УДК 004.855.5
DOI: 10.25559/SITITO.019.202303.598-606

Оригинальная статья

Применение больших языковых моделей для разметки наборов данных в задачах обработки естественного языка

К. А. Маковейчук[1*], А. В. Олифиров[2], Г. М. Деменчук[1], Я. Т Маковейчук[1]
[1] ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации», г. Москва, Российская Федерация
Адрес: 125167, Российская Федерация, г. Москва, Ленинградский проспект, д. 49/2
[*] christin2003@yandex.ru
[2] ФГАОУ ВО «Крымский федеральный университет им. В. И. Вернадского», г. Ялта, Российская Федерация
Адрес: 298635, Российская Федерация, г. Ялта, Севастопольская ул., д. 2А

Аннотация
В статье предложена методика использования больших языковых моделей компании OpenAI для разметки текстовых данных через доступный программный интерфейс. Разработанная методика является первым этапом решения задачи из категории обработки естественного языка. Задача в целом состоит в классификации курсов, которые могут быть многоклассовыми или с множественными метками, с помощью алгоритмов машинного обучения. Предоставленные слабоструктурированные данные о курсах включали большое количество различных колонок и имели размер 275811 строк, однако категории, подкатегории и предметы не были определены. Их разметка была выполнена с помощью большой языковой модели text-davinci-003, с использованием функций, написанных на языке Python. Была проведена нормализация результатов разметки и выполнен их анализ. Для проверки качества работы модели выборочно часть курсов для каждого предмета в исходных данных была размечена вручную. Более 98% курсов были классифицированы верно, следовательно, данную методику автоматизированной разметки данных с помощью большой языковой модели можно рекомендовать к использованию в дальнейшем.

Ключевые слова: большая языковая модель, text-davinci-003, интерфейс прикладного программирования, обработка естественного языка, классификация текстов, качество, Python

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Для цитирования: Применение больших языковых моделей для разметки наборов данных в задачах обработки естественного языка / К. А. Маковейчук [и др.] // Современные информационные технологии и ИТ-образование. 2023. Т. 19, № 3. С. 598-606. https://doi.org/10.25559/SITITO.019.202303.598-606

Original article

The Use of Large Language Models for Marking Datasets in Natural Language Processing

K. A. Makoveichuk[a*], A. V. Olifirov[b], G. M. Demenchuk[a], Y. T. Makoveichuk[a]
[a] Financial University under the Government of the Russian Federation, Moscow, Russian Federation
Address: 49/2 Leningradsky Prospekt, Moscow 125167, Russian Federation
[*] christin2003@yandex.ru
[b] V. I. Vernadsky Crimean Federal University, Yalta, Russian Federation
Address: 2A Sevastopol'skaya St., Yalta 298635, Russian Federation

Abstract
The article offers a method of using large language models of OpenAI company for marking text data through an accessible software interface. The developed method is the first step of solving the problem from the category of processing natural language. The problem in general consists in classifying courses that may be multi-class or multi-labeled using machine learning algorithms. The loosely structured rate data provided included a large number of different columns and were 275,811 rows long, but categories, subcategories and items were not defined. Their markup was done using the large text-davinci-003 language model, using functions written in Python. The tagging results were normalized and analyzed. To check the quality of the model, a sample of courses for each subject were manually marked in the source data. More than 98% of the courses were classified correctly, so this method of automated data marking with a large language model can be recommended for later use.

Keywords: large language model (LLM), text-davinci-003, application programming interface (API), natural language processing (NLP), text classification, quality, Python

Conflict of interests: The authors declare no conflict of interest.

For citation: Makoveichuk K. A., Olifirov A. V., Demenchuk G. M., Makoveichuk Y. T. The Use of Large Language Models for Marking Datasets in Natural Language Processing. *Modern Information Technologies and IT-Education.* 2023;19(3):598-606. https://doi.org/10.25559/SITITO.019.202303.598-606

Введение

Использование больших языковых моделей (LLM — large language models) искусственного интеллекта третьего и четвертого поколений для практического применения является актуальным вопросом в связи с появлением общедоступных генеративных моделей, в частности таких как модели компании OpenAI GPT 3.5 и GPT 4 и другие. Также важным вопросом является методика их использования через предоставляемый на платной основе API (Application Programming Interface) компании [1-12].

Постановка задачи

В исследовании была поставлена задача обработки естественного языка (NLP) по классификации текстов, которые могут быть многоклассовыми (более двух классов) или с множественными метками (более одной метки на экземпляр) [13]. Задача поставлена на примере деятельности образовательного агрегатора, который получает данные обучающих курсов с помощью API площадок-партнеров, валидирует и агрегирует данные по различным фильтрам и предоставляет пользователю удобный интерфейс для поиска и выбора курсов, а также регистрации на них. Определение информации о курсе, в том числе и его категории, подкатегории и предмета, происходит во время процесса его интеграции. Однако партнеры агрегатора предоставляют лишь частично структурированную информацию о курсах. Полученная информация содержит данные в разных форматах, которые нужно преобразовать в единый формат для дальнейшей обработки.

Кроме того, предоставленное партнерами описание курсов состоит из большого количества различных колонок, однако для выполнения разметки и формирования датасета, подходящего для дальнейшей работы с моделями машинного обучения для классификации курсов, нас интересуют лишь имя курса (name), его описание (article) и список тегов (category_tags).

Совокупность большинства процессов, связанных с интеграцией новых курсов на сайт, можно разделить на 3 ключевых этапа:

1. Получение частично структурированной информации о новом курса от партнера.
2. Структуризация данных о курсе.
3. Индексация нового курса в системе.

Рассмотрим каждый этап подробнее.

На первом этапе партнер предоставляет частично структурированную информацию о курсах. Полученная информация содержит данные в разных форматах, которые нужно преобразовать в единый формат для дальнейшей обработки. Информация, предоставляемая партнерами, показана в таблице 1.

Таблица 1. **Описание исходных данных об обучающих курсах**
Table 1. **Description of initial data about training courses**

Идентификатор	Описание	Структурировано
name	Название курса	Да
article	Описание курса, может быть представлено в различных форматах (HTML, Markdown, Plaintext)	Нет
url	URL-адрес курса на сайте сервиса	Да
article_short	Краткое описание курса, может быть представлено в различных форматах (HTML, Markdown, Plaintext)	Нет
category_tags	Теги категорий: это могут быть как категории, так и подкатегории, а также отдельные дисциплины в рамках подкатегорий. В некоторых случаях это и вовсе могут быть предложения	Нет
original_url	Оригинальный URL-адрес курса у партнера	Да

Источник: составлено авторами.
Source: Compiled by the authors.

На втором этапе происходит добавление новой информации о курсе путем структурирования существующих полей: определение категории, подкатегории и предмета на основе полей name, category_tags и article. Определение происходит с помощью примитивного классификатора на основе мешка слов — метода анализа текста, в котором текст представляется в виде множества слов без учета порядка следования слов. Этот метод используется для классификации текстов, поиска похожих документов и анализа частотности слов в тексте [14, 15].

После структуризации данных о курсе следующим этапом является индексация курса в системе. Это процесс, при котором новый курс добавляется в базу данных сайта сервиса и появляется в списке доступных курсов для пользователей.

Размер предоставленных данных — таблицы с описанием курсов — 275811 записей, при этом категории с подкатегориями и предметами не определены.

Ручная разметка таких больших данных (около 300 тысяч значений) является трудоемкой и не отвечающей критериям времени и затрат. Поэтому такая задача разметки нуждается в автоматизации.

Методы и инструментарий исследования

В качестве решения для автоматизации этой задачи предложено использовать для разметки большую языковую модель text-davinci-003 компании OpenAI, через предоставляемые методы библиотеки API для Python.

Text-davinci-003 — это один из вариантов архитектуры GPT (Generative Pretrained Transformer) [16]. OpenAI открыла API обновленной модели Davinci на базе GPT-3 в ноябре 2022 года. Модель представляет собой мощную нейронную сеть, обученную на огромном корпусе текстов (книги, статьи и веб-стра-

ницы) и способную генерировать тексты высокого качества. Text-davinci-003 обучалась с подкреплением на основе фидбека пользователей. Метод RLHF (Reinforcement Learning from Human Feedback) применялся при обучении других моделей GPT-3 и улучшил качество и скорость обучения моделей[1] [17]. Модель имеет емкость 175 млрд. параметров, что делает ее одной из самых крупных языковых моделей, доступных в настоящее время. Модель text-davinci-003 способна выполнять широкий спектр задач языка, включая генерацию текста, ответ на вопросы и классификацию текста.

Чтобы использовать модель GPT через API OpenAI, пользователь должны отправить запрос, содержащий входные данные и персональный ключ API, и получить ответ, содержащий выходные данные модели.

Основная часть

Рассмотрим процесс разметки слабоструктурированных данных с помощью модели text-davinci-003.

Имеется проблема использования модели для данных такого размера, связанная с ограничением запроса в 4000 токенов (приблизительно 3000 слов латиницы), включая ввод и вывод. Поэтому просто передать список предметов для определения не представляется возможным. Для обхода ограничений необходимо сделать разметку в несколько этапов.

1. Определение основной категории, к которой относится заданный курс. На вход сети поступает имя курса, его список тегов и список категорий. Определяется главная категория курса.
2. Программа получает список предметов на основе определенной категории.
3. Определение множества предметов и главного предмета, к которой относится заданный курс. На вход сети поступает имя курса, его список тегов и список предметов.
4. Программа получает список множества подкатегорий и главных подкатегорий на основе распознанных ранее предметов.

Для использования библиотеки OpenAI API в Python необходимо задать 7 аргументов в методе API (см. рис. 1) [18].

```
In [7]: prompt = """
Course "Introduction to Python Programming" is related to:

a. Software Development
b. Music study
c. Accounting

Chose one
"""

response = openai.Completion.create(
    model="text-davinci-003",
    prompt=prompt,
    temperature=0,
    max_tokens=60,
    top_p=1,
    frequency_penalty=0.5,
    presence_penalty=0,
)
print(response)
```

```
{
  "choices": [
    {
      "finish_reason": "stop",
      "index": 0,
      "logprobs": null,
      "text": "a. Software Development"
    }
  ],
  "created": 1681739824,
  "id": "cmpl-76JdoG9iEuCQc0j6sFHKE0AmPRHtk",
  "model": "text-davinci-003",
  "object": "text_completion",
  "usage": {
    "completion_tokens": 4,
    "prompt_tokens": 33,
    "total_tokens": 37
  }
}
```

Р и с. 1. Входные аргументы модели для разметки датасета в API OpenAI

F i g. 1. Model input arguments for dataset layout in the OpenAI API

Источник: здесь и далее в статье все рисунки составлены авторами.

Source: Hereinafter in this article all figures were drawn up by the authors.

[1] ChatGPT can now see, hear, and speak [Электронный ресурс] // OpenAI. September 25, 2023. URL: https://openai.com/blog/chatgpt-can-now-see-hear-and-speak (дата обращения: 14.09.2023).

Для определения категории используем генерацию промпта с последующим получением результата в методе detect_primary_category. На вход методу подается название курса (headline), список категорий для выбора (categories) и список тегов (tags_list) (см. рис. 2). Далее происходит проверка на то, есть ли значения в списке тегов и уже с учетом этой информации формируется промпт, который затем передается в API.

Раз в 500 строк осуществляем выгрузку результатов распознавания в файл (тип csv). После завершения распознавания категорий необходимо нормализовать результаты, т. к. text-davinci-003, как и другие генеративные модели, имеет особенность «галлюцинировать» и выдавать те категории, которых нет в исходном списке или изменять исходный список [19-21]. Пример таких преобразований представлен на рисунке 3.

```
1 usage  ± Demka
def detect_primary_category(headline: str, categories: List[str], tags_list: List[str]) -> str:
    if len(tags_list) == 0:
        prompt = f"""
        Choose the most appropriate category for the following headline from the list below. Note that you can only select one category.

        Headline: "{headline}"
        Categories list: "{categories}"
        Primary category:"""

    else:
        prompt = f"""
        Choose the most appropriate category for the following headline from the list below.
        Note that you can only select one category. You can use the tags provided to assist in generating the best category for the headline.

        Headline: "{headline}"
        Categories list: "{categories}"
        Tags list: "{tags_list}"
        Primary category:"""

    response = openai.Completion.create(
        model="text-davinci-003",
        prompt=prompt,
        temperature=0,
        max_tokens=60,
        top_p=1,
        frequency_penalty=0.5,
        presence_penalty=0,
    )
    return response["choices"][0]["text"]
```

Р и с. 2. Метод detect_primary_category для генерации промптов категорий и взаимодействия с API OpenAI
F i g. 2. The detect_primary_category method for generating category prompts and interacting with the OpenAI API

```
In [7]: with open("./all_hierarhy.json", "r") as file:
            hierarhy_data = json.loads(file.read())

        categories_list = list(hierarhy_data.keys())
        categories_list

Out[7]: ['Music',
         'Design',
         'Office',
         'Humanities',
         'Lifestyle',
         'Business & Finance',
         'Sciences',
         'IT & Software',
         'Photography & Video']
```

```
In [9]: list(df[~df['OPENAI_CATEGORY'].isin(categories

Out[9]: ['Teaching & Academics',
         'Other Health & Fitness',
         'Religion & Spirituality',
         'Martial Arts & Self Defense',
         'Sports',
         'Law',
         'Other Teaching & Academics',
         'Gesundheit & Fitness',
         'Teacher Training',
         'Parenting & Relationships',
         'Engineering',
         'Language Learning',
         'Social Science',
         'Health',
         'Test Prep',
         'None',
         'Online Education',
         'Martial Arts & Self-Defense',
         'Martial Arts & Selbstverteidigung']
```

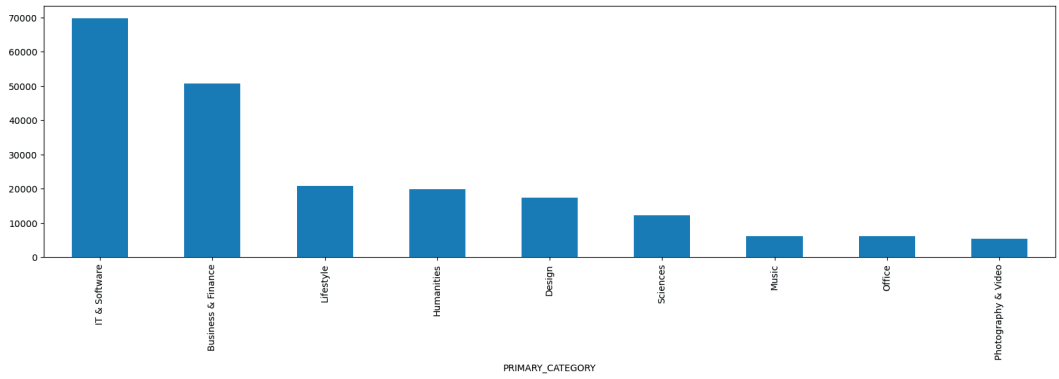
Р и с. 3. Пример «галлюцинирования» модели (исходный список категорий и список категорий, которых нет в переданном списке)
F i g. 3. Example of “hallucination” model (the source list of categories and the list of categories that are not in the transmitted list)

Для нормализации категорий на первом этапе удаляем пробелы и специальные символы, на втором — оставляем лишь те записи, где категории есть в исходном списке категорий.

После применения нормализации предметов, выборка сократилась с 275811 элементов до 275223, т. е. отсеялось 588 элементов, или 0.2 % выборки, что является отличным результатом.

Далее определяем предметы с помощью метода detect_

subjects_request. В метод передается название курса, (headline), список предметов (subjects), а также список тегов (tags_list).
После продолжительной обработки получаем список неструктурированных предметов. Для его нормализации используем подход, аналогичный нормализации категорий.
Переходим к этапу фильтрации предметов на существующие в нашем списке. На основании каждого распознанного предмета мы заново сформируем иерархию в виде категории и подкатегории курса.
Так же, как и в случае с категориями, OpenAI может менять предмет из переданного списка на свой собственный. Для этого вводим словарь, содержащий информацию об альтернативных названиях предметов, и далее фильтруем предметы на существование в нашей иерархии.
После чего остается последний шаг — выделение категорий и подкатегорий на основе распознанных предметов.
После применения всех методов, нацеленных на нормализацию данных в датасете и формирования всех основных объектов в иерархии, остается 208656 значений из 275223, т. е. в этом уже случае произошел уже отсев 24 % выборки, что все еще удовлетворяет нашим критериям. Сделаем группировку в полученном датафрейме и выведем график количества курсов по главным категориям (см. рис. 4).



Р и с. 4. Столбцовая диаграмма количества курсов по главным категориям
F i g. 4. Column chart of number of courses by main categories

Р и с. 5. Результаты ручной перепроверки сгенерированных результатов
F i g. 5. Results of manual double-check of generated results

Ожидаемо, распределение курсов по категориям разбалансировано, однако это не является проблемой с учетом объема выборки [22, 23]. Категория с минимальным количеством курсов — Photography & Video (5419 значений), максимальным — IT & Software (69830 значений).
Переходя к подкатегориям, получаем 14369 курса для Programming languages и лишь 174 для Remote Work & Collaboration.

После анализа данных берем лишь колонки с актуальной информацией и экспортируем полученный датасет в .csv файл для дальнейшей обработки на стороне классификаторов.
После формирования категорий, подкатегорий и предметов на основе данных OpenAI нам следует убедиться, что полученные результаты действительно можно использовать в качестве канонических меток. Для этого от всего датасета берем по несколько курсов для каждого предмета и вручную проставляем

метку соответствия на каждом курсе: если все правильно, то ставим 1, иначе 0 и пишем комментарий.
Как видим на рисунке 5, более 98 % курсов было классифицировано верно, значит, автоматизированную разметку данных использовать можно [24, 25].

Результаты

В работе была произведена автоматическая разметка исходных больших данных по учебным курсам. С этой целью был выполнен анализ принципов использования API OpenAI и использована генеративная модель OpenAI text-davinci-003.
Разметка является первым этапом разработки сервиса поиска и подбора учебных курсов. Дальнейшие этапы исследования включали использование размеченного датасета для разработки различных вариаций классификаторов учебных курсов, проведение тестирования на реальных данных.
Использование LLM позволяет улучшить качество разметки данных и повысить точность работы классификатора.

Список использованных источников

[1] Multimodal Few-Shot Learning with Frozen Language Models / M. Tsimpoukelli [et al.] // 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Curran Associates, Inc., 2021. Vol. 34. P. 200-212. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/01b7575c38dac42f3cfb7d500438b875-Paper.pdf (дата обращения: 14.09.2023).

[2] GPT-4V(ision) is a Human-Aligned Evaluator for Text-to-3D Generation / Y. Zhengyuan [et al.] // arXiv:2401.04092. 2024. https://doi.org/10.48550/arXiv.2401.04092

[3] Supervised Multimodal Bitransformers for Classifying Images and Text / D. Kiela [et al.] // arXiv:1909.02950. 2019. https://doi.org/10.48550/arXiv.1909.02950

[4] A Survey of Large Language Models / W. X. Zhao [et al.] // arXiv:2303.18223. 2023. https://doi.org/10.48550/arXiv.2303.18223

[5] A comprehensive overview of large language models / H. Naveed [et al.] // arXiv:2307.06435. 2024. https://doi.org/10.48550/arXiv.2307.06435

[6] Намиот Д. Е., Ильюшин Е. А., Чижов И. В. Искусственный интеллект и кибербезопасность // International Journal of Open Information Technologies. 2022. Т. 10, № 9. С. 135-147. EDN: DYQWEH

[7] Намиот Д. Е. Схемы атак на модели машинного обучения // International Journal of Open Information Technologies. 2023. Т. 11, № 5. С. 68-86. EDN: YVRDOB

[8] On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? / E. M. Bender [et al.] // Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21). New York, NY, USA: Association for Computing Machinery, 2021. P. 610-623. https://doi.org/10.1145/3442188.3445922

[9] Arora S., Goyal A. A Theory for Emergence of Complex Skills in Language Models // arXiv:2307.15936. 2023. https://doi.org/10.48550/arXiv.2307.15936

[10] Privacy in Large Language Models: Attacks, Defenses and Future Directions / H. Li [et al.] // arXiv:2310.10383. 2023. https://doi.org/10.48550/arXiv.2310.10383

[11] Navigli R., Conia S., Ross B. Biases in Large Language Models: Origins, Inventory, and Discussion // Journal of Data and Information Quality. 2023. Vol. 15, no. 2. Article number: 10. https://doi.org/10.1145/3597307

[12] On the application of Large Language Models for language teaching and assessment technology / A. Caines [et al.] // CEUR Workshop Proceedings. 2023. Vol. 3487. P. 173-197. URL: https://ceur-ws.org/Vol-3487/paper12.pdf (дата обращения: 14.09.2023).

[13] Казакова М. А., Султанова А. П. Анализ технологии обработки естественного языка: современные проблемы и подходы // Advanced Engineering Research (Rostov-on-Don). 2022. Т. 22, № 2. С. 169-176. https://doi.org/10.23947/2687-1653-2022-22-2-169-176

[14] Ray P. P. ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope // Internet of Things and Cyber-Physical Systems. 2023. Vol. 3. P. 121-154. https://doi.org/10.1016/j.iotcps.2023.04.003

[15] Alkhalifa R., Kochkina E., Zubiaga A. Building for tomorrow: Assessing the temporal persistence of text classifiers // Information Processing & Management. 2023. Vol. 60, issue 2. Article number: 103200. https://doi.org/10.1016/j.ipm.2022.103200

[16] Beyond Lexical Consistency: Preserving Semantic Consistency for Program Translation / Y. Du [et al.] // 2023 IEEE International Conference on Data Mining (ICDM). Shanghai, China: IEEE Computer Society, 2023. P. 91-100. https://doi.org/10.1109/ICDM58522.2023.00018

[17] Human-Centered Reinforcement Learning: A Survey / G. Li [et al.] // IEEE Transactions on Human-Machine Systems. 2019. Vol. 49, no. 4. P. 337-349. https://doi.org/10.1109/THMS.2019.2912447

[18] Маркеев М. В. Методика автоматизированной разметки изображений и нахождения ключевых слов // Международный журнал гуманитарных и естественных наук. 2022. Т. 11-2, № 74. С. 115-120. https://doi.org/10.24412/2500-1000-2022-11-2-115-120

[19] Heyman T., Heyman G. The impact of ChatGPT on human data collection: A case study involving typicality norming data // Behavior Research Methods. 2023. https://doi.org/10.3758/s13428-023-02235-w

[20] ChatGPT for good? On opportunities and challenges of large language models for education / E. Kasneci [et al.] // Learning and Individual Differences. 2023. Vol. 103. Article number: 102274. https://doi.org/10.1016/j.lindif.2023.102274

[21] Opinion Paper: «So what if ChatGPT wrote it?» Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy / Dwivedi Y. K. [et al.] // International Journal of Information Management. 2023. Vol. 71. Article number: 102642. https://doi.org/10.1016/j.ijinfomgt.2023.102642
[22] Сальникова К. В. Анализ массива данных с помощью инструмента визуализации «ящик с усами» // Universum: экономика и юриспруденция. 2021. № 6(81). С. 11-17. EDN: APSOIG
[23] Дарманян А. П. Использование показателей описательной статистики для характеристики эмпирических выборок макроэкономических индикаторов // Экономика региона. 2013. № 2(34). С. 157-163. https://doi.org/10.17059/2013-2-16
[24] Development of a Cyber-Resistant Platform for the Internet of Things Based on Dynamic Control Technology / S. Petrenko [et al.] // Futuristic Trends in Network and Communication Technologies. FTNCT 2020. Communications in Computer and Information Science ; ed. by P. K. Singh, G. Veselov, V. Vyatkin, A. Pljonkin, J. M. Dodero, Y. Kumar. Vol. 1395. Singapore: Springer, 2021. P. 144-154. https://doi.org/10.1007/978-981-16-1480-4_13
[25] Analysis and Synthesis of Educational Content of Courses in Moodle LMS Based on the Competence Approach of FSES / K. Makoveichuk [et al.] // CEUR Workshop Proceedings. 2021. Vol. 3057. P. 176-183. URL: https://ceur-ws.org/Vol-3057/paper19.pdf (дата обращения: 14.09.2023).

Поступила 14.09.2023; одобрена после рецензирования 03.10.2023; принята к публикации 11.10.2023.

Об авторах:

Маковейчук Кристина Александровна, доцент Департамента анализа данных и машинного обучения, ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации» (125167, Российская Федерация, г. Москва, Ленинградский проспект, д. 49/2), кандидат экономических наук, доцент, **ORCID: https://orcid.org/0000-0003-1258-0463,** christin2003@yandex.ru

Олифиров Александр Васильевич, профессор кафедры экономики и финансов Гуманитарно-педагогической академии (филиала), ФГАОУ ВО «Крымский федеральный университет им. В. И. Вернадского» (298635, Российская Федерация, г. Ялта, Севастопольская ул., д. 2А), доктор экономических наук, профессор, **ORCID: https://orcid.org/0000-0002-5288-2725,** alex.olifirov@gmail.com

Деменчук Георгий Максимович, бакалавр (выпускник) Департамента анализа данных и машинного обучения, ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации» (125167, Российская Федерация, г. Москва, Ленинградский проспект, д. 49/2), **ORCID: https://orcid.org/0000-0003-2849-982X**

Маковейчук Ян Тарасович, магистрант Департамента анализа данных и машинного обучения, ФГОБУ ВО «Финансовый университет при Правительстве Российской Федерации» (125167, Российская Федерация, г. Москва, Ленинградский проспект, д. 49/2), **ORCID: https://orcid.org/0000-0002-8919-7828**

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

[1] Tsimpoukelli M., Menick J., Cabi S., Ali Eslami S. M., Vinyals O., Hill F. Multimodal Few-Shot Learning with Frozen Language Models. In: Ranzato M. et al. (eds.) 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Curran Associates, Inc.; 2021. Vol. 34. P. 200-212. Available at: https://proceedings.neurips.cc/paper_files/paper/2021/file/01b7575c38dac42f3cfb7d500438b875-Paper.pdf (accessed 14.09.2023).
[2] Zhengyuan Y., Li L., Lin K., Wang J., Lin Ch.-Ch., Liu Z., Wang L. GPT-4V(ision) is a Human-Aligned Evaluator for Text-to-3D Generation. *arXiv*:2401.04092. 2024. https://doi.org/10.48550/arXiv.2401.04092
[3] Kiela D., Bhooshan S., Firooz H., Perez E., Testuggine D. Supervised Multimodal Bitransformers for Classifying Images and Text. *arXiv*:1909.02950. 2019. https://doi.org/10.48550/arXiv.1909.02950
[4] Zhao W.X. et al. A Survey of Large Language Models. *arXiv*:2303.18223. 2023. https://doi.org/10.48550/arXiv.2303.18223
[5] Naveed H. A comprehensive overview of large language models. *arXiv*:2307.06435. 2024. https://doi.org/10.48550/arXiv.2307.06435
[6] Namiot D.E., Ilyushin E.A., Chizhov I.V. Artificial Intelligence and Cybersecurity. *International Journal of Open Information Technologies*. 2022;10(9):135-147. (In Russ., abstract in Eng.) EDN: DYQWEH
[7] Namiot D.E. Schemes of attacks on machine learning models. *International Journal of Open Information Technologies*. 2023;11(5):68-86. (In Russ., abstract in Eng.) EDN: YVRDOB
[8] Bender E.M. et al. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT '21). New York, NY, USA: Association for Computing Machinery; 2021. p. 610-623. https://doi.org/10.1145/3442188.3445922
[9] Arora S., Goyal A. A Theory for Emergence of Complex Skills in Language Models. *arXiv*:2307.15936. 2023. https://doi.org/10.48550/arXiv.2307.15936

[10] Li H. et al. Privacy in Large Language Models: Attacks, Defenses and Future Directions. *arXiv*:2310.10383. 2023. https://doi.org/10.48550/arXiv.2310.10383
[11] Navigli R., Conia S., Ross B. Biases in Large Language Models: Origins, Inventory, and Discussion. *Journal of Data and Information Quality*. 2023;15(2):10. https://doi.org/10.1145/3597307
[12] Caines A. et al. On the application of Large Language Models for language teaching and assessment technology. *CEUR Workshop Proceedings*. 2023;3487:173-197. Available at: https://ceur-ws.org/Vol-3487/paper12.pdf (accessed 14.09.2023).
[13] Kazakova M.A., Sultanova A.P. Analysis of natural language processing technology: modern problems and approaches. *Advanced Engineering Research (Rostov-on-Don)*. 2022;22(2):169-176. https:doi.org/10.23947/2687-1653-2022-22-2-169-176
[14] Ray P.P. ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*. 2023;3:121-154. https://doi.org/10.1016/j.iotcps.2023.04.003
[15] Alkhalifa R., Kochkina E., Zubiaga A. Building for tomorrow: Assessing the temporal persistence of text classifiers. *Information Processing & Management*. 2023;60(2):103200. https://doi.org/10.1016/j.ipm.2022.103200
[16] Du Y., Ma Y. -F., Xie Z., Li M. Beyond Lexical Consistency: Preserving Semantic Consistency for Program Translation. In: 2023 IEEE International Conference on Data Mining (ICDM). Shanghai, China: IEEE Computer Society; 2023. p. 91-100. https://doi.org/10.1109/ICDM58522.2023.00018
[17] Li G. et al. Human-Centered Reinforcement Learning: A Survey. *IEEE Transactions on Human-Machine Systems*. 2019;49(4):337-349. https://doi.org/10.1109/THMS.2019.2912447
[18] Markeev M.V. *Metodika avtomatizirovannoj razmetki izobrazhenij i nahozhdeniya klyuchevyh slov* [methods of automated image markup and keyword finding]. *Mezhdunarodnyj zhurnal gumanitarnyh i estestvennyh nauk* = International Journal of Humanities and Natural Sciences. 2022;11-2(74):115-120. (In Russ., abstract in Eng.) https:doi.org/10.24412/2500-1000-2022-11-2-115-120
[19] Heyman T., Heyman G. The impact of ChatGPT on human data collection: A case study involving typicality norming data. *Behavior Research Methods.* 2023. https://doi.org/10.3758/s13428-023-02235-w
[20] Kasneci E. et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*. 2023;103:102274. https://doi.org/10.1016/j.lindif.2023.102274
[21] Dwivedi Y.K. et al. Opinion Paper: "So what if ChatGPT wrote it?" Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *International Journal of Information Management*. 2023;71:102642. https://doi.org/10.1016/j.ijinfomgt.2023.102642
[22] Salnikova K.V. *Analiz massiva dannyh s pomoshch'yu instrumenta vizualizacii «yashchik s usami»* [The analysis of data amount using the visualization tool "box-and-whisker"]. *Universum: ekonomika i yurisprudenciya* = Universum: economics and law. 2021;(6):11-17. (In Russ., abstract in Eng.) EDN: APSOIG
[23] Darmanyan A.P. *Ispol'zovanie pokazatelej opisatel'noj statistiki dlya harakteristiki empiricheskih vyborok makroekonomicheskih indikatorov* [The use of descriptive statistics for the characteristics of the empirical samples macroeconomic indicators]. *Ekonomika regiona* = Economy of Regions. 2013;(2):157-163. (In Russ., abstract in Eng.) https://doi.org/10.17059/2013-2-16
[24] Petrenko S., Petrenko A., Makoveichuk K.A., Olifirov A. Development of a Cyber-Resistant Platform for the Internet of Things Based on Dynamic Control Technology. In: Singh P.K., Veselov G., Vyatkin V., Pljonkin A., Dodero J.M., Kumar Y. (eds.) Futuristic Trends in Network and Communication Technologies. FTNCT 2020. *Communications in Computer and Information Science*. Vol. 1395. Singapore: Springer; 2021. p. 144-154. https://doi.org/10.1007/978-981-16-1480-4_13
[25] Makoveichuk K., Oleinikov N., Gorbunova N., Ponomareva E., Makoveichuk Ya. Analysis and Synthesis of Educational Content of Courses in Moodle LMS Based on the Competence Approach of FSES. *CEUR Workshop Proceedings*. 2021;3057:176-183. Available at: https://ceur-ws.org/Vol-3057/paper19.pdf (accessed 14.09.2023).

Submitted 14.09.2023; approved after reviewing 03.10.2023; accepted for publication 11.10.2023.

About the authors:

Krystina A. Makoveichuk, Associate Professor of the Department of Data Analysis and Machine Learning, Financial University under the Government of the Russian Federation (49/2 Leningradsky Prospekt, Moscow 125167, Russian Federation), Cand. Sci. (Econ.), Associate Professor, **ORCID: https://orcid.org/0000-0003-1258-0463,** christin2003@yandex.ru

Alexander V. Olifirov, Professor of the Chair of Economics and Finance of Humanities and Education Science Academy (branch) of V. I. Vernadsky Crimean Federal University (2A Sevastopol'skaya St., Yalta 298635, Russian Federation), Dr. Sci. (Econ.), Professor, **ORCID: https://orcid.org/0000-0002-5288-2725,** alex.olifirov@gmail.com

Georgiy M. Demenchuk, A Bachelor's Degree in Data Analytics, Financial University under the Government of the Russian Federation (49/2 Leningradsky Prospekt, Moscow 125167, Russian Federation), **ORCID: https://orcid.org/0000-0003-2849-982X**

Yan T. Makoveichuk, Master degree student of the Department of Data Analysis and Machine Learning, Financial University under the Government of the Russian Federation (49/2 Leningradsky Prospekt, Moscow 125167, Russian Federation), **ORCID: https://orcid.org/0000-0002-8919-7828**

All authors have read and approved the final manuscript.