имеющими декодер-архитектуру. Результаты показали более высокую эффективность и более низкую вычислительную сложность моделей NER (F-мера — 81 %, доля правильно размеченных навыков — 73 %).

Другой близкой по тематике задачей является подбор ключевых слов. Большинство существующих исследований по подбору ключевых слов охватывает методы извлечения без учителя (в частности, статьи [21, 22]). В работах последних лет уделяется внимание генеративным методам подбора ключевых слов. В частности, в [23] показано, что мультиязычная энкодер-декодер модель способна генерировать ключевые слова в нормализованном виде и превосходить по метрикам (F-мера — 11.24 %, BERTScore — 76.89 %) традиционные подходы. В статье [24] сравниваются ключевые слова, подобранные с помощью ChatGPT, статистических подходов и подходов, основанных на машинном обучении. Результаты показывают низкую долю совпадений между полученными ключевыми словами. Однако наборы ключевых слов, составленные ChatGPT и экспертами, имеют относительно

высокую долю совпадений. Авторами работы [25] сравниваются модели генерации ключевых слов, основанные на энкодер-декодер и декодер-архитектуре. Лучший результат (F-мера — 20.16 %) был получен моделью, основанной на инструкциях, с помощью подхода с применением примеров.

2. Методы

2.1. Данные

Была собрана коллекция, включающая 300 отчетных документов по практическим подготовкам, проектным дисциплинам и выпускным квалификационным работам студентов, поступивших на ИТ-направление «Математическое обеспечение и администрирование информационных систем» в 2019–2022 годах. В каждом отчете экспертами были идентифицированы и аннотированы предметно-ориентированные сущности, относящиеся к предметной области ИТ-сферы, которые не просто упоминались, а использовались в процессе выполнения соответствующих работ. В текстах были представлены такие сущности, как названия языков программирования (Python, C++, C#), фреймворков и библиотек (Django, React, Next.js, Pandas, Sklearn), баз данных и технологий ORM (PostrgeSQL, MySQL, SQLAlchemy), инструментов для DevOps (Docker, Kubernetes, Nginx) и других групп инструментов разработки, названия алгоритмов (алгоритм k-ближайших соседей, алгоритм классификации, агломеративная кластеризация) и структур данных (граф, дерево, хэш-таблица). Для выделения начала и конца вхождения упоминания предметно-ориентированной сущности в тексте использовались теги [TAG*] и [*TAG] соответственно. Примеры размеченных текстов представлены в таблице 1.

Для последующего исследования из каждого студенческого отчета были извлечены абзацы — тексты (всего 2195), содержащие упоминания предметно-ориентированных сущностей. Всего выделено 1460 уникальных ПОС. Полученный датасет был разделен на обучающую и тестовую выборки в соотношении 70:30 (таблица 2). В тестовой выборке представлено 290 уникальных сущностей, что составляет 19 % от общего числа уникальных ПОС.

В большинстве случаев (72 %) в текстах присутствовало упоминание одной предметно-ориентированной сущности. Чаще всего упоминались сущности: «python» (4.9 % текстов), «база данных» (3.4 %), «javascript» (2.7 %) и «postgresql» (2.6 %). Остальные ПОС встречались меньше, чем в 2 % текстов. Подробные характеристики датасета визуализированы на рисунке 1, на котором отражены сущности, присутствующие не менее чем в 0.8 % текстов, и рисунке 2, демонстрирующем частоту появления в текстах определенного количества ПОС.

2.2. Модели

В данном исследовании были сравнены несколько подходов к извлечению предметно-ориентированных сущностей из студенческих отчетных документов. Поскольку анализ работ по тематике исследования показал эффективность моделей, основанных на архитектуре Transformer [18], для сравнения были выбраны предварительно обученные лингвистические модели, использующие эту архитектуру. В частности, были рассмотрены модели, основанные только на энкодере, на энкодере и декодере, только на декодере. Подробное описание используемых моделей представлено в таблице 3. Для обучения моделей использовались библиотеки SimpleTransformers[1] и Transformers [26].

- Модели, основанные только на энкодере:
  - ruBERT, русскоязычная версия модели BERT [8]. Предварительно обучена на русскоязычных текстах Википедии и новостей с помощью маскированного языкового моделирования (masked language modeling, MLM) с учетом контекста с обеих сторон, а также использует NSP (next sentence prediction), который определяет, следует ли одно предложение за другим, чтобы улучшить понимание связей между ними.

[1] https://simpletransformers.ai/

Table 1. Examles of texts in dataset **Таблица 1.** Примеры текстов в датасете

Описание ПОС	Размеченный текст
СУБД — понятие из области разработки баз данных PostgreSQL — название используемой СУБД	При выборе [TAG*]СУБД[*TAG] мы остановились на [TAG*]PostgreSQL[*TAG], так как каждый член нашей команды имел опыт работы с этой базой данных. Это облегчило командную работу и позволило использовать уже накопленные знания и навыки.
WPF — название системы, используемой в разработке	[TAG*]WPF[*TAG] — система для построения клиентских приложений Windows. С помощью данной системы был построен пользовательский интерфейс приложения.
FastHTTP — название фреймворка	[TAG*]FastHTTP[*TAG]: Высокопроизводительный HTTP сервер для Go. Использовался для обработки HTTP-запросов с минимальными задержками и высокой пропускной способностью [6].
HTML — название языка разметки CSS — название языка стилей JavaScript — название языка программирования	Клиентская часть веб-приложения, с которой осуществляет работу пользователь, использует [TAG*]HTML[*TAG] разметку для расположения элементов интерфейса, [TAG*]CSS[*TAG] для визуального представления страницы, [TAG*]JavaScript[*TAG] код для динамической работы со страницей и визуализацией. Именно с использованием данного языка происходит фильтрация таблицы данных с существующими элективными дисциплинами.
Алгоритм k-ближайших соседей — название используемого алгоритма k-NN — сокращенное название алгоритма	Полностью выполнить задачу классификации, опираясь лишь на правила, установленные заказчиком, не представляется возможным, из-за таких классов, как «шум», «тишина» и «музыка». Поэтому было принято решения использовать [TAG*]алгоритм k-ближайших соседей[*TAG] ([TAG*]k-NN[*TAG]), который был выбран в качестве первичной модели классификации аудиофайлов по нескольким причинам. Во-первых, это интуитивно понятный и простой в реализации метод. Во-вторых, он демонстрирует хорошие результаты на небольших объемах данных.
Стемминг и лемматизация — названия методов нормализации текстовых данных	Методы векторизации взяты из [4]. Из текста функций заранее удаляются знаки препинания, после этого он разбивается на токены (слова), если слово является стоп-словом (предлог, союз, местоимение и пр.) или содержит символы, не являющиеся буквами, то оно не используется. Разные формы одних и тех же слов приводятся к одной с помощью [TAG*]стемминга[*TAG] или [TAG*]лемматизации[*TAG]. Для стемминга используется стеммер, который отсекает от слов их части.

Table 2. Dataset statistics **Таблица 2.** Характеристики датасета

Характеристика	Обучающая выборка	Тестовая выборка
Количество текстов	1538	657
Средняя длина текстов (токенов)	28.04±19.54	27.32±19.59
Среднее количество предметно-ориентированных сущностей	1.51±1.07	1.48±0.98

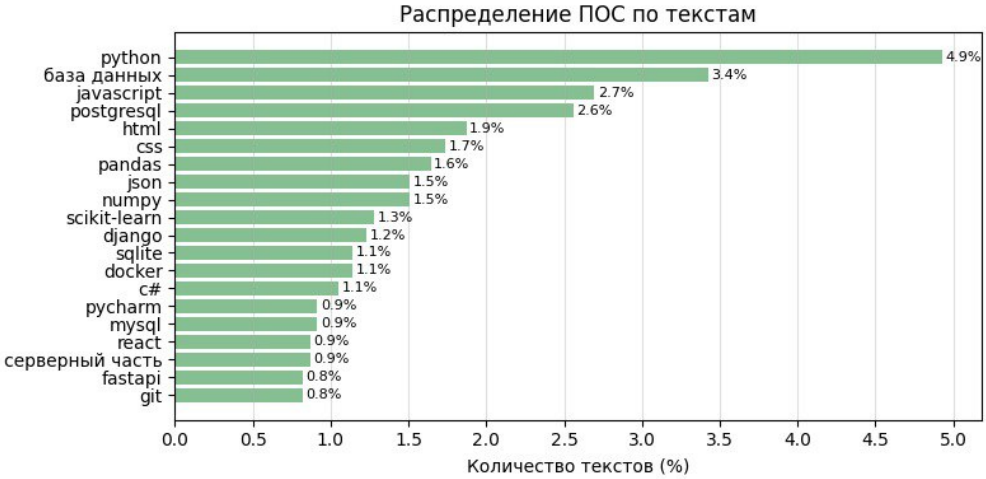

Fig. 1. Frequency of occurrence of object-oriented entities

Рис. 1. Частота встречаемости предметно-ориентированных сущностей

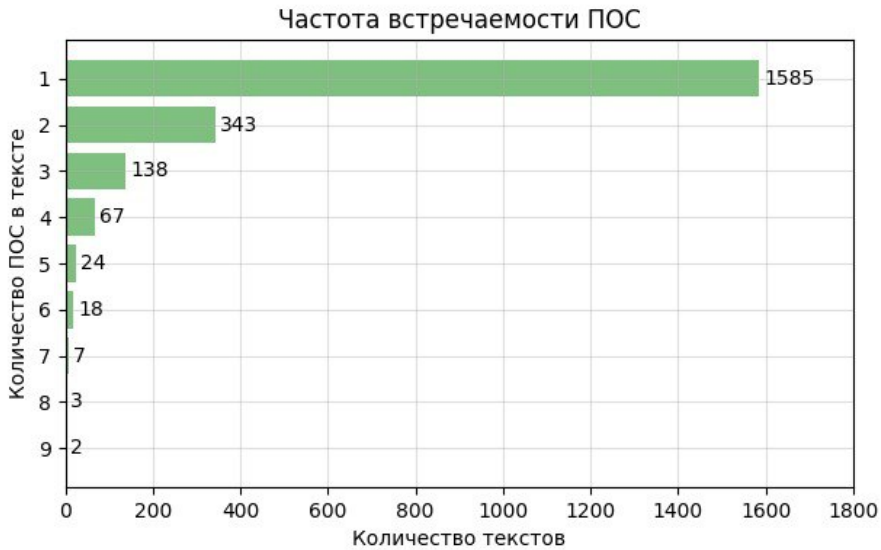

Fig. 2. Distribution of object-oriented entities across texts

Рис. 2. Распределение предметно-ориентированных сущностей по текстам

  - ruRoBERTa, улучшенная версия BERT с пересмотренным подходом к предобучению [27]. При обучении на русскоязычных текстах Википедии, новостей, книг, использовалось динамическое маскирование на каждой эпохе для больших батчей и более длинными последовательностями.
- Модели, основанные на энкодере и декодере:
  - ruT5, русскоязычная версия модели T5 [28] с архитектурой sequence-to-sequence. Предварительно обучена на русскоязычных текстах (книги, статьи, веб-страницы) для генерации текстов и использованием техники маскирования спангов (span corruption), где случайные фрагменты текста заменяются на маски, и модель учится их восстанавливать.
  - mBART, модель машинного перевода с архитектурой sequence-to-sequnce, построенная на базовой архитектуре BART [29]. Модель была предварительно обучена на более чем

Table 3. The models used in this study **Таблица 3.** Модели, используемые в работе

Модель	Версия	Архитектура	Количество параметров	Данные для обучения
ruBERT	DeepPavlov/rubert-base-cased [10]	Энкодер	178M	Википедия, новостные тексты
	ai-forever/ruBert-base [32]			Википедия, новостные тексты, Librusec, C4, OpenSubtitles
ruRoBERTa	ai-forever/ruRoberta-large [32]		355M	
ruT5	ai-forever/ruT5-base [32]	Энкодер-декодер	222M	
mBART	facebook/mbart-large-50 [33]		680M	Common Crawl (CC25), XLMR
ruGPT	ai-forever/rugpt3large_based_on_gpt2 [32]	Декодер	760M	Википедия, новостные тексты, Librusec, C4, OpenSubtitles
T-lite	t-bank-ai/T-lite-instruct-0.1		8B	Открытые англоязычные датасеты, переводы англоязычных датасетов, синтетический контекст для вопросно-ответных систем

50 языках с использованием комбинации техник маскирования фрагментов (span masking) и перемешивания предложений (sentence shuffling).

- Модели, основанные только на декодере:
  - ruGPT, русскоязычная модель, основанная на архитектуре GPT-3 [30] и использующая кодовую базу GPT-2 из библиотеки Transformers [26, 31].
  - T-lite, большая языковая модель, основанная на инструкциях, для обучения которой использовался набор данных, на 85 % состоящий из текстов на русском языке.

Модели, основанные на энкодере, обучались на задаче разметки последовательности токенов по аналогии с задачей распознавания именованных сущностей. ruBERT и ruRoBERTa были дообучены (fine-tuned) на обучающей выборке. При этом DeepPavlov/rubert-base-cased, ai-forever/ruBert-base, ruRoBERTa обучались в течение четырех, шести и пяти эпох соответственно. Количество эпох определялось экспериментальным путем: начиная с одной эпохи и постоянно увеличивая на одну эпоху, пока не будет достигнуто лучшее качество модели. Использовались следующие параметры для обучения: максимальная длина последовательности — 128 токенов, скорость обучения — 4e-5, размер батча — 16, оптимизатор — AdamW [34]. На вход каждой модели подавался текст, для каждого токена которого было необходимо предсказать одну из меток (B — токен является первым словом в ПОС, I — токен является не первым словом в ПОС, O — токен не является частью ПОС). Теги [TAG*] и [*TAG] были удалены перед процессом дообучения.

Поскольку модели ruT5 и mBART относятся к типу sequence-to-sequence, они обучались на задаче генерации последовательности токенов, а именно требовалось сгенерировать исходными текст с верно расставленными тегами [TAG*] и [*TAG]. Дообучение модели ruT5 проводилось в течение 50 эпох со следующими параметрами: 128 токенов, скорость обучения — 4e-5, размер батча — 16, оптимизатор — AdamW. Дообучение mBART на обучающей выборке проводилось в течение 20 эпох с использованием следующих параметров: максимальная длина последовательности — 256 токенов, скорость обучения — 4e-5, размер батча — 8, оптимизатор — AdamW. На вход моделей, основанных на архитектуре энкодер-декодер, подавался текст без тегов [TAG*] и [*TAG], сгенерированный текст

должен был представлять собой исходный текст с выделенными тегами [TAG*] и [*TAG], обозначающими вхождения предметно-ориентированных сущностей.

Особенностью процесса обучения моделей с декодер-архитектурой является то, что для заданного слова слои внимания могут получать доступ только к словам, расположенным перед ним в тексте [31]. Таким образом, обучение декодера сосредоточено на предсказании следующего слова в тексте. Языковая модель ruGPT была дообучена с использованием задачи причинного моделирования языка (causal language modeling, CLM) с максимальной длиной последовательности 1024 токена в течение десяти эпох. Количество эпох было установлено по аналогии с экспериментом по дообучению данной модели для одной из задач генерации текста на естественном языке, представленном в оригинальной статье [32]. В качестве остальных параметров дообучения взяты параметры, использующиеся в библиотеке SimpleTransformers по умолчанию. Входной текст подавался в следующем формате: *текст без тегов [TAG*] и [*TAG] + </s> + текст с выделенными тегами [TAG*] и [*TAG] + </end>*. Текст тестовой выборки подавался в виде: *текст без тегов [TAG*] и [*TAG] + </s>*.

Модель T-lite, которая относится к типу моделей, основанных на инструкциях, была обучена в технике обучения при помощи запросов (prompt-based learning) и подхода с применением примеров (few-shot learning). В качестве примеров были взяты десять случайных текстов из обучающей выборки. Для генерации текстов с выделенными тегами были установлены следующие параметры: температура, равная 0.1, и максимальная длина сгенерированного текста, равная 256 токенам. Выбор низкого значения параметра температуры обусловлен необходимостью модели повторять исходный текст с добавлением тегов и избегать перефразирования [35].

2.3. Метрики

Оценка результатов моделей проводилась с использованием двух метрик: посимвольной F-меры и F-меры на уровне сущностей. Использование метрик на уровне сущностей широко распространено в задаче извлечения именованных сущностей (в частности, в упомянутых выше работах [14, 15]). Авторы обзора [36] указывают, что помимо метрик, оценивающих точное совпадение сущностей, для оценки качества моделей также используют метрики, характеризующие частичное совпадение. Кроме того, посимвольные метрики широко используются в задачах, связанных с выделением фрагментов в тексте на естественном языке (например, в статье [37]).

Для оценки результатов каждый текст из тестовой выборки был разбит на токены (в зависимости от выбранной метрики — по словам или по символам). Также были разбиты соответствующие тексты, размеченные моделью. Для каждого текста в тестовом наборе рассчитывалось количество токенов, которые были верно предсказаны моделью. Это число использовалось для вычисления точности и полноты.

Точность (precision) и полнота (recall) вычислялись следующим образом:

$$Precision = \frac{TruePredicted}{PredictedTokens},$$

где *TruePredicted* — количество верно предсказанных токенов, *PredictedTokens* — общее количество токенов в тексте, размеченном моделью.

$$Recall = \frac{TruePredicted}{TestTokens},$$

где *TruePredicted* — количество верно предсказанных токенов, *TestTokens* — общее количество токенов в тексте из тестовой выборки.

Точность и полнота вычислялись для каждой пары текстов (текста из тестового набора и соответствующего текста, размеченного моделью). Затем средние значения точности и полноты использовались для вычисления F-меры:

Table 4. Results, %

Таблица 4. Результаты, %

Модель	Метрики на уровне сущностей			Посимвольные метрики		
	F1	Precision	Recall	F1	Precision	Recall
Модели, основанные только на энкодере						
ruBERT (DeepPavlov/rubert-base-cased)	72.48	73.43	71.56	67.14	70.16	64.36
ruBERT (ai-forever/ruBert-base)	73.43	73.39	73.47	68.42	69.15	67.70
ruRoBERTa	73.56	73.60	73.52	70.36	68.28	72.57
Модели, основанные на энкодере и декодере						
ruT5	84.58	86.48	82.76	57.21	57.21	56.34
mBART	93.55	93.53	93.56	70.88	70.84	70.92
Модели, основанные только на декодере						
ruGPT	87.32	87.31	87.34	50.35	50.70	50.00
T-lite	86.96	86.98	86.95	51.26	51.08	51.44

$$F1 = \frac{2 \cdot AvgPrecision \cdot AvgRecall}{AvgPrecision + AvgRecall},$$

где *AvgPrecision* — среднее значение точности по каждой паре текстов, *AvgRecall* — среднее значение полноты по каждой паре текстов.

Использование двух способов расчета F-меры позволило оценить результаты как с точки зрения полного совпадения вручную размеченных предметно-ориентированных сущностей и сущностей, выделенных моделью, так и с точки зрения их частичного совпадения на уровне посимвольного пересечения.

3. Результаты

Результаты сравнения моделей представлены в таблице 4. Полученные значения метрик показывают, что модель mBART в большинстве случаев демонстрирует лучшее качество извлечения предметно-ориентированных сущностей из студенческих отчетных документов. Наибольшее преимущество mBART демонстрирует в значениях метрик, рассчитанных на уровне сущностей. Это показывает превосходство mBART с точки зрения полного совпадения вручную размеченных и выделенных моделью предметно-ориентированных сущностей. Основываясь на результатах посимвольных метрик, можно сделать вывод о близком качестве моделей mBART, ruRoBERTa и ruBERT.

Модель mBART показывает лучшее значение посимвольной F-меры (70.88 %) и Precision (70.84 %), при этом второе по величине значение F-меры демонстрирует ruRoBERTa (70.36 %), значение Precision — DeepPavlov/rubert-base-cased (70.16 %). Лучшее значение посимвольной метрики Recall показывает ruRoBERTa (72.57 %), превосходя mBART (70.92 %) на 1.65 процентных пункта.

Был проведен анализ ошибок, которые совершали модели. Результаты представлены в таблице 5. Для проведения анализа было осуществлено сопоставление сущностей каждого текста тестового набора с сущностями, идентифицированными моделями в указанных текстах. Неправильными сущностями текста являются те, которые выделила модель, но в изначальной разметке их не было. Оценивалось также, насколько много было пропущено ПОС из размеченных текстов, а для генеративных моделей способность размечать текста в соответствии с форматом.

Среди моделей, предназначенных для генерации текста с разметкой в формате [TAG*] и [*TAG], наименьшее количество ошибок при идентификации предметно-ориентированных сущностей демонстрирует модель mBART, которая также характеризуется меньшим количеством ошибок при расстановке тегов. Для других моделей характерно появление случаев, когда в результирующий текст встраивается значительное число открывающих тегов [TAG*], не имеющих соответствующих закрывающих пар (до 10–20 несбалансированных тегов в некоторых примерах).

Table 5. Errors of the models **Таблица 5.** Ошибки моделей

Модель	Среднее количество неправильно выделенных сущностей в тексте	Среднее количество сущностей, которые модель не смогла выделить в тексте	Среднее количество непарных меток [TAG*] и [*TAG]
DeepPavlov/rubert-base-cased	0.45±0.69	0.42±0.68	—
ai-forever/ruBert-base	0.47±0.73	0.44±0.69	—
ruRoBERTa	0.47±0.69	0.45±0.68	—
ruT5	0.64±0.82	0.75±0.96	0.10±0.31
mBART	0.53±0.85	0.58±0.85	0.11±0.63
ruGPT	0.56±0.93	0.86±0.93	0.22±1.32
T-lite	0.98±1.27	0.84±0.86	0.19±1.04

Модель T-lite выделяет множество элементов текста, которые не соответствуют предметно-ориентированным сущностям, включая числовые значения, подписи к иллюстрациям и таблицам, а также общие лексические единицы вроде «студент», «сотрудник», «сделка». Модель ruGPT, в свою очередь, идентифицирует английские слова и выражения, являющиеся наименованиями объектов или функций в библиотеках программного обеспечения. Наблюдаются ситуации, когда происходит выделение слов или выражений, относящихся к сфере информационных технологий, однако они оказываются усеченными или с заменой отдельных букв (например, «ransformers» — в тексте описывается использование библиотеки transformers, но модель выделила ее как «t[TAG*]ransformers[*TAG]»). Модель ruT5 зачастую искажает выделенные слова или фразы (например, преобразуя «Jupiter Notebook» в «Jeeppiter Notebook», а «postgresql» в «pargresql») и усекает отдельные предметно-ориентированные сущности, подобно модели ruGPT. Модель mBART в основном допускает ошибки при распознавании наименований алгоритмов и методов, ограничиваясь идентификацией имен их разработчиков.

Модели, предназначенные для извлечения именованных сущностей, демонстрируют меньшую частоту ошибок при идентификации предметно-ориентированных сущностей благодаря сохранению всех слов оригинального текста в неизмененном виде. Тем не менее, они склонны выделять лишь фрагменты сущностей (например, «база» вместо «база данных», «бинарная» вместо «бинарная классификация»), а лучше всего справляются с сущностями, состоящими из одного слова.

Часть ПОС (290), идентифицированных в текстах тестового набора, отсутствовали в обучающей выборке. Для оценки способности моделей к обобщению был проведен анализ точности извлечения ранее неизвестных сущностей. Худшие результаты продемонстрировали модели генерации текста T-lite, ruGPT и ruT5, корректно распознавшие лишь 34–38 % новых сущностей. Модель mBART показала более высокую точность — 52 %. Модели ruBERT и ruRoBERTa немного превзошли предыдущие модели, выявив правильный процент новых ПОС в диапазоне от 56 до 60 %.

Заключение

В ходе проведенного исследования была выполнена оценка качества моделей для генерации текстов, основанных на энкодере и декодере (mBART, T5) и только на декодере (T-lite, ruGPT), и моделей для извлечения именованных сущностей, основанных только на энкодере (ruBERT, ruRoBERTa), в контексте идентификации используемых предметно-ориентированных сущностей в текстах отчетных документов студентов, обучающихся на ИТ-направлении. Модели были дополнительно обучены на корпусе текстов, размеченном экспертами, что обеспечило их адаптацию к специфическим характеристикам данной предметной области.

Максимальные показатели по метрике F-меры были достигнуты генеративной моделью mBART (93.55 %), что свидетельствует о высокой точности этой модели в идентификации ПОС в тексте. Тем не менее, существенным недостатком генеративного подхода является наличие артефактов в виде орфографических ошибок, неправильных вставок меток и фрагментации слов. В данном отношении модели для извлечения именованных сущностей проявили себя как более надежные решения, поскольку они обеспечивают сохранение целостности исходных данных, сводя к минимуму вероятность возникновения ошибок форматирования. Вместе с тем, NER-модели часто извлекают лишь частичные фрагменты сущностей.

Полученные результаты свидетельствуют о применимости обеих групп моделей для обработки отчетных документов. Выбор между ними должен осуществляться исходя из конкретных требований. Если первоочередное значение имеет высокая точность сохранения оригинального текста и способность к идентификации новых, ранее не встречавшихся ПОС, предпочтительнее использовать модели для извлечения именованных сущностей. Если же основным критерием выступает максимальная полнота выявления сущностей, несмотря на возможные искажения, целесообразнее выбрать модель mBART.

С целью улучшения качества выделения ПОС в текстах работ студентов необходимо дообучать модели дополнительными типами отчетных документов (отзывы, характеристики, дневники практики, технологические карты студенческих проектов), что позволит расширить возможности моделей и улучшить точность извлечения ключевых фрагментов текста.

Результаты извлечения ПОС можно применить для создания цифрового портрета студента, проверки студенческих отчетов, оценки текстовых характеристик, сравнительного анализа документов, анализа учебных материалов и разработки ИТ-онтологий, используемых в образовательном процессе.

References

[1] Q. Guohao, W. Bin, W. Bai, and Z. Baoli, “Competency analysis in human resources using text classification based on deep neural network”, in *Proceedings of the IEEE Fourth International Conference on Data Science in Cyberspace*, 2019, pp. 322–329. DOI: 10.1109/DSC.2019.00056.

[2] I. Zakharova, Y. V. Boganyuk, M. Vorobyova, and E. Pavlova, “Diagnostics of professional competence of IT students based on digital footprint data”, *Informatics and Education*, vol. 4, no. 313, pp. 4–11, 2020. DOI: 10.32517/0234-0453-2020-35-4-4-11.

[3] Z. Alami Merrouni, B. Frikh, and B. Ouhbi, “Automatic keyphrase extraction: A survey and trends”, *Journal of Intelligent Information Systems*, vol. 54, no. 2, pp. 391–424, 2020. DOI: 10.1007/s10844-019-00558-9.

[4] E. Bruches, A. Pauls, T. Batura, V. Isachenko, and D. Shcherbatov, “Semantic analysis of scientific texts: Experience in creating a corpus and building language models”, *Software & Systems*, vol. 34, no. 1, pp. 132–144, 2021. DOI: 10.15827/0236-235X.133.132-144.

[5] Y. I. Butenko, N. S. Nikolaeva, and T. D. Margaryan, “Structural models of terminological word combinations for marking up a corpus of scientific and technical texts”, *NSU Vestnik. Series: Linguistics and Intercultural Communication*, vol. 19, no. 3, pp. 45–56, 2021. DOI: 10.25205/1818-7935-2021-19-3-45-56.

[6] A. Novikova, “Comparison of Sketch Engine and TermoStat tools for terminology extraction”, *International Journal of Open Information Technologies*, vol. 8, no. 11, pp. 73–79, 2020.

[7] E. Bruches and T. Batura, “Method for automatic term extraction from scientific articles based on weak supervision”, *Vestnik NSU. Series: Information Technologies*, vol. 19, no. 2, pp. 5–16, 2021, in Russian. DOI: 10.25205/1818-7900-2021-19-2-5-16.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding", in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[9] Y. Dementyeva, E. Bruches, and T. Batura, "Terms extraction from texts of scientific papers", *Software & Systems*, vol. 35, no. 4, pp. 689–697, 2022. DOI: 10.15827/0236-235X.140.689-697.

[10] Y. Kuratov and M. Arkhipov, "Adaptation of deep bidirectional multilingual transformers for Russian language", in *Komp'juternaja Lingvistika i Intellektual'nye Tehnologii*, 2019, pp. 333–339.

[11] V. K. Pimeshkov, M. L. Nikonorova, and M. G. Shishaev, "A combined term extraction method for the problem of monitoring thematic discussions in social media", *Informatics and Automation*, vol. 23, no. 4, pp. 1110–1138, 2024. DOI: 10.15622/ia.23.4.7.

[12] M. D. Averina and O. A. Levanova, "Extracting named entities from Russian-language documents with different expressiveness of structure", *Modeling and Analysis of Information Systems*, vol. 30, no. 4, pp. 382–393, 2023, in Russian. DOI: 10.18255/1818-1015-2023-4-382-393.

[13] X. Liu, J. A. Erkoyuncu, J. Y. H. Fuh, W. F. Lu, and B. Li, "Knowledge extraction for additive manufacturing process via named entity recognition with LLMs", *Robotics and Computer-Integrated Manufacturing*, vol. 93, p. 102 900, 2025. DOI: 10.1016/j.rcim.2024.102900.

[14] T. Atnashev *et al.*, "Razmecheno: Named entity recognition from digital archive of diaries "Prozhito"", in *Proceedings of the Fifth International Conference on Computational Linguistics in Bulgaria (CLIB 2022)*, 2022, pp. 22–38.

[15] M. Tikhomirov, N. Loukachevitch, A. Sirotina, and B. Dobrov, "Using BERT and augmentation in named entity recognition for cybersecurity domain", in *Proceedings of the 25th International Conference on Applications of Natural Language to Information Systems*, 2020, pp. 16–24. DOI: 10.1007/978-3-030-51310-8_2.

[16] P. V. Korytov, Y. Y. Gribetskiy, E. A. Andreeva, and I. I. Kholod, "Analysis of approaches for identifying key skills in vacancies", in *Proceedings of the International Conference on Soft Computing and Measurement*, 2024, pp. 300–303. DOI: 10.1109/SCM62608.2024.10554269.

[17] I. E. Nikolaev, "Knowledge and skills extraction from the job requirements texts", *Ontology of Designing*, vol. 13, no. 2, pp. 282–293, 2023, in Russian. DOI: 10.18287/2223-9537-2023-13-2-282-293.

[18] A. Vaswani *et al.*, "Attention is all you need", *Advances in neural information processing systems*, vol. 30, pp. 1–11, 2017.

[19] A. V. Melnikova, M. S. Vorobeva, E. V. Egorova, and E. D. Chekanova, "Development of an algorithm for the formation of IT project teams based on data from the digital footprint of students", *Proceedings of the Institute for System Programming of the RAS*, vol. 36, no. 3, pp. 213–224, 2024. DOI: 10.15514/ispras-2024-36(3)-15.

[20] N. Matkin *et al.*, *Comparative analysis of encoder-based NER and large language models for skill extraction from Russian job vacancies*, 2024. arXiv: 2407.19816 [cs.CL].

[21] M. Khokhlova and M. Koryshev, "Keyness analysis and its representation in Russian academic papers on computational linguistics: Evaluation of algorithms", in *RASLAN*, 2022, pp. 25–33.

[22] O. Mitrofanova and D. Gavrilic, "Experiments on automatic keyphrase extraction in stylistically heterogeneous corpus of Russian texts", *Terra Linguistica*, vol. 13, no. 4, pp. 22–40, 2022. DOI: 10.18721/JHSS.13402.

[23] A. V. Glazkova, D. A. Morozov, M. S. Vorobeva, and A. A. Stupnikov, "Keyword generation for Russian-language scientific texts using the mT5 model", *Automatic Control and Computer Sciences*, vol. 58, no. 7, pp. 995–1002, 2024. DOI: 10.3103/S014641162470041X.

[24] D. Guseva and O. Mitrofanova, "Keyphrases in Russian-language popular science texts: comparison of oral and written speech perception with the results of automatic analysis", *Terra Linguistica*, vol. 15, no. 1, pp. 20–35, 2024. DOI: 10.18721/JHSS.15102.

[25] A. Glazkova, D. Morozov, and T. Garipov, *Key algorithms for keyphrase generation: Instruction-based LLMs for Russian scientific keyphrases*, 2024. arXiv: 2410.18040 [cs.CL].

[26] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing", Association for Computational Linguistics, 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

[27] Y. Liu *et al.*, *RoBERTa: A robustly optimized BERT pretraining approach*, 2019. arXiv: 1907.11692 [cs.CL].

[28] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer", *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020. DOI: 10.5555/3455716.3455856.

[29] M. Lewis *et al.*, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension", in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703.

[30] T. Brown *et al.*, "Language models are few-shot learners", *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020. DOI: 10.5555/3495724.3495883.

[31] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners", *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[32] D. Zmitrovich *et al.*, "A family of pretrained transformer language models for Russian", in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, 2024, pp. 507–524.

[33] Y. Tang *et al.*, "Multilingual translation from denoising pre-training", in *Findings of the Association for Computational Linguistics*, 2021, pp. 3450–3466. DOI: 10.18653/v1/2021.findings-acl.304.

[34] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization", in *International Conference on Learning Representations*, 2019, p. 53 592 270.

[35] A. Kartelj, M. Mladenović, and S. Vujičić Stanković, "Comparison of algorithms for the recognition of ChatGPT paraphrased texts", *Journal of Big Data*, vol. 12, no. 1, pp. 1–17, 2025. DOI: 10.1186/s40537-025-01082-0.

[36] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition", *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 50–70, 2022. DOI: 10.1109/TKDE.2020.2981314.

[37] G. Da San Martino, S. Yu, A. Barrón-Cedeño, R. Petrov, and P. Nakov, "Fine-grained analysis of propaganda in news articles", in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2019, pp. 5636–5646. DOI: 10.18653/v1/D19-1565.

MODELING AND ANALYSIS OF INFORMATION SYSTEMS, VOL. 32, NO. 1, 2025
journal homepage: www.mais-journal.ru

ARTIFICIAL INTELLIGENCE

Hierarchical Classification of Scientific Articles Using Deep Learning (Using the UDC Hierarchy as an Example)

V. Y. Mamedov[1], D. A. Kovalevsky[1], D. A. Morozov[1], S. S. Stolyarov[1], S. S. Ospichev[1]

DOI: 10.18255/1818-1015-2025-1-80-94

[1]Novosibirsk National Research State University, Novosibirsk, Russia

MSC2020: 68T50
Research article
Full text in Russian

Received February 14, 2025
Revised February 24, 2025
Accepted February 26, 2025

The exponential growth in scientific publications has heightened the need for robust tools to organize and retrieve research effectively. The Universal Decimal Classification (UDC) serves as a valuable framework for categorizing articles by subject area. However, manual assignment of UDC codes is often prone to inaccuracies or oversimplification, limiting its utility. In this study, we present a novel approach for the automated assignment of UDC codes to scientific articles using BERT-based models. Our methodology was trained and evaluated on a dataset comprising over 19,000 articles in mathematics and related disciplines. To address the hierarchical structure of UDC, we developed two specialized evaluation metrics: hierarchical classification accuracy and hierarchical recommendation accuracy. We also explored multiple strategies for flattening hierarchical labels. Our results demonstrated a hierarchical recommendation accuracy of 0.8220. Furthermore, blind expert evaluation revealed that discrepancies between reference and predicted labels often stem from errors in the original UDC code assignments by article authors. Our approach demonstrates strong potential for automating the classification of scientific articles and can be extended to other hierarchical classification systems.

Keywords: text classification; hierarchical text classification; universal decimal classifier; deep learning

INFORMATION ABOUT THE AUTHORS

Mamedov, Valentin Y.	ORCID iD: 0009-0004-4154-5522. E-mail: v.mamedov@g.nsu.ru PhD student
Kovalevsky, Danil A.	ORCID iD: 0009-0002-8484-7366. E-mail: d.kovalevskii@g.nsu.ru Master student
Morozov, Dmitry A. (corresponding author)	ORCID iD: 0000-0003-4464-1355. E-mail: morozowdm@gmail.com Junior Researcher
Stolyarov, Stepan S.	ORCID iD: 0009-0005-7651-6948. E-mail: s.stolyarov@g.nsu.ru Junior Researcher
Ospichev, Sergey S.	ORCID iD: 0000-0001-9912-6364. E-mail: s.ospichev@nsu.ru Deputy Director of the Mathematical Center in Akademgorodok, PhD

For citation: V. Y. Mamedov, D. A. Kovalevsky, D. A. Morozov, S. S. Stolyarov, and S. S. Ospichev, "Hierarchical classification of scientific articles using deep learning (using the UDC hierarchy as an example)", *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 80–94, 2025. DOI: 10.18255/1818-1015-2025-1-80-94.

© Mamedov V. Y., Kovalevsky D. A., Morozov D. A., Stolyarov S. S., Ospichev S. S., 2025
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ, ТОМ 32, № 1, 2025
сайт журнала: www.mais-journal.ru

Иерархическая классификация научных статей при помощи глубокого обучения (на примере иерархии УДК)

В. Ю. Мамедов[1], Д. А. Ковалевский[1], Д. А. Морозов[1], С. С. Столяров[1], С. С. Оспичев[1]

DOI: 10.18255/1818-1015-2025-1-80-94

[1]Новосибирский национальный исследовательский государственный университет, Новосибирск, Россия

УДК 004.912
Научная статья
Полный текст на русском языке

Получена 14 февраля 2025 г.
После доработки 24 февраля 2025 г.
Принята к публикации 26 февраля 2025 г.

В условиях стремительного роста числа научных публикаций актуальной задачей становится разработка эффективных инструментов для их систематизации и поиска. Одним из таких инструментов является универсальная десятичная классификация (УДК), которая позволяет структурировать статьи по тематическим областям. Однако ручное присвоение кодов УДК зачастую оказывается неточным или недостаточно детализированным, что снижает эффективность использования этого подхода. В данной статье предлагается подход к автоматическому присвоению кодов УДК научным статьям с использованием моделей на основе архитектуры BERT. Для обучения и оценки модели был использован набор данных, содержащий более 19 тысяч статей по математике и смежным наукам. Мы разработали две специализированные метрики качества, учитывающие иерархическую природу УДК: иерархическую классификационную точность и иерархическую рекомендательную точность. Кроме того, мы предложили несколько стратегий преобразования иерархических меток в плоские. В ходе экспериментов нам удалось достичь значения иерархической рекомендательной точности 0,8220. Дополнительно проведено слепое тестирование с участием экспертов, которое выявило, что часть расхождений между эталонными и сгенерированными метками обусловлена некорректным выбором кода УДК авторами статей. Предложенный подход демонстрирует высокий потенциал для автоматической классификации научных статей и может быть адаптирован для других иерархических систем классификации.

Ключевые слова: классификация текстов; иерархическая классификация текстов; универсальный десятичный классификатор; глубокое обучение

ИНФОРМАЦИЯ ОБ АВТОРАХ

Мамедов, Валентин Юрьевич	ORCID iD: 0009-0004-4154-5522. E-mail: v.mamedov@g.nsu.ru Аспирант
Ковалевский, Данил Анатольевич	ORCID iD: 0009-0002-8484-7366. E-mail: d.kovalevskii@g.nsu.ru Магистрант
Морозов, Дмитрий Алексеевич (автор для корреспонденции)	ORCID iD: 0000-0003-4464-1355. E-mail: morozowdm@gmail.com Младший научный сотрудник
Столяров, Степан Сергеевич	ORCID iD: 0009-0005-7651-6948. E-mail: s.stolyarov@g.nsu.ru Младший научный сотрудник
Оспичев, Сергей Сергеевич	ORCID iD: 0000-0001-9912-6364. E-mail: s.ospichev@nsu.ru Заместитель директора Международного Математического Центра, канд. физ.-мат. наук

Для цитирования: V. Y. Mamedov, D. A. Kovalevsky, D. A. Morozov, S. S. Stolyarov, and S. S. Ospichev, “Hierarchical classification of scientific articles using deep learning (using the UDC hierarchy as an example)”, *Modeling and Analysis of Information Systems*, vol. 32, no. 1, pp. 80–94, 2025. DOI: 10.18255/1818-1015-2025-1-80-94.

© Мамедов В. Ю., Ковалевский Д. А., Морозов Д. А., Столяров С. С., Оспичев С. С., 2025
Эта статья открытого доступа под лицензией CC BY license (https://creativecommons.org/licenses/by/4.0/).

Введение

Количество ежегодно публикуемых научных статей стабильно растёт. Так, в 2019 году количество документов, индексируемых Google Scholar превысило 389 млн [1], а по состоянию на февраль 2023 года в Scopus было проиндексировано более 90 млн публикаций[1]. Важной особенностью является ускорение этого роста: если за 1980 год было опубликовано в совокупности около одного миллиона статей, то к концу 2010-х число ежегодно публикуемых статей превысило семь миллионов [2].

Вместе со стремительным ростом числа работ всё более актуальными становятся инструменты для поиска и фильтрации публикаций. Большой популярностью пользуются сервисы семантического поиска статей (например, Semantic Scholar[2]). Подобные сервисы опираются на формирование семантических векторов статей и сопоставление их с запросом пользователей. Среди прочих инструментов следует упомянуть поиск по ключевым словам. Этот подход давно зарекомендовал себя в научной среде, однако по тем или иным причинам выбранные авторами ключевые слова могут отсутствовать в библиографической базе знаний. Преодолеть эту трудность помогают алгоритмы автоматической генерации ключевых слов. В последнее время лучших результатов в этой области удаётся добиться при помощи генеративных нейронных сетей, в том числе больших языковых моделей (LLM) [3, 4].

При этом невозможно утверждать, что какой-либо из этих инструментов является идеальным. Автоматические методы реферирования и формирования семантических векторов могут сталкиваться с проблемами из-за использования схожей лексики в различных не связанных областях науки. Ключевые слова обладают большой гибкостью в описании тематики, однако это же и усложняет поиск в случае, например, выбора синонимичных слов или фраз. Так, авторы могут указать ключевую фразу «нейронные сети», и такая статья не будет найдена при поиске по фразе «глубокое обучение». Унифицировать синонимичные понятия позволяет использование иерархических систем классификации тематики. Одной из таких систем является универсальная десятичная классификация.

Универсальная десятичная классификация (УДК) — это библиографическая система классификации, широко используемая по всему миру[3]. Она используется для систематизации документов, в том числе научных статей, из всех областей человеческого знания. Как и многие другие системы классификации, УДК имеет иерархическую структуру. Код УДК представляет собой последовательность десятичных цифр, разделённую для удобства чтения точками. Уточнение классификации происходит с помощью добавления дополнительных символов к последовательности справа. Таким образом, чем больше символов в УДК, тем точнее задана тематика статьи. Средняя глубина дерева классификации УДК равняется 6–9 уровням. Пример дерева представлен на рис. 1.

Общей проблемой ключевых слов и систем классификации, подобных УДК, на практике оказывается недостаточная точность определения авторами тематики своей статьи. Так, авторы могут указать слишком общие ключевые слова [5] или выбрать недостаточно глубокий код УДК. В иных случаях ключевые слова или код УДК могут быть не указаны вовсе, если этого не предполагает формат журнала или если издание использует другую систему классификации, например, Mathematics Subject Classification (MSC)[4] (причём однозначно транслировать коды между различными системами не всегда возможно). Решением этой проблемы может стать автоматическая генерация соответствующих значений. При этом, в последние годы было опубликовано множество алгоритмов автоматической генерации ключевых слов, в том числе, на материале русского языка [5, 6], тогда

[1] https://blog.scopus.com/posts/scopus-now-includes-90-million-content-records
[2] https://www.semanticscholar.org
[3] https://udcc.org/index.php
[4] https://zbmath.org/classification/

```
0 Общий отдел. Наука изнание~. Организация. Информационные технологии. Информация. Документация.
    Библиотечное дело. Учреждения. Публикации вцелом~
...
5 Математика иестественные~ науки
  50 Общие вопросы математических иестественных~ наук
  51 Математика
    510 Фундаментальные иобщие~ проблемы математики
    ...
    517 Анализ
      517.1 Введение ванализ~
      ...
      517.9 Дифференциальные, интегральные идругие~ функциональные уравнения. Конечные разности.
    Вариационное исчисление. Функциональный анализ
        517.91 Общая теория обыкновенных дифференциальных уравнений
        ...
        517.95 Дифференциальные уравнения вчастных~ производных
          517.951 Общая теория дифференциальных уравнений исистем~ счастными~ производными
          ...
          517.956 Линейные иквазилинейные~ уравнения исистемы~ уравнений
            517.956.1 Уравнения спостоянными~ коэффициентами
            517.956.2 Эллиптические уравнения исистемы~
              ...
              517.956.23 Эллиптические уравнения высокого порядка
                517.956.232 Общие свойства эллиптических уравнений высокого порядка
                517.956.233 Краевые задачи дляэллиптических~ уравнений высокого порядка
          ...
        ...
      ...
    ...
  ...
```

Fig. 1. Fragment of the UDC code tree

Рис. 1. Фрагмент дерева кодов УДК

как исследований, направленных на классификацию статей согласно УДК, нам удалось обнаружить гораздо меньше.

Цель настоящей работы состоит в создании автоматической системы присвоения научным статьям кодов согласно УДК. Для достижения этой цели нами был использован набор данных, содержащий информацию о более чем 19 тысячах статей по математике и смежным наукам. Мы рассмотрели ряд подходов, опирающихся на архитектуру BERT. Для того, чтобы учесть при классификации иерархическую природу классов УДК, мы предложили специальные метрики качества и опробовали несколько стратегий преобразования иерархических меток в плоские. Нам удалось достичь качества модели 0, 8220 в смысле иерархической рекомендательной точности (эта метрика подробно описана в разделе 3.5). Для большей репрезентативности исследования, помимо подсчёта автоматических метрик, мы привлекли к оценке качества модели экспертов. Слепое тестирование показало, что по крайней мере часть расхождений между эталонными и сгенерированными метками может быть объяснена ошибками в наборе данных.

Настоящая работа устроена следующим образом. В разделе 1 приводится краткий обзор предметной области. В разделе 2 описан собранный для нужд эксперимента набор данных. Стратегии преобразования меток и методология эксперимента описываются в разделе 3. Результаты экспериментов и их обсуждение приведены в разделе 4. Наконец, в разделе 5 подводятся итоги работы.

1. Обзор предметной области

Классификацию текстов следует считать одной из наиболее изученных областей обработки естественного языка. Долгое время в этой области лучшие результаты показывали методы классическо-

го машинного обучения, такие как метод опорных векторов, наивный байесовский классификатор и логистическая регрессия, а для векторизации текстов использовались такие простые алгоритмы как TF-IDF и n-граммы. Эти методы показывали хорошие результаты на небольших наборах данных, но их эффективность ограничивалась слабым учётом контекста и семантики текста в целом [7]. С развитием глубокого обучения удалось достичь принципиального повышения качества автоматической разметки [8]. В настоящее время прогресс в классификации текстов сосредоточен в основном вокруг подходов, опирающихся на архитектуру Transformer, в частности, BERT-подобных [9] и GPT-подобных [10] моделей.

Задача выбора класса УДК является частным случаем задачи классификации текстов: иерархической классификацией. В этом случае различные возможные классы не независимы, а упорядочены в виде подвешенного дерева, причём, чем дальше вершина находится от корня, тем более узкий специализированный класс она представляет. В таком дереве считается, что классы-потомки некоторой вершины являются подклассами класса, связанного с вершиной-предком. Автоматическая иерархическая классификация текстов естественным образом наиболее часто востребована при упорядочивании больших коллекций документов с неполной или отсутствующей метаинформацией. К исследованным в ходе предыдущих исследований коллекциям относятся, например, библиотеки научных документов [11] или выписки из медицинских карт [12]. Используемые разными авторами методы при этом значительно различаются. Так, в работе [13] в качестве алгоритмов векторизации рассматриваются GloVe, word2vec и FastText, а в качестве классификаторов — FastText, XGBoost, метод опорных векторов и свёрточная нейронная сеть. Лучшие результаты при этом продемонстрировал алгоритм FastText. В качестве корпуса для исследования использовалась коллекция RCV1, содержащая статьи издательства Reuters [14]. В работе [12] для классификации медицинских записей согласно кодам заболеваний по МКБ предлагается подход, основанный на обучении сети Multilabel-XLNet с архитектурой Transformer. При этом сравнение десятков актуальных подходов на материале нескольких наборов данных [15, 16] показывает, что лучшие из этих методов различаются по качеству разметки слабо (в пределах нескольких процентов).

Подходы, направленные на классификацию текстов согласно УДК, встречаются в то же время достаточно редко. В работе [17] рассматривается задача присвоения меток УДК текстам научных статей и газет. Рассмотренные методы достаточно просты: для векторизации текстов использованы TF-IDF и FastText, а для классификации — метод опорных векторов, многослойный перцептрон, логистическая регрессия и наивный байесовский классификатор. Авторы рассматривают классификацию с точностью до двух символов в коде УДК (всего 73 класса). Полученные значения F-меры (с макроусреднением) для разных комбинаций методов векторизации и классификации находятся в интервале от 0,42 до 0,85. Применительно к русскому языку можно упомянуть работы [18, 19]. В работе [18] для классификации использован многослойный перцептрон, а в качестве набора данных использованы статьи, размещённые на портале КиберЛенинка[5]. Авторы ограничились определением первого символа в коде УДК (всего 10 классов). Предложенный подход не позволил эффективно разделить классы 5 «Математика и естественные науки» и 6 «Прикладные науки. Медицина. Технологии». В работе [19] проводится оценка значимости различных автоматически обнаруживаемых именованных сущностей при классификации текстов согласно УДК. Авторы исследуют поддерево 517 «Анализ» на примере именованных сущностей, относящихся к методам, постановке проблемы и вычислениям. К сожалению, авторы последних двух работ не приводят численных оценок качества автоматической классификации. Таким образом, область расстановки меток УДК остаётся достаточно слабо изученной, а применяемые методы значительно отстают от аналогичных, используемых в других задачах иерархической классификации текстов.

[5] https://cyberleninka.ru/

2. Данные

Нам не удалось найти в открытом доступе набора данных, который содержал бы всю необходимую для нашего исследования информацию, а именно: название статьи, ключевые слова, аннотацию и значение УДК. В связи с этим мы решили подготовить такой набор данных самостоятельно. В качестве домена мы выбрали научные статьи по математике и смежным наукам, а в качестве источника данных — базу данных Math-Net.Ru[6]. Для того, чтобы гарантировать достаточную полноту и корректность данных, совместно с экспертами мы определили список из 21 периодического издания, из числа индексируемых Math-Net.Ru, обладающих достаточно корректной разметкой УДК и не содержащих большого числа пропусков в описании статей. Для выбранных изданий мы собрали все имеющиеся в сервисе статьи, описание которых было полным. В итоговый набор данных вошло 19233 статей, что составляет около 6 % от всех имеющихся в сервисе статей.

Средняя длина заголовка статей составила около 83 символов. В совокупности в заголовках встретилось 33337 уникальных словоформ со средней длиной слова равной 7,9 символа. Средняя длина аннотации составила около 592 символов. В совокупности в аннотациях нашлось 139627 уникальных словоформ, причём их средняя длина оказалась чуть меньше, чем в случае заголовков: 7,3 символа. Наконец, среди ключевых фраз встретилось 47332 уникальных. Средняя длина ключевой фразы составила 23,8 символа.

Основные тематики статей, вошедших в выборку, относятся к областям математики и физики. Среди собранных данных мы обнаружили 4282 уникальных значения УДК. Важно отметить, что у заметной доли статей было указано два или более значения УДК. Всего таких статей оказалось 3123 (16,2 % от всех статей в наборе данных). Эти статьи были исключены из дальнейшего рассмотрения. Статей, содержащих единственное значение УДК, оказалось 16110 (83,8 % от всех статей в наборе данных). На эти статьи в совокупности пришлось 2138 уникальных значения УДК.

Собранные статьи распределены по УДК достаточно неравномерно, что проявляется как в количестве статей с заданным значением УДК, так и в различной степени детализации разметки. Так, самое популярное значение УДК — 539.3 (Механика деформируемых тел. Упругость. Деформации) — указано в 529 статьях, а 20 самых популярных классов содержат в совокупности 4776 статей, или в 29,7 % от всего набора данных, 100 самых популярных — 9244 статей (57,4 % данных). На каждое из 1600 наименее популярных значений УДК приходится менее пяти статей, в совокупности им соответствует всего 2633 статей (16,4 % данных).

Детализация значений УДК тоже значительно разнится от статьи к статье. Чем более детализированное значение УДК указано в статье, тем точнее оно задаёт предметную область статьи. При этом из табл. 1 видно, что наиболее популярным является шестой уровень вложенности. Помимо слишком общо заданной тематики статьи, значения УДК с малой глубиной вложенности осложняют процесс обучения и оценку качества модели. Это связано с тем фактом, что сгенерированная метка может оказаться подклассом истинной. В подобных случаях весьма затруднительна автоматическая интерпретация, является ли сгенерированное значение действительно ошибочным, или уточнение истинной метки допустимо.

3. Модели и методы

3.1. Стратегии формирования целевой метки

Все классы УДК могут быть представлены как дерево-подобная структура, где каждый узел дерева представляет класс, и рёбра между узлами представляют иерархические отношения между классами. В рамках нашей работы мы рассматривали поставленную задачу как задачу плоской классификации, то есть генерировали сразу итоговую метку класса, а не проводили последова-

[6] https://www.mathnet.ru

Table 1. Number of articles depending on the discreteness of UDC codes

Таблица 1. Количество статей в зависимости от детализации УДК

Уровень детализации УДК	Количество статей
2	9
3	554
4	3252
5	3971
6	5307
7	2050
8	691
9	217
10	17
11	7
12	9
13	3
14	1
15	2
16	1

тельность иерархичных генераций, в ходе которых модель итеративно уточняет сгенерированную метку. При этом анализ набора данных показал, что распределение объектов по классам и детализация УДК крайне неравномерны. В связи с этим нами были предложены несколько стратегий для формирования целевой метки на основе оригинального кода УДК. Пример использования этих стратегий приведён на рис. 2. Рассматриваются статьи из четырёх различных УДК (стопки белых прямоугольников, высота стопки соответствует числу статей с соответствующим УДК). Для каждой из четырёх стратегий в зелёных прямоугольниках указаны метки, присвоенные статьям из стопки согласно выбранной стратегии.

1. **Наивный подход (Naive classifier).** В этом подходе мы игнорируем иерархическую структуру УДК и рассматриваем каждый класс как независимый. Этот метод прост в реализации, но он не учитывает отношения между классами и может привести к потере информации.
2. **Класс «редкие» (With «rare» class).** Этот подход похож на предыдущий, но в этом случае все классы, содержащие менее N экземпляров в обучающей выборе объединяются в специальный класс «редкие». Это позволяет сбалансировать соотношение классов в обучающей выборке и улучшить обучение модели.
3. **Фиксированный уровень иерархии (Fixed level).** В этом подходе на предварительном этапе метки классов обрезаются до определённой глубины УДК L. В случае, если после этой операции некоторые классы получают одинаковые метки, такие классы объединяются. Это позволяет уменьшить число слабопредставленных классов, однако может привести к менее точным результатам из-за огрубления меток.
4. **Объединение классов по количеству примеров с учётом таксономии (Union by number of examples).** Этот подход заключается в итеративном объединении классов, которые имеют малое количество статей, в родительский. В ходе предварительного этапа рассматриваются поддеревья УДК на материале обучающей выборки. В случае, если поддерево содержит менее S экземпляров, такое поддерево заменяется единственной вершиной, а меткой вершины становится наибольший общий префикс меток вершин, составлявших исходное поддерево. Как и в предыдущих случаях, это позволяет сбалансировать обучающую выборку, при этом потеряв меньше информации, чем при стратегиях 2 и 3.

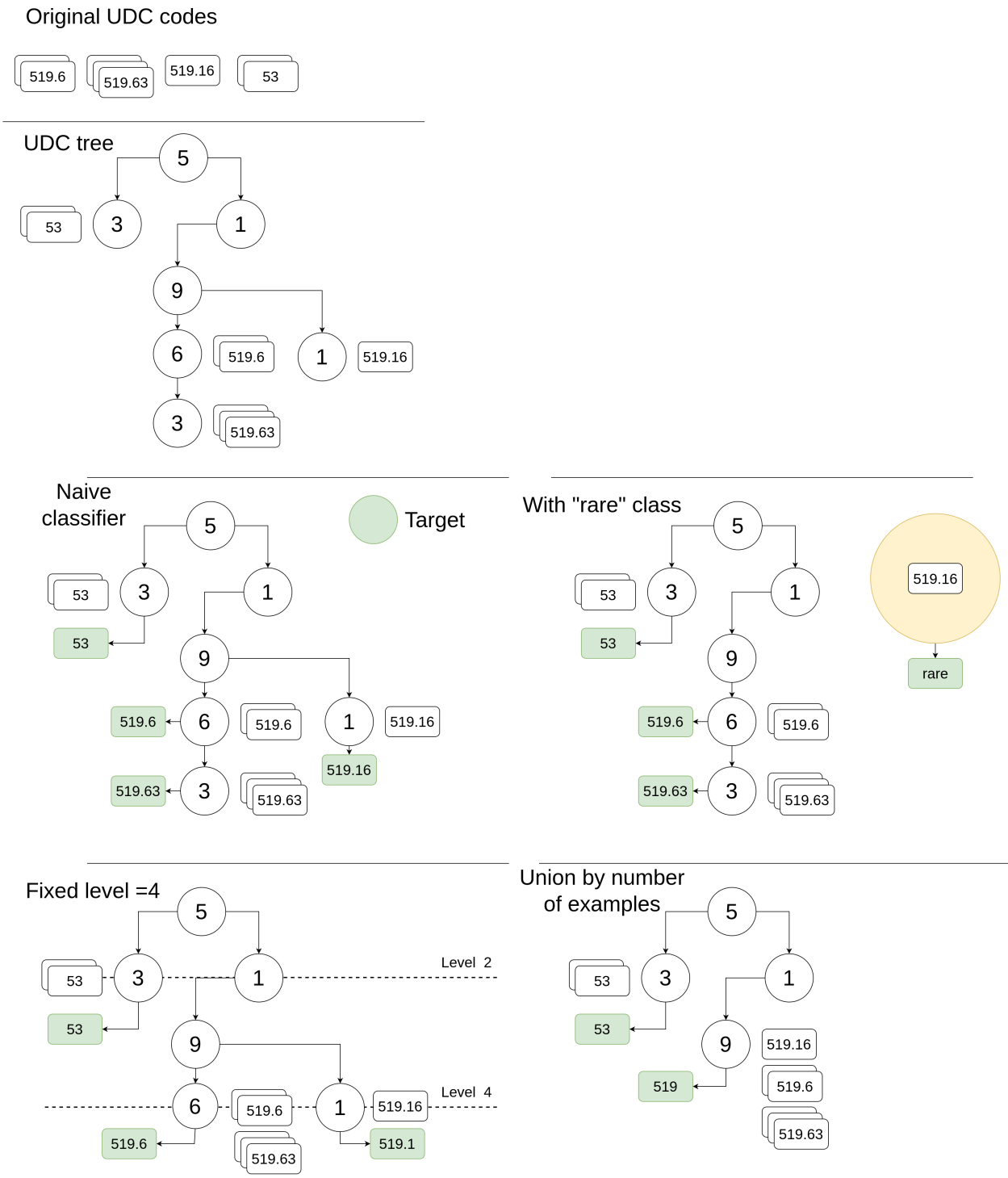

Fig. 2. Example of preparing target labels using different strategies

Рис. 2. Пример подготовки целевых меток при помощи различных стратегий

3.2. Стратегии формирования признакового описания

Собранный набор данных содержит значительное число полей, описывающих каждую публикацию. В качестве источников признакового описания мы выбрали три из них: заголовки статей, ключевые слова (указанные авторами публикаций) и аннотации. Мы изучили четыре способа формирования признакового описания на основе этих данных:

1. **Заголовки**. В этом случае в качестве признакового описания использовались только заголовки статей. Заголовки обычно содержат наиболее важную информацию о содержании статьи и могут быть полезны для её классификации. В то же время они могут быть недостаточно информативными для точной классификации, особенно если статья посвящена узкой тематике или в случае метафоричных заголовков[7].
2. **Заголовки + ключевые слова**. В этом случае в качестве признакового описания использовались заголовки и ключевые слова. Ключевые слова выбираются авторами так, чтобы отобразить тематику статьи, то есть решают схожую с метками УДК задачу. Следовательно, они могут предоставить дополнительную информацию о содержании статьи и помочь улучшить точность классификации. Тем не менее, предыдущие исследования [5] показали, что авторские ключевые слова не всегда идеально решают свою задачу. Кроме того, ключевые слова могут отсутствовать, в отличие от заголовка.
3. **Заголовки + аннотации**. В этом случае в качестве признакового описания использовались заголовки и аннотации. Аннотация статьи обычно содержит краткое описание проведённого исследования и может предоставить более полную информацию о теме статьи, чем заголовки и ключевые слова.
4. **Заголовки + ключевые слова + аннотации**. В этом случае в качестве признакового описания использовались все три выбранных поля. Подход предоставляет наиболее полную информацию о содержании статьи. Из общих соображений представляется, что это может быть наиболее эффективным способом формирования признакового описания с точки зрения точности итогового алгоритма.

3.3. Модели

Мы использовали две значительно различающиеся между собой предобученные языковые модели: многоязычную *intfloat/multilingual-e5-large*[8] [20] и ориентированную в первую очередь на русский язык *cointegrated/rubert-tiny2*[9].

Модель *multilingual-e5-large* является крупной (560 миллионов параметров) многоязычной моделью, поддерживающей 100 языков. Эта модель является дообученной на многоязычном наборе текстовых данных моделью *xlm-roberta-large*. Размер контекстного окна модели составляет 512 токенов. Эта модель ранее показала хорошие результаты на репрезентативном бенчмарке RuMTEB [21].

Крупные модели, подобные *multilingual-e5-large*, требуют для дообучения и использования значительных вычислительных мощностей. В качестве альтернативы мы рассмотрели модель *rubert-tiny2*, разработанную специально для обработки текстов на русском языке. Эта модель основана на архитектуре BERT и содержит 30 миллионов параметров. Размер её контекстного окна модели составляет 2048 токенов. За счёт малого размера возможно использование *rubert-tiny2* без графических ускорителей, что важно с прикладной точки зрения. При этом за счёт спецификации на русском языке модель может быть конкурентна на соответствующем домене.

[7] Особенно часто метафоричные заголовки используются в области искусственного интеллекта, как, например, в случае основополагающей для развития архитектуры Transformer статьи «Attention Is All You Need».

[8] https://huggingface.co/intfloat/multilingual-e5-large

[9] https://huggingface.co/cointegrated/rubert-tiny2

3.4. Постановка эксперимента

Для проведения численного эксперимента набор данных был случайным образом разделён на обучающую, валидационную и тестовую выборки в соотношении 70:15:15. Далее каждая из моделей дообучалась в течение 5 эпох с размером батча 12. В качестве оптимизатора использовался Adam [22]. Для модели *multilingual-e5-large* learning-rate зафиксирован в 10^{-5}, для модели *rubert-tiny2* — 10^{-4}. По результатам предварительных экспериментов были выбраны значения гиперпараметров для стратегий формирования целевой метки: в случае фиксированного уровня иерархии была выбрана длина метки 4, в случае объединения классов с учетом таксономии пороговый размер поддерева был выбран равным 100 статьям.

3.5. Метрики качества

Для оценки того, насколько хорошо модель обучилась, мы сравнивали её прогнозы с классами, сформированными при помощи соответствующей стратегии на базе авторской разметки. Например, для экспериментов с фиксированным уровнем иерархии, равным 4, мы сравнивали прогнозируемый класс не с исходным, а с полученным после огрубления разметки до четвёртого разряда в коде УДК. В качестве таких метрик мы использовали F-меру и Recall@k:

F-мера (F1). Одним из возможных сценариев использования нашей модели является присвоение статье наиболее вероятной метки УДК без рассмотрения остальных. Для оценки качества в таком случае хорошо подходит широко используемая в задачах классификации F-мера, позволяющая оценить баланс между точностью и полнотой модели. F-мера вычисляется как среднее гармоническое точности (precision) и полноты (recall). В свою очередь, точность вычисляется как

$$Precision = \frac{TP}{TP + FP},$$

а полнота — как

$$Recall = \frac{TP}{TP + FN}.$$

Вообще говоря, F-мера исходно разработана для бинарной классификации, и существует несколько способов обобщить её на многоклассовую постановку задачи. В настоящей работе мы использовали так называемое макроусреднение, при котором вычисляются глобальные TP, FP и FN без учёта класса.

Recall@k (R@k). В альтернативном случае потенциальный пользователь модели может рассмотреть несколько наиболее вероятных меток и выбрать из них самостоятельно. В таком случае подходящей метрикой является k-полнота (Recall@k) — доля статей, для которых истинный (авторский) УДК попадает в k наиболее вероятных предсказанных. Мы рассмотрели значения k равные 1, 2, 3 и 4.

Так как задача классификации научных статей имеет в первую очередь прикладное значение, мы сконструировали две метрики, отражающие различные сценарии использования итогового алгоритма:

Иерархическая классификационная точность (HA). С учётом того, что в процессе решения задачи классификации иерархическая природа УДК преобразуется в независимые (для модели) классы, необходимо уделять особое внимание тому, чтобы это не оказывало негативного влияния на пользовательскую оценку результата. Интуитивно понятно, что ошибка в старших разрядах УДК покажется пользователю значительно более грубой, чем ошибка в младших разрядах. Так, неумение модели отличить, например, юридические науки от физики (УДК 34 и 53, соответственно) намного менее приемлемо, чем ошибки в отделении консультационных экспертных систем (УДК

004.891.2) от диагностических (УДК 004.891.3). Для того, чтобы получить более справедливое представление о качестве создаваемой модели, мы разработали метрику, которую назвали *иерархической классификационной точностью*.

Для подсчёта иерархической классификационной точности целевое значение класса сравнивается с предсказанным поразрядно. Сравнение идёт слева направо до первого несовпадающего символа, таким образом, учитывается наибольший общий префикс двух классов p. Для подсчёта метрики суммируются веса w_i: совпадение i-го слева разряда добавляет к итоговому значению метрики $w_i = \frac{1}{i}$. Затем полученное значение делится на теоретический максимум величины, то есть на $\sum_{i=1}^{l} w_i$, где l — длина целевого значения УДК. Пусть тестовая выборка состоит из N статей, целевыми метками для которых являются $\overline{t} = (t^1, \ldots, t^N)$, а предсказанными — $\overline{p} = (p^1, \ldots, p^N)$. Тогда значение иерархической классификационной точности равно

$$\mathrm{HA}(\overline{t}, \overline{p}) = \frac{1}{N} \times \sum_{k=1}^{N} \frac{\sum_{i=1}^{lp_k} \frac{\mathbb{I}(t_i^k = p_i^k)}{i}}{\sum_{j=1}^{l_k} \frac{1}{j}},$$

где l_k — число разрядов в t^k, lp_k — длина наибольшего общего префикса t^k и p^k, t_i^k (p_i^k) — i-ый слева разряд t^k (p^k).

Иерархическая рекомендательная точность (HRA). Описывая взаимодействие с реальным пользователем системы, следует учитывать, что он способен в той или иной мере корректировать ошибки модели. В этом случае можно представить следующий сценарий использования: пользователь совершает последовательность выборов, углубляющих УДК, а модель на каждом шаге консультирует пользователя о наиболее вероятных с её точки зрения вариантах с учётом уже совершённых выборов. В таком случае, даже если модель перепутала, например, юридические науки и физику, пользователь сможет поправить её, и далее следует рассматривать предсказанные вероятности среди подклассов класса 53 «Физика». Заметим, что разрабатываемый нами классификатор позволяет такое использование: фактически, модель генерирует распределение вероятностей по классам. В таком случае, для любого префикса можно определить наиболее вероятный следующий разряд. Для оценки, учитывающей условные распределения, мы ввели метрику, которую назвали *иерархической рекомендательной точностью*. Пусть используются те же условные обозначения, что и в случае иерархической классификационной точности. Тогда значение иерархической рекомендательной точности равно:

$$\mathrm{HRA}(\overline{t}, \overline{p}) = \frac{1}{N} \times \sum_{k=1}^{N} \frac{\sum_{i=1}^{l_k} \frac{\mathbb{I}(t_i^k = p_i^k | (t_1^k, \ldots, t_{i-1}^k))}{i}}{\sum_{j=1}^{l_k} \frac{1}{j}},$$

где l_k — число разрядов в t^k, $p_i^k|(t_1^k, \ldots, t_{i-1}^k)$ — класс, который будет выбран в случае, если задан префикс $(t_1^k, \ldots, t_{i-1}^k)$, t_i^k (p_i^k) — i-й слева разряд t^k (p^k).

4. Результаты и их обсуждение

Полученные в ходе экспериментов результатов представлены в табл. 2. Для Recall@k приведены значения метрики при $k = 1 \ldots 4$. Для F-меры и Recall@k для каждого из типов классификаторов и каждой модели серым выделены лучшие полученные результаты в зависимости от способа формирования признакового описания. Для обеих иерархических точностей жирным на сером

фоне выделено лучшее полученное значение метрики среди всех изученных подходов. Серым выделены значения, отставшие от лучшего не более, чем на 1 п. п.

В большинстве случаев наибольших значений согласно F-мере и Recall@k удалось добиться при помощи наиболее подробного описания статей, включающего заголовок, ключевые слова и аннотацию. Это соотносится и с значениями иерархической точности обоих типов: за исключением классификатора, с использованием «редкого» класса лучшего результата с использованием полного признакового описания удалось добиться в четырёх из шести случаев согласно иерархической классификационной точности, в пяти из шести — согласно иерархической рекомендательной точности. При этом значения этих метрик существенно отличаются в зависимости от стратегии формирования целевых меток. Так, для фиксированного уровня иерархии удалось добиться значения Recall@1 более чем в два раза превышающего аналогичные значения для наивного классификатора (то есть для исходных меток из набора данных) и классификатора с использованием «редкого» класса.

Согласно обеим иерархическим метрикам, худшие результаты показал подход с «редким» классом. Остальные три подхода показали схожие результаты. Лучшего значения иерархической классификационной точности удалось добиться с использованием классификатора с фиксированным уровнем, полного признакового описания и модели multilingual-e5-large, а иерархической рекомендательной точности — с использованием наивного классификатора, полного признакового описания и модели multilingual-e5-large. Более того, результаты наивного классификатора для полного признакового описания и модели multilingual-e5-large уступили аналогичному классификатору с фиксированным уровнем менее 1 п. п.

Попарное сравнение иерархических точностей идентичных по архитектуре и способу формирования целевых меток классификаторов, использующих multilingual-e5-large и rubert-tiny2, показывает преимущество более крупной многоязычной модели. В большинстве пар разница составила более 1 п. п.

Дополнительно мы решили подробнее изучить случаи, при которых лучшая согласно иерархической рекомендательной сложности модель (наивный классификатор, полное признаковое описание, модель multilingual-e5-large) с высокой степенью уверенности выбрала метку УДК, отличающуюся от той, которая указана в наборе данных. Для этого мы упорядочили ошибки модели по степени уверенности и рассмотрели 100 первых среди них. В ходе слепого оценивания трое экспертов в области математики должны были ответить, какая из меток лучше подходит статье, если судить об этом по заголовку, ключевым словам и аннотации. Каждый из экспертов мог использовать одну из четырёх оценок: «первый УДК подходит лучше», «второй УДК подходит лучше», «оба УДК подходят хорошо», «оба УДК не подходят». Оказалось, что в 61 случае из 100 хотя бы два эксперта из трёх предпочли сгенерированную метку, и только в 19 случаев хотя бы два эксперта предпочли авторский код УДК. Это может подтверждает исходную гипотезу о том, что в части случаев выбираемый авторами УДК не лучшим образом соответствует их статье.

Для случаев, когда из двух предложенных большинство экспертов выбрали авторский вариант УДК, мы выделили три основных типа допущенных ошибок.

1. Недостаточно глубокая метка, например, 512.54 «Группы. Теория групп» при авторской 512.542 «Конечные группы».
2. Избыточно глубокая метка, например, 517.92 (в современной классификации отсутствует, заменён на 517.911 «Общие вопросы. Теоремы существования, теоремы единственности и теоремы о дифференциальных свойствах решений») при авторской 517.9 «Дифференциальные, интегральные и другие функциональные уравнения. Конечные разности. Вариационное исчисление.».
3. Авторская метка из слабо представленного класса, например, 004.8 «Искусственный интеллект».

Table 2. Experimental results **Таблица 2.** Результаты экспериментов

Способ учёта иерархии	Модель	Способ подачи данных	F1	R@1	R@2	R@3	R@4	HA	HRA
Наивный классификатор	**multilingual-e5-large**	Title	0,0461	0,2581	0,3410	0,4009	0,4387	0,7521	0,7935
		Title+KW	0,0619	0,3063	0,4072	0,4680	0,5108	0,7762	0,8214
		Title+Abstr	0,0611	0,3009	0,3919	0,4572	0,4968	0,7697	0,8173
		Title+KW+Abstr	0,0618	0,3032	0,4180	0,4806	0,5185	0,7756	**0,8220**
	rubert-tiny2	Title	0,0375	0,2437	0,3243	0,3730	0,4063	0,7337	0,7708
		Title+KW	0,0515	0,2815	0,3784	0,4275	0,4604	0,7541	0,7972
		Title+Abstr	0,0657	0,3113	0,3986	0,4599	0,4946	0,7584	0,8011
		Title+KW+Abstr	0,0724	0,3266	0,4086	0,4568	0,4928	0,7689	0,8042
С «редким» классом	**multilingual-e5-large**	Title	0,0470	0,2928	0,3973	0,4734	0,5203	0,5008	0,5236
		Title+KW	0,0650	0,3333	0,4586	0,5252	0,5788	0,5188	0,5481
		Title+Abstr	0,0696	0,3288	0,4500	0,5266	0,5811	0,5232	0,5527
		Title+KW+Abstr	0,0822	0,3257	0,4608	0,5392	0,5874	0,4830	0,5081
	rubert-tiny2	Title	0,0633	0,2671	0,3766	0,4347	0,4721	0,4259	0,5923
		Title+KW	0,0932	0,2941	0,4072	0,4824	0,5252	0,4791	0,6350
		Title+Abstr	0,1095	0,2995	0,4266	0,4946	0,5324	0,4660	0,6629
		Title+KW+Abstr	0,0930	0,2973	0,4311	0,4964	0,5392	0,4678	0,6325
Фиксированный (4-ый) уровень	**multilingual-e5-large**	Title	0,2092	0,6415	0,7710	0,8263	0,8548	0,7550	0,7733
		Title+KW	0,2408	0,7260	0,8316	0,8690	0,8907	0,7777	0,7909
		Title+Abstr	0,2335	0,6999	0,8189	0,8683	0,8922	0,7716	0,7885
		Title+KW+Abstr	0,2335	0,7433	0,8466	0,8855	0,9012	**0,7805**	0,7935
	rubert-tiny2	Title	0,1590	0,5966	0,7193	0,7747	0,8099	0,7430	0,7603
		Title+KW	0,1648	0,6699	0,7799	0,8234	0,8428	0,7629	0,7743
		Title+Abstr	0,1870	0,6654	0,7897	0,8353	0,8585	0,7619	0,7710
		Title+KW+Abstr	0,2368	0,7036	0,8099	0,8540	0,8795	0,7698	0,7787
С объединением классов (порог 100)	**multilingual-e5-large**	Title	0,3446	0,4126	0,5685	0,6536	0,7086	0,7444	0,7795
		Title+KW	0,4493	0,4932	0,6640	0,7410	0,7901	0,7687	0,8007
		Title+Abstr	0,4564	0,5009	0,6486	0,7302	0,7806	0,7652	0,7980
		Title+KW+Abstr	0,4707	0,5104	0,6671	0,7378	0,7955	0,7679	0,8037
	rubert-tiny2	Title	0,3580	0,4135	0,5383	0,6140	0,6694	0,7231	0,7587
		Title+KW	0,3580	0,4644	0,6149	0,7068	0,7563	0,7528	0,7877
		Title+Abstr	0,4334	0,4784	0,6288	0,6869	0,7351	0,7539	0,7835
		Title+KW+Abstr	0,4551	0,4901	0,6324	0,7009	0,7545	0,7536	0,7981

Среди случаев, когда эксперты не смогли определить лучшую метку, выделяется статья с кодом 544.452.2 «Горение. Распространение пламени» из области химии, для которой модель указала семантически схожий код 536.46 «Горение и другие реакции. Пламя» из области физики. Это показывает, что предметом отдельного исследования должен стать анализ семантической близости вершин дерева УДК, номинально далёких друг от друга с точки зрения расстояния по дереву.

Полученные нами результаты значительно расширяют ранее представленные в этой области. Обученные нами классификаторы работают с гораздо более детализированными кодами УДК. Этим, в частности, объясняется низкое качество наших моделей согласно F-мере в сравнении с [17]: если у авторов указанной работы рассматривалась классификация с фиксированным уровнем 2 (причём отсутствовала проблема вложенных классов), то в нашей работе наиболее приближенным следует считать эксперимент с фиксированным уровнем 4 (причём вложенные классы присутствуют, так как в наборе данных присутствовали статьи с уровнями 2 и 3). При этом значение HA для этой модели показывает, что модель с высокой точностью восстанавливает префиксы длины 4. Кроме того, в нашем подходе, в отличие от [17], не задействован полный текст статьи, что расширяет область потенциального применения алгоритма: во многих интернет-библиотеках научных статей полный текст работы в общем случае недоступен, в отличие от аннотации, заголовка и ключевых слов. Важным достижением в сравнении с предыдущими подходами мы считаем и внедрение двух иерархических метрик: в [17] используются лишь точность (accuracy), F-мера, точность (precision) и полнота (recall), не учитывающие иерархическую природу классификации, а в [18, 19] отсутствуют численные метрики качества классификации.

5. Заключение

В настоящей работе рассматривается задача автоматической иерархической классификации научных статей согласно универсальному десятичному классификатору. Актуальность исследования подобных подходов обусловлена быстрым ростом числа ежегодно публикуемых научных работ и необходимостью организации поиска по научным библиотекам. Мы рассмотрели подход, основанный на дообучении BERT-подобных моделей: многоязычной multilingual-e5-large и ориентированной на русский язык rubert-tiny2. Мы решали задачу при помощи плоских классификаторов, не учитывающих иерархичность классификации. При этом для учёта иерархичности УДК мы использовали несколько стратегий формирования целевых меток. Для оценки качества итоговых моделей нами были предложены две метрики, направленные на различные сценарии использования моделей: иерархическая классификационная точность (HA) и иерархическая рекомендательная точность (HRA). Лучших результатов удалось достигнуть при помощи модели multilingual-e5-large, использующей признаковое описание из заголовка, ключевых слов и аннотации к статьям. С точки зрения иерархической классификационной точности лучшей (HA=0,7805) оказалась модель, обученная с приведением целевых меток УДК к четвёртому уровню вложенности, а с точки зрения иерархической рекомендательной точности лучший результат (HRA=0,8220) показал классификатор, обученный на исходных метках УДК из набора данных. Эту же модель следует считать лучшей по совокупности метрик среди опробованных подходов. Выборочное тестирование случаев, когда предсказанный класс отличается от эталонного, показало, что по крайней мере некоторые расхождения объясняются некорректными кодами УДК, выбранными авторами статей. К ограничениям нашего исследования следует отнести малое число опробованных BERT-подобных моделей и отсутствие сравнения с иерархическими классификаторами. В ходе последующих экспериментов мы планируем устранить эти недостатки.

References

[1] M. Gusenbauer, “Google Scholar to overshadow them all? Comparing the sizes of 12 academic search engines and bibliographic databases”, *Scientometrics*, vol. 118, pp. 177–214, 2019. DOI: 10.1007/s11192-018-2958-5.

[2] M. Fire and C. Guestrin, “Over-optimization of academic publishing metrics: Observing Goodhart’s Law in action”, *GigaScience*, vol. 8, giz053, Jun. 2019. DOI: 10.1093/gigascience/giz053.

[3] R. Martinez-Cruz, A. J. Lopez-Lopez, and J. Portela, “ChatGPT vs state-of-the-art models: A benchmarking study in keyphrase generation task”, *Applied Intelligence*, vol. 55, no. 1, pp. 1–25, 2025. DOI: 10.1007/s10489-024-05901-4.

[4] M. Song *et al.*, *Is ChatGPT a good keyphrase generator? A preliminary study*, 2023. arXiv: 2303.13001 [cs.CL].

[5] A. Glazkova, D. Morozov, and T. Garipov, *Key algorithms for keyphrase generation: Instruction-based LLMs for Russian scientific keyphrases*, 2024. arXiv: 2410.18040 [cs.CL].

[6] A. V. Glazkova, D. A. Morozov, M. S. Vorobeva, and A. A. Stupnikov, “Keyword generation for Russian-language scientific texts using the mT5 model”, *Automatic Control and Computer Sciences*, vol. 58, no. 7, pp. 995–1002, 2024. DOI: 10.3103/S014641162470041X.

[7] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, “Text classification algorithms: A survey”, *Information*, vol. 10, no. 4, p. 150, 2019. DOI: 10.3390/info10040150.

[8] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, “Deep learning–based text classification: A comprehensive review”, *ACM Computing Surveys*, vol. 54, no. 3, 2021. DOI: 10.1145/3439726.

[9] S. Garg and G. Ramakrishnan, "BAE: BERT-based adversarial examples for text classification", in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020, pp. 6174–6181. DOI: 10.18653/v1/2020.emnlp-main.498.

[10] X. Sun *et al.*, "Text classification via large language models", in *Findings of the Association for Computational Linguistics*, 2023, pp. 8990–9005. DOI: 10.18653/v1/2023.findings-emnlp.603.

[11] K. Kowsari, D. E. Brown, M. Heidarysafa, K. Jafari Meimandi, M. S. Gerber, and L. E. Barnes, "HDLTex: Hierarchical deep learning for text classification", in *Proceedings of the 16th IEEE International Conference on Machine Learning and Applications*, 2017, pp. 364–371. DOI: 10.1109/ICMLA.2017.0-134.

[12] S. Strydom, A. M. Dreyer, and B. van der Merwe, "Automatic assignment of diagnosis codes to free-form text medical note", *Journal of Universal Computer Science*, vol. 29, no. 4, pp. 349–373, 2023. DOI: 10.3897/jucs.89923.

[13] R. A. Stein, P. A. Jaques, and J. F. Valiati, "An analysis of hierarchical text classification using word embeddings", *Information Sciences*, vol. 471, pp. 216–232, 2019. DOI: 10.1016/j.ins.2018.09.001.

[14] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "RCV1: A new benchmark collection for text categorization research", *Journal of Machine Learning Research*, vol. 5, pp. 361–397, 2004.

[15] Y. Wang *et al.*, "Towards better hierarchical text classification with data generation", in *Findings of the Association for Computational Linguistics*, 2023, pp. 7722–7739. DOI: 10.18653/v1/2023.findings-acl.489.

[16] A. Zangari, M. Marcuzzo, M. Rizzo, L. Giudice, A. Albarelli, and A. Gasparetto, "Hierarchical text classification and its foundations: A review of current research", *Electronics*, vol. 13, no. 7, p. 1199, 2024. DOI: 10.3390/electronics13071199.

[17] M. Kragelj and M. Borstnar, "Automatic classification of older electronic texts into the Universal Decimal Classification-UDC", *Journal of Documentation*, vol. 77, no. 3, pp. 755–776, 2021. DOI: 10.1108/JD-06-2020-0092.

[18] A. Y. Romanov, K. E. Lomotin, E. S. Kozlova, and A. L. Kolesnichenko, "Research of neural networks application efficiency in automatic scientific articles classification according to UDC", in *Proceedings of the International Siberian Conference on Control and Communications (SIBCON)*, IEEE, 2016, pp. 1–5.

[19] O. Nevzorova and D. Almukhametov, "Towards a recommender system for the choice of UDC code for mathematical articles", in *Supplementary Proceedings of the XXIII International Conference on Data Analytics and Management in Data Intensive Domains*, 2021, pp. 54–62.

[20] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei, *Multilingual E5 text embeddings: A technical report*, 2024. arXiv: 2402.05672 [cs.CL].

[21] A. Snegirev, M. Tikhonova, A. Maksimova, A. Fenogenova, and A. Abramov, *The Russian-focused embedders' exploration: ruMTEB benchmark and Russian embedding model design*, 2025. arXiv: 2408.12503 [cs.CL].

[22] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2017. arXiv: 1412.6980 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1412.6980.