

УДК 004.42:004.032.26:378.1
DOI 10.17513/snt.40523

ВНЕДРЕНИЕ ТЕХНОЛОГИЙ ГЕНЕРАТИВНОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В МОДУЛЬ КОНТРОЛЯ ЗНАНИЙ

Буткина А.А., Егоров М.С.

ФГБОУ ВО «Мордовский государственный исследовательский университет
имени Н.П. Огарёва», Россия, Саранск, e-mail: butkinaaa@gmail.com

В настоящее время одной из актуальных задач, поставленных в сфере цифровизации образования, является персонификация образования с применением технологий искусственного интеллекта. В частности, одной из проблем дистанционного обучения можно назвать отсутствие персонализированной обратной связи, так как обучающиеся часто не понимают, где именно они ошиблись при прохождении контроля знаний, и не получают четких рекомендаций по дальнейшему изучению материала, что может снижать их мотивацию к обучению и ухудшать эффективность обучения в целом. Данные аспекты подтверждают актуальность исследований, направленных на разработку инструментов, призванных решить описанную проблему. В статье приводятся результаты проведенного авторами исследования, целью которого является повышение эффективности образовательного процесса за счет внедрения в модуль контроля знаний обучающихся онлайн-курсов технологий генеративного искусственного интеллекта для формирования персонализированных рекомендаций обучающимся на основе результатов их тестирования. Внедрение данных технологий в разработанный программный модуль позволит обучающимся получать обратную связь с использованием генеративной текстовой модели YandexGPT, которая настроена анализировать уже пройденные тесты и предоставлять рекомендации о том, какие темы и материалы необходимо повторить обучающимся для более успешного прохождения испытаний. Статья содержит описание используемого авторами стека технологий для разработки описанного модуля, удовлетворяющего современным требованиям к гибкости и масштабируемости приложений. В статье обоснован выбор микросервисной архитектуры при разработке приложения и приведена диаграмма классов микросервиса, реализующего подключение к YandexGPT API и обработку запросов к ней, с описанием основных классов. Продемонстрированы примеры сгенерированных рекомендаций по результатам тестирования в разработанном модуле. Сделаны выводы о применимости данного модуля.

Ключевые слова: образование, генеративный искусственный интеллект, YandexGPT, разработка, контроль знаний, персонификация рекомендаций, тестирование

IMPLEMENTATION OF GENERATIVE ARTIFICIAL INTELLIGENCE TECHNOLOGIES INTO THE KNOWLEDGE CONTROL MODULE

Butkina A.A., Egorov M.S.

Mordovian State Research University named after N.P. Ogarev,
Russia, Saransk, e-mail: butkinaaa@gmail.com

Currently, one of the urgent tasks set in the field of education digitalization is learning personalization using artificial intelligence technologies. In particular, one of the problems of distance learning can be called the lack of personalized feedback, since students often do not understand where exactly they made a mistake when passing the knowledge test, and do not receive clear recommendations for further study of the material, which can reduce their motivation for learning and worsen the effectiveness of learning in general. These aspects confirm the relevance of research aimed at developing tools to solve the described problem. The article presents the results of investigation carried out by the authors, the purpose of which is to improve the efficiency of the educational process by introducing generative artificial intelligence technologies into the knowledge control module of online course students to provide personalized recommendations for students based on their test results. The implementation of these technologies in the developed software module will allow students to receive feedback using the YandexGPT generative text model, which is configured to analyze already completed tests and provide recommendations on what topics and materials students need to repeat for more successful passing of tests. The article contains a description of the technology stack used by the authors to develop the described module, which satisfy modern requirements for flexibility and scalability of software. The article substantiates the choice of microservice architecture when developing an application and provides a class diagram of the microservice implementing the connection to the YandexGPT API and processing requests to it, with a description of the main classes. Examples of generated recommendations based on the testing results in the developed module are demonstrated. Conclusions are made about the applicability of this module.

Keywords: education, generative artificial intelligence, YandexGPT, development, knowledge control, personalization of recommendations, testing

Введение

С развитием информационных технологий все большее внимание уделяется автоматизации образовательных процессов в вузах и развитию онлайн-курсов. В качестве

одного из наиболее актуальных для этого инструментов в настоящее время выступают различные приложения на базе искусственного интеллекта, которые внедряются в образование в целях персонификации об-

разования [1–3]. Естественно, применение данных приложений имеет как свои преимущества, так и риски, которые широко обсуждаются в научном сообществе [4–6]. Однако нужно принять тот факт, что развитие информационных технологий и цифровизация образования не стоят на месте, и, для того чтобы идти в ногу со временем, требуется адаптировать данные приложения под нужды преподавателей и обучающихся. В частности, одной из основных проблем онлайн-курсов можно назвать отсутствие персонализированной обратной связи, так как обучающиеся часто не понимают, где именно они ошиблись при прохождении контроля знаний, и не получают четких рекомендаций по дальнейшему изучению материала, что может снижать их мотивацию к обучению и ухудшать эффективность обучения в целом. Поэтому разработка инструментов, устраняющих данную проблему посредством использования технологий искусственного интеллекта, поможет сделать онлайн-образование более эффективным и удобным, а также закроет пробелы, связанные с недостаточной обратной связью, и повысит удовлетворенность пользователей, что обуславливает актуальность проводимого авторами исследования.

Научная новизна работы заключается в том, что авторы предлагают новый подход к анализу результатов тестирования обучающихся – внедрение генеративной текстовой модели в модуль контроля знаний обучающихся для проведения данного анализа, что позволит получить более глубокий и детальный анализ по сравнению с традиционными методами. Использование данной модели в модуле контроля знаний предоставит обучающимся возможность не только выявить общий уровень их знаний посредством тестирования, но и определить конкретные темы и материалы, которые требуют дополнительного изучения за счет формирования индивидуальных рекомендаций. Именно применение технологий генеративного искусственного интеллекта позволит адаптировать рекомендации под индивидуальные потребности и уровень знаний каждого обучающегося, что, в свою очередь, способствует более эффективному повторению материала, повышению мотивации к обучению, а также улучшению качества обратной связи по сравнению с традиционными методами.

Цель исследования – повышение эффективности образовательного процесса за счет внедрения в модуль контроля знаний обучающихся онлайн-курсов технологий генеративного искусственного интеллекта

для формирования персонализированных рекомендаций обучающимся на основе результатов их тестирования.

Материалы и методы исследования

Внедрение технологий генеративного искусственного интеллекта для формирования персонализированных рекомендаций по результатам прохождения тестирования выполнялось в разработанный авторами модуль контроля знаний, описанный в [7].

Авторами был проведен анализ научных публикаций, посвященных исследованию генеративных моделей искусственного интеллекта [8–10], а также был проведен ряд экспериментов по оценке функциональности приложений данных моделей и анализа их потенциала для внедрения в образовательный процесс. Особое внимание было уделено сравнительному анализу трех нейросетевых онлайн-сервисов: ChatGPT, YandexGPT и GigaChat. Выбор данных генеративных моделей обусловлен универсальностью их архитектур, популярностью в сфере ИТ, доступностью (предоставление API), наличием собственных инновационных разработок и предложением разнообразного функционала пользователям. В результате проведенного авторами анализа для выдачи индивидуальных рекомендаций в модуле контроля знаний среди перечисленных сервисов была выбрана российская генеративная текстовая модель от Яндекса YandexGPT [11–13], которая способна анализировать уже пройденные тесты и подсказывать, какие темы и материалы необходимо повторить обучающимся для более успешного прохождения испытаний. Среди факторов, повлиявших на данный выбор, следует отметить то, что данный сервис оптимизирован для русскоязычных пользователей, соответствует требованиям конфиденциальности и безопасности данных, а также предоставляет возможность большего количества запросов по генерации текста в час.

Для обеспечения высокой производительности, гибкости разработки и удобства поддержки в качестве целевой платформы рассматривается реализация, выполненная с использованием следующего технологического стека:

- язык программирования Java и фреймворк Spring для реализации бэкенда;
- фреймворк Flutter и язык программирования Dart – для клиентской части приложения;
- СУБД MongoDB для хранения информации о вопросах, тестах и их прохождениях и PostgreSQL для хранения информации о пользователях.

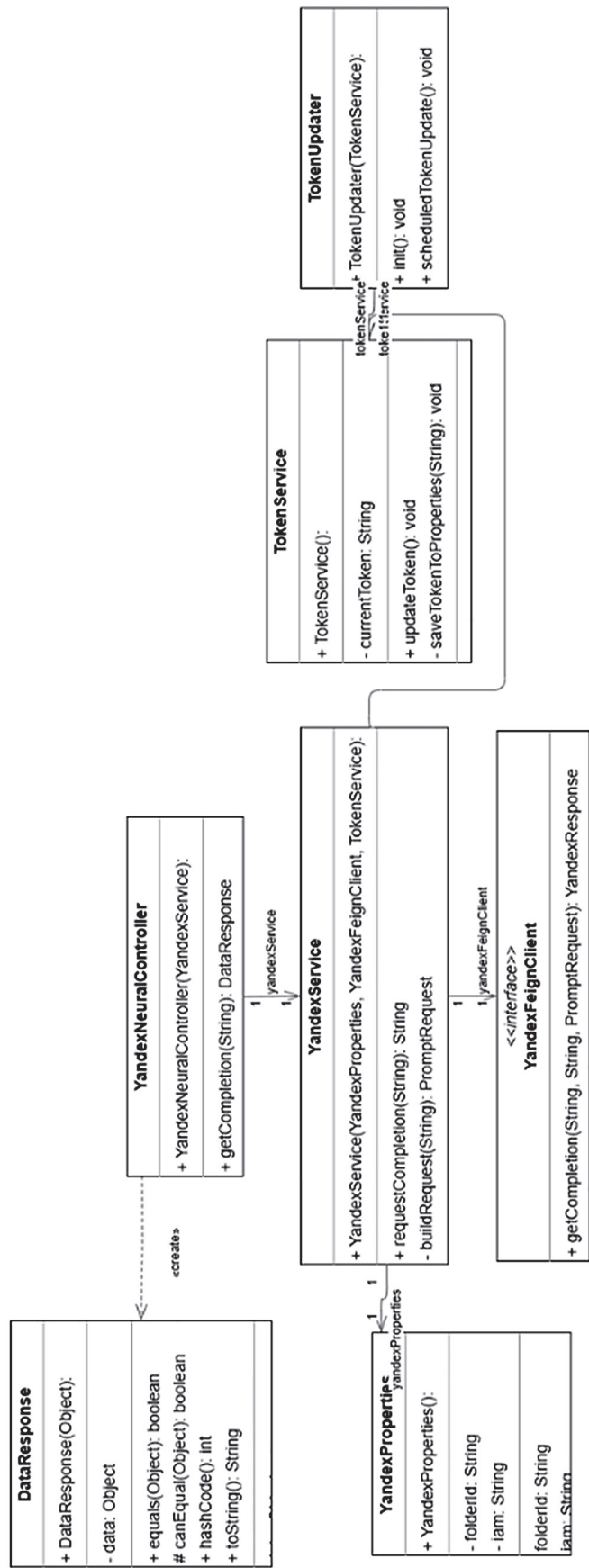


Рис. 1. Диаграмма классов для микросервиса yandex-service
Источник: составлено авторами

При реализации модуля контроля знаний, входящего в состав программно-информационной системы создания онлайн-курсов, была выбрана микросервисная архитектура – архитектурный подход к разработке, который предполагает разбиение приложения на маленькие и независимые компоненты, каждый из которых выполняет определенную функцию [14]. Эти компоненты могут взаимодействовать друг с другом через API, что позволяет разрабатывать приложения более гибко и масштабируемо [15]. Таким образом, для интеграции ранее разработанного модуля контроля знаний с Yandex GPT API для выдачи персонализированных рекомендаций по результатам прохождения тестирования был реализован отдельный микросервис `yandex-service`, диаграмма классов которого представлена на рис. 1. Данная диаграмма демонстрирует основные классы, реализованные авторами для обеспечения успешного подключения к YandexGPT API и настройки обработки запросов к данной генеративной текстовой модели.

Рассмотрим основные классы данного микросервиса, показанные на рис. 1:

1. `DataResponse`. Данный класс применяется для передачи универсальных ответов и позволяет инкапсулировать объект результата.

2. `YandexNeuralController`. Данный контроллер реализует получение входящих запросов, выполняет их обработку и позволяет в дальнейшем получать AI-ответы по сгенерированным запросам (`PromptRequest`). Также данный контроллер используется для обеспечения взаимодействия клиентской части разработанного приложения с генеративной текстовой моделью YandexGPT и осуществляет управление маршрутизацией запросов.

3. `YandexService`. Данный класс является ключевым классом для организации взаимодействия с YandexGPT API. Он позволяет обеспечить вызов методов API, а также выполнить обработку данных и реализовать взаимодействие с другими модулями системы, обеспечивая централизованное управление сервисом.

4. `YandexProperties`. Данный класс отвечает за конфигурацию сервиса, позволяя указать параметры подключения и настройки для осуществления интеграции с YandexGPT API.

5. `YandexFeignClient`. Данный интерфейс напрямую взаимодействует с классом `YandexService`, представляя функциональность удаленной службы. Использование интерфейса упрощает создание и использование API в архитектуре микросервисов,

а также обеспечивает улучшенную обработку ошибок, включая поддержку пользовательской обработки ошибок и повторных попыток.

6. `TokenService`. Данный класс используется для управления токенами, включая их создание, прохождение процедуры валидации и предоставление доступа к защищенным методам YandexGPT API.

7. `TokenUpdater`. Автоматически обновляет токены при их истечении или по требованию. Интегрируется с хранилищем токенов и сервисом авторизации для обеспечения бесперебойного доступа.

Результаты исследования и их обсуждение

Теперь коротко рассмотрим демонстрацию разработанного функционала после интеграции модуля контроля знаний с генеративной текстовой моделью YandexGPT. По завершению теста обучающемуся отображается отчет о результатах тестирования с указанием корректности полученных ответов (рис. 2). Как показано на рис. 2, вопросы, на которые обучающийся дал неверный ответ, подсвечиваются красным цветом и содержат дополнительную кнопку в виде знака вопроса. При нажатии на данную кнопку у обучающегося появляется возможность запросить индивидуальную рекомендацию для выбранного вопроса (рис. 3). Также обучающемуся предоставляется возможность просмотреть общие рекомендации, сформированные сразу по всему пройденному тесту в целом, при нажатии на кнопку «Что у меня неправильно?» в нижней части отчета (рис. 4). При получении общих рекомендаций по тесту YandexGPT предоставляет краткую сводку по каждому вопросу, на который обучающийся ответил неверно.

Для проверки точности генерируемых YandexGPT индивидуальных рекомендаций по результатам тестирования были оценены результаты ее работы для двух курсов по изучению языков высокого уровня (JavaScript, Python). Всего были проанализированы результаты тестирования 23 студентов по 15 темам данных курсов. Было выявлено, что YandexGPT корректно генерирует рекомендации по теоретическим вопросам данной тематики в 94% случаях. При генерации рекомендаций к вопросам практического характера на написание кода уровень достоверности YandexGPT немного снизился и зависел от конкретного языка программирования, на котором требовалось предложить решение задачи. Так, доля успешных решений, сгенерированных на языке Python, составила около 85%, на языке JavaScript – около 77%.

Процент от от общего количества баллов: 71.43%

Расположите операторы сверху вниз в порядке увеличения их приоритета при выполнении арифметических операций в Python. ?

Примечание. Операторы с самым низким приоритетом должны быть в самом верху, а с самым высоким приоритетом – в самом низу.

Что покажет приведённый ниже код?

```
print(2 * 3**2 - 1)
```

напишите функцию, возвращающую n число числа Фибоначчи ?

Дано пятизначное число, которое хранится в переменной n. В переменных a, b, c, d, e хранятся:

a – число десятков тысяч (первая цифра),
b – число тысяч (вторая цифра),
c – число сотен (третья цифра),
d – число десятков (четвертая цифра),
e – число единиц (пятая цифра).
Установите соответствие между элементами первого и второго столбцов.

Что покажет приведённый ниже код?

```
print((-10)**2)
```

Что у меня неправильно?

пока пусто

Рис. 2. Пример отображения отчета о результатах тестирования
Источник: составлено авторами

Исходя из вышесказанного, авторы считают, что полученные результаты количественно подтверждают успешность применения предложенного подхода к анализу результатов тестирования, а генеративная текстовая модель YandexGPT может быть успешно использована для анализа результатов тестирования обучающихся в модуле контроля знаний и повышения эффективности образовательного процесса в целом за счет формирования персонализированных рекомендаций.

Для подтверждения положительного влияния внедрения технологий генеративного искусственного интеллекта в модуль контроля знаний на эффективность образовательного процесса авторами были

проведены следующие дополнительные исследования. Во-первых, среди студентов, проходивших тестирования по темам курсов с внедренной системой обратной связи, генерирующей персонализированные рекомендации, был проведен опрос, чтобы оценить, насколько им понравился опыт взаимодействия с генеративной текстовой моделью в модуле контроля знаний. По результатам опроса было выявлено, что 74% респондентов считают, что полученные персонализированные рекомендации помогли им сократить время на поиск ошибок и пробелов в знаниях и тем самым способствовали более быстрому усвоению материала в проблемных областях.

Расположите операторы сверху вниз в порядке увеличения их приоритета при выполнении арифметических операций в Python.

Примечание. Операторы с самым низким приоритетом должны быть в самом верху, а с самым высоким приоритетом – в самом низу.

Чтобы правильно расположить операторы в порядке увеличения их приоритета при выполнении арифметических операций в Python, необходимо знать таблицу приоритетов операторов. Вероятно, вы допустили ошибки в порядке расстановки операторов. Возможно, вы не учли, что некоторые операторы имеют одинаковый приоритет и выполняются слева направо, а также не обратили внимание на то, что у других операторов приоритет выше.

Повторите материал о приоритетах операторов в Python. Вот несколько ресурсов, которые могут помочь:

- [Документация Python](#) содержит подробную таблицу приоритетов операторов.
- На сайте [Python.org](#) есть множество материалов и руководств по изучению языка программирования Python.
- Существуют онлайн-курсы и видеоуроки, посвящённые изучению Python, где также можно найти информацию о приоритетах операторов.

Рис. 3. Пример генерации рекомендаций для одного вопроса, на который обучающийся ответил неверно
Источник: составлено авторами

0. Порядок операторов по приоритету при выполнении арифметических операций в Python:

В Python существует определённый порядок выполнения операций, который определяется их приоритетом. Операторы с более высоким приоритетом выполняются раньше операторов с более низким приоритетом. Вот основные арифметические операторы в порядке увеличения их приоритета:

- Сложение и вычитание (+, -).
- Умножение, деление, целочисленное деление и остаток от деления (*, /, //, %).
- Возведение в степень (**).

Более подробную информацию о приоритете операторов в Python можно найти в [документации](#).

1. Функция для вычисления n-го числа Фибоначчи:

Числа Фибоначчи – это последовательность чисел, в которой каждое следующее число равно сумме двух предыдущих. Первые два числа в последовательности равны 0 и 1 (иногда используют 1 и 1). Вот пример функции на Python, которая возвращает n-е число Фибоначчи:

```
def fibonacci(n):  
    if n <= 0:  
        return "Ошибка: n должно быть положительным целым числом"  
    elif n == 1:  
        return 0  
    elif n == 2:  
        return 1  
    else:  
        a, b = 0, 1  
        for _ in range(2, n):  
            a, b = b, a + b  
        return b
```

Эта функция использует цикл for для вычисления n-го числа Фибоначчи. Она начинается с первых двух чисел последовательности (0 и 1) и затем вычисляет следующие числа, пока не достигнет n-го числа. Более подробную информацию о числах Фибоначчи и их вычислении можно найти в [википедии](#).

Рис. 4. Пример генерации рекомендаций по всем вопросам, на которые обучающийся ответил неверно
Источник: составлено авторами

Также следует отметить, что 39% опрошенных считают, что их результаты прохождения тестов по темам курсов постепенно улучшались за счет получения рекомендаций, что повлияло в целом на их успеваемость по данным курсам. Во-вторых, было организовано прохождение тех же тестов другой группой студентов, состоящей из 20 чел., без возможности получения индивидуальных рекомендаций с помощью генеративной текстовой модели. В результате сравнения результатов освоения курсов двумя группами обучающихся было

выявлено, что средний процент прохождения тестов первой группы превысил аналогичный показатель второй группы на 12%. Таким образом, полученные результаты демонстрируют успешность повышения эффективности образовательного процесса за счет внедрения в модуль контроля знаний обучающихся технологий генеративного искусственного интеллекта.

Фрагменты кода, используемого для подготовки запросов через YandexService и настройку контроллера YandexNeuralController, приведены на рисунке 5.

```
....
@RestController
@RequiredArgsConstructor
@RequestMapping(«/api/v1/course/yandex/neural»)
public class YandexNeuralController {

    private final YandexService yandexService;

    @GetMapping(«/completion»)
    public DataResponse getCompletion(@RequestParam String prompt) {
        return new DataResponse(yandexService.requestCompletion(prompt));
    }
}
....
@Service
@RequiredArgsConstructor
public class YandexService {
    private final YandexProperties yandexProperties;
    private final YandexFeignClient yandexFeignClient;
    private final TokenService tokenService;
    public String requestCompletion(String prompt) {
        PromptRequest promptRequest = buildRequest(prompt);
        return yandexFeignClient.
            getCompletion(«Bearer “ + tokenService.getCurrentToken(),
                yandexProperties.getFolderId(),
                promptRequest
            ).getResult().getAlternatives().stream().findFirst()
            .map(e->e.getMessage().getText()).orElse(null);
    }
    private PromptRequest buildRequest(String text) {
        return PromptRequest.builder()
            .modelUri(«gpt://b1gebvniis9kf6dhlm57/yandexgpt/rc»)
            .completionOptions(
                PromptRequest.CompletionOptions.builder()
                    .maxTokens(500)
                    .temperature(0.3)
                    .build()
            )
            .messages(Arrays.asList(
                PromptRequest.Message.builder()
                    .role(«user»)
                    .text(text)
                    .build()
            ))
            .build();
    }
}
....
```

Рис. 5. Фрагменты кода, используемого для подготовки запросов через YandexService и настройку контроллера YandexNeuralController
Источник: составлено авторами

Заключение

Основным результатом работы является внедрение технологий генеративного искусственного интеллекта (на примере использования генеративной текстовой модели YandexGPT) в разработанный модуль контроля знаний обучающихся онлайн-курсов для персонализации анализов результатов пройденного тестирования и получения индивидуальных рекомендаций на основе выполненного анализа. В результате внедрения данного модуля обучающиеся получают возможность не только оценить уровень своих знаний при прохождении тестирования, но и получить персонализированную обратную связь, так как Yandex GPT выдает им индивидуальные сгенерированные рекомендации, содержащие список тем и материалов, которые необходимо проработать обучающемуся для более успешного прохождения испытаний. В случае, когда обучающийся не смог написать код, отвечающий требованиям, YandexGPT даже способна предложить собственное решение простых типов задач.

Обобщая вышесказанное, можно сделать вывод о том, что внедрение генеративных текстовых моделей может быть использовано для модернизации образовательных процессов в целом, а разработанный авторами модуль с использованием YandexGPT может применяться как инструмент персонализации обучения в частности. Авторы планируют дальнейшее развитие данного модуля с использованием других технологий искусственного интеллекта для анализа данных результатов тестирования и создания адаптивных учебных программ онлайн-обучения на основе выполненного анализа.

Список литературы

1. Галагузова М.А., Перекальский И.Н. Трансформация образования с внедрением искусственного интеллекта: постановка проблемы // Ценности и смыслы. 2024. № 1. С. 84–94. URL: <https://cyberleninka.ru/article/n/transformatsiya-obrazovaniya-s-vnedreniem-iskusstvennogo-intellekta-postanovka-problemy> (дата обращения: 08.06.2025). DOI: 10.24412/2071-6427-2024-1-84-94.
2. Шамсутдинова Т.М. Проблемы и перспективы применения нейронных сетей в сфере образования // Открытое образование. 2022. № 6. С. 4–10. URL: <https://cyberleninka.ru/article/n/problemy-i-perspektivy-primeneniya-neyronnyh-setey-v-sfere-obrazovaniya> (дата обращения: 08.06.2025). DOI: 10.21686/1818-4243-2022-6-4-10.
3. Лютова Л.И., Гончар Л.В., Частухина Т.В. Возможности применения технологий искусственного интеллекта при изучении иностранного языка в вузе // Международный научно-исследовательский журнал. 2024. № 7. URL: <https://research-journal.org/archive/7-145-2024-july/10.60797/IRJ.2024.145.182> (дата обращения: 08.06.2025). DOI: 10.60797/IRJ.2024.145.182.
4. Бутреева А.С. Проблемы и перспективы применения приложений на базе искусственного интеллекта в целях персонализации обучения в системе высшего образования // Мир науки, культуры, образования. 2025. № 1. С. 131–133. URL: <https://cyberleninka.ru/article/n/problemy-i-perspektivy-primeneniya-prilozheniy-na-baze-iskusstvennogo-intellekta-v-tselyah-personifikatsii-obucheniya-v-sisteme-vyshego-obrazovaniya> (дата обращения: 08.06.2025). DOI: 10.24412/1991-5497-2025-1110-131-133.
5. Короткевич М.А. Функции и примеры внедрения больших языковых моделей в современных медиакommunikациях // Наука и школа. 2025. № 2. С. 42–52. URL: <https://cyberleninka.ru/article/n/funktsii-i-primery-vnedreniya-bolshih-yazykovykh-modeley-v-sovremennykh-mediakommunikatsiyah> (дата обращения: 08.06.2025). DOI: 10.31862/1819-463X-2025-2-42-52.
6. Кузьменко Е.Л., Белоусова Т.М., Лещенко Е.М. Проблемы и риски интеграции искусственного интеллекта в высшее образование // Регион: системы, экономика, управление. 2024. № 2. С. 164–168. URL: <https://cyberleninka.ru/article/n/problemy-i-riski-integratsii-iskusstvennogo-intellekta-v-vysshee-obrazovanie> (дата обращения: 08.06.2025). DOI: 10.22394/1997-4469-2024-65-2-164-168.
7. Буткина А.А., Егоров М.С. Проектирование модуля контроля знаний для программно-информационной системы создания онлайн-курсов // Современные наукоемкие технологии. 2024. № 7. С. 10–15. URL: <https://top-technologies.ru/ru/article/view?id=40078> (дата обращения: 08.06.2025). DOI: 10.17513/snt.40078.
8. Манакова Е.И., Хачетлов Б.А. Анализ генеративных моделей искусственного интеллекта: преимущества и перспективы применения // Актуальные вопросы науки и практики: сборник научных статей по материалам XII Международной научно-практической конференции (Уфа, 20 июня 2023 г.). Ч. 2. Уфа: ООО «Научно-издательский центр «Вестник науки», 2023. С. 247–262. URL: https://perviy-vestnik.ru/wp-content/uploads/2023/07/2023-K-387-2-06_23.pdf (дата обращения: 29.08.2025).
9. Sherstinova T., Firsanova V., Novoseltseva A., Megre M., Savchenko E. Automated Speech Act Annotation in a Russian Spoken Corpus Using Large Language Models: A Comparative Study // Proceedings – Conference of Open Innovations Association, FRUCT. 2024. № 36. P. 912–920. URL: <https://www.fruct.org/files/publications/volume-36/acm36/She.pdf> (дата обращения: 24.08.2025).
10. Леохин Ю.Л., Фатхулин Т.Д., Кожанов М.С. Анализ и исследование применения нейросетевых технологий для генерации программного кода // Вестник РГПУ. 2024. № 87. С. 41–53. URL: <https://vestnik.rsreu.ru/images/archive/2024/87/2.2.-.pdf> (дата обращения: 22.08.2025). DOI: 10.21667/1995-4565-2024-87-41-53.
11. Долгошеева Д.В. YandexGPT – сервис для создания текста при помощи искусственного интеллекта // Постулат. 2024. № 1 (99). С. 91. URL: <https://www.e-postulat.ru/index.php/Postulat/article/view/5424/5510> (дата обращения: 10.08.2025).
12. Ворона А.А. Применение технологий искусственного интеллекта: современные реалии и перспективы // Ученые записки Санкт-Петербургского имени В.Б. Бобкова филиала Российской таможенной академии. 2023. № 4. URL: <https://cyberleninka.ru/article/n/primenenie-tehnologiy-iskusstvennogo-intellekta-sovremennye-realii-i-perspektivy> (дата обращения: 08.06.2025).
13. Пеньшина Д.С., Заботкина Е.М. YandexGPT 3 – новое поколение больших языковых моделей Яндекса // Научный аспект. 2024. Т. 15. № 5. С. 1967–1975. URL: <https://na-journal.ru/5-2024-informacionnye-tehnologii/11695-yandexgpt-3-novoe-pokolenie-bolshih-yazykovykh-modeley-yandeksa> (дата обращения: 10.08.2025).
14. Гольчевский Ю.В., Ермоленко А.В. Актуальность использования микросервисов при разработке информационных систем // Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика. 2020. № 2. С. 25–36. URL: https://vestnik-mmi.syktu.ru/issue/35/v35_3.pdf (дата обращения: 24.08.2025).
15. Балес А.И. Унифицированная модель данных и ее применение в микросервисной архитектуре // Современные информационные технологии и ИТ-образование. 2020. Т. 16. № 2. С. 416–425. URL: <http://sitito.cs.msu.ru/index.php/SITITO/article/view/658/689> (дата обращения: 08.06.2025). DOI: 10.25559/SITITO.16.202002.416-425.

Конфликт интересов: Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest: The authors declare that there is no conflict of interest.