



ИНТЕГРАЦИЯ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ В ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ СИМУЛЯЦИИ КАНАЛОВ ПИСЬМЕННОЙ КОММУНИКАЦИИ¹

Войтко Даниил Алексеевич,

студент кафедры автоматизации и процессов управления
Санкт-Петербургский государственный электротехнический
университет «ЛЭТИ» имени В.И. Ульянова (Ленина)
Email: mediumm333@gmail.com

Притчин Андрей Андреевич,

студент кафедры автоматизации и процессов управления
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени
В.И. Ульянова (Ленина)
Email: sedartes07@gmail.com

Аннотация

В данном исследовании разработано веб-приложение для изучения английского языка, ориентированное на тренировку письменной коммуникации в реалистичных цифровых сценариях (чат, электронное письмо, обращение в поддержку, эссе и отзывы). В отличие от универсальных чат-ботов и стандартных лингвистических тренажеров, предложенная система использует интеграцию с API большой языковой модели DeepSeek для динамической генерации заданий и автоматической проверки ответов по четырем критериям: смысл, грамматика, лексика и соответствие каналу общения. Архитектура приложения построена на стеке React, FastAPI и PostgreSQL с использованием жесткого JSON-формата для структурирования ответов нейросети. Тестирование подтвердило корректность работы пользовательских сценариев, устойчивость к граничным ситуациям и эффективность генерации дополнительных заданий (follow-up) на основе выявленных ошибок. Предложенная структура представляет собой гибкий образовательный инструмент, а дальнейшая работа включает расширение базы каналов и внедрение административной аналитики.

Ключевые слова: веб-приложение, изучение английского языка, письменная коммуникация, большие языковые модели, LLM, DeepSeek, генерация заданий, автоматическая проверка, FastAPI, React.

DEVELOPMENT OF A WEB APPLICATION FOR PRACTICING WRITTEN COMMUNICATION IN ENGLISH USING LARGE LANGUAGE MODELS

¹ Научный руководитель: **Туральчук Константин Александрович**, кандидат технических наук, доцент кафедры автоматизации и процессов управления Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени В.И. Ульянова (Ленина)
Email: katuralchuk@etu.ru

Scientific supervisor: **Turalchuk Konstantin Anatolyevich**, Candidate of Technical Sciences, Associate Professor at the Department of Automation and Control Processes, Saint Petersburg Electrotechnical University "LETI"

Voytko Daniil Alexeevich,

Student, Department of Automation and Control Processes Saint Petersburg Electrotechnical University "LETI"

Email: mediumm333@gmail.com

Pritchkin Andrey Andreevich,

Student, Department of Automation and Control Processes Saint Petersburg Electrotechnical University "LETI"

Email: sedartes07@gmail.com

ABSTRACT

This study develops a web application for learning English, focused on practicing written communication in realistic digital scenarios (chat, email, support request, essay, and review). Unlike universal chatbots and standard linguistic trainers, the proposed system integrates the DeepSeek Large Language Model API to dynamically generate tasks and automatically evaluate user responses based on four criteria: meaning, grammar, vocabulary, and channel appropriateness. The application architecture is built on the React, FastAPI, and PostgreSQL stack, utilizing a strict JSON format to structure neural network responses. Testing confirmed the correct operation of user scenarios, resilience to edge cases, and the effectiveness of generating follow-up tasks based on identified mistakes. The proposed framework offers a flexible educational tool, with future work including the expansion of communication channels and the implementation of administrative analytics.

Keywords: web application, English language learning, written communication, large language models, LLM, DeepSeek, task generation, automated evaluation, FastAPI, React.

ВВЕДЕНИЕ

Рынок цифрового образования претерпел значительную трансформацию благодаря распространению мобильных приложений и интерактивных платформ. Исследования последних лет показывают, что интеграция генеративного искусственного интеллекта в образовательный процесс создает как значительные возможности, так и серьезные методологические вызовы [1, 2]. В частности, Kasneci et al. отмечают, что успешное применение LLM в образовании требует четкой методической рамки, предотвращающей фрагментарность учебного процесса [1].

Традиционные лингвистические платформы обеспечивают регулярную практику, однако преимущественно ориентируются на лексику, грамматику и шаблонные упражнения [3]. Значительная часть повседневного и профессионального общения в настоящее время реализуется в текстовой форме (электронная почта, мессенджеры, обращения в поддержку), что требует от обучающегося не только знания правил, но и умения применять язык в разных коммуникативных условиях с учетом стиля и формальности [4].

Появление универсальных GPT-сервисов предоставило пользователям мощный инструмент для свободной языковой практики [5]. Однако их использование в качестве основы системного обучения сопряжено с рядом ограничений: отсутствие устойчивого учебного маршрута, единого формата заданий, объективных критериев оценки и истории

прогресса. Пользователю приходится самостоятельно проектировать промпты, что увеличивает когнитивную нагрузку и снижает методическую ценность процесса обучения [6].

Актуальность данной работы определяется необходимостью преодоления разрыва между неструктурированными возможностями LLM и педагогически обоснованной структурой учебного процесса. Современные исследования в области автоматизированной оценки письменных работ (Automated Essay Scoring) [7] подтверждают эффективность многокритериального подхода к анализу текстов, однако их интеграция в специализированные учебные платформы с динамической генерацией контента остается недостаточно изученной.

ЦЕЛЬ ИССЛЕДОВАНИЯ

Целью исследования является разработка веб-приложения для тренировки письменной коммуникации на английском языке, использующего интеграцию с большой языковой моделью в качестве встроенного интеллектуального инструмента для динамической генерации заданий, многокритериальной оценки ответов и формирования персонализированной обратной связи в рамках заранее спроектированной учебной структуры.

Для достижения поставленной цели решались следующие задачи: анализ предметной области и существующих решений; проектирование клиент-серверной архитектуры с интеграцией LLM; разработка механизма структурирования ответов нейросети в формате JSON; реализация модуля многокритериальной оценки; проведение комплексного тестирования системы.

МАТЕРИАЛЫ И МЕТОДЫ ИССЛЕДОВАНИЯ

Архитектура и технологический стек

Разработанное приложение построено на клиент-серверной архитектуре, что позволяет разделить пользовательский интерфейс, бизнес-логику и взаимодействие с внешней языковой моделью. Клиентская часть реализована как одностраничное веб-приложение (SPA) с использованием фреймворка React и инструмента сборки Vite. Маршрутизация и защита внутренних страниц реализованы с использованием JWT-токенов. Серверная часть разработана на базе FastAPI (Python) и выступает центральным координатором бизнес-логики: обрабатывает авторизацию, управляет доступом к реляционной базе данных PostgreSQL (через ORM SQLAlchemy) и инкапсулирует все взаимодействия с внешним API языковой модели. Такой подход позволяет изолировать клиентскую часть от критически важных механизмов, скрыть ключи доступа API и централизовать валидацию данных с помощью Pydantic-схем. В качестве интеллектуального ядра системы выбрана модель DeepSeek (deepseek-v4-flash). Выбор обусловлен сочетанием достаточного качества текстовой генерации, нативной поддержки возврата структурированных данных в формате JSON и относительно низкой стоимости использования по сравнению с коммерческими аналогами [5, 8].

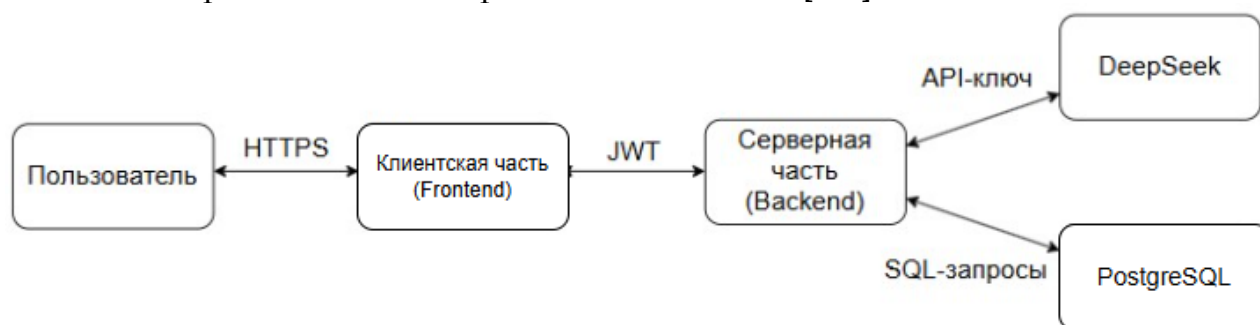


Рисунок 1 – Схема взаимодействия компонентов системы

Интеграция с большой языковой моделью

Ключевой особенностью системы является отказ от статического набора упражнений в пользу динамической генерации контента. При выборе пользователем уровня языка и канала общения backend формирует специализированный системный промпт. Промпт жестко ограничивает модель рамками выбранного формата: для комментария требуется краткая реакция в 1-2 предложения, для письма – соблюдение структуры и формального тона. Для обеспечения предсказуемости и автоматической обработки ответов модели используется параметр `response_format: {"type": "json_object"}`. Это гарантирует, что нейросеть возвращает структурированный объект, содержащий ситуацию, инструкции, список полезных слов и пояснения. На стороне сервера применяется дополнительная нормализация и валидация JSON-ответов через функцию `normalize_model_payload(...)`, которая:

- обрабатывает ответ в зависимости от типа содержимого;

- выполняет парсинг JSON;

- проверяет, что результат является объектом, а не произвольным списком или строкой;

- выбрасывает серверную ошибку (HTTP 502) при некорректном формате.

После нормализации результат валидируется через Pydantic-схемы (`TaskGenerateResponse`, `TaskCheckResponse`), что исключает появление некорректных данных на клиенте и обеспечивает предсказуемость интерфейса.

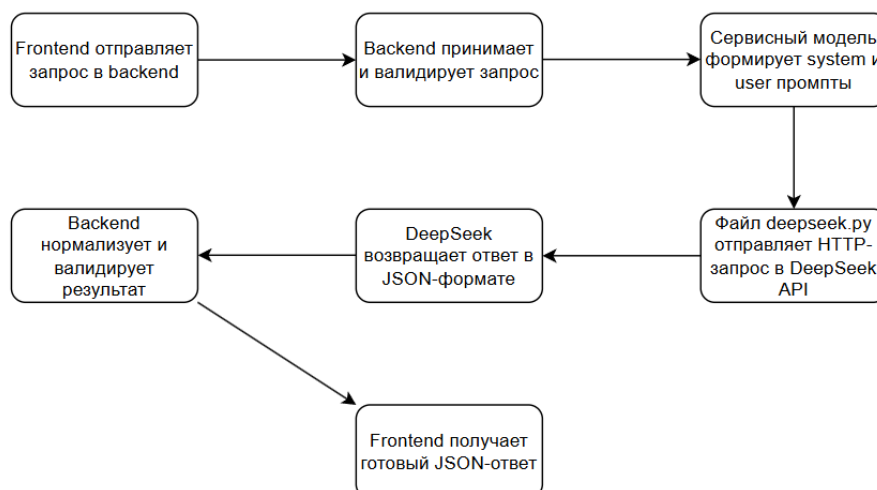


Рисунок 2 – Схема использования DeepSeek в приложении

Листинг 1 – Пример системного промпта

You generate vocabulary training blocks for a Russian-speaking learner of English.
Return only valid JSON.

The learner level is {level}.

The theme in English is "{theme_en}".

The theme in Russian is "{theme_ru}".

Rules:

Generate exactly 6 English words or short phrases that fit the chosen theme and level.

The words must be genuinely useful for written communication and everyday vocabulary.

For each word, provide one correct Russian translation.

For each word, provide 1 or 2 short natural example sentences in English.

Generate exactly 6 sentence items for stage two.

Each stage two sentence must contain one blank written as `___` and must be solvable with one of the generated words.

Use each generated word exactly once in stage two.

The order of stage two sentences must not mirror the order of the words list.

Mix the stage two items so the learner cannot solve the task just by following the same sequence.

Keep A1–A2 vocabulary simple and common.

Allow B1–B2 to use more abstract, academic, or professional vocabulary.

Do not include markdown.

Структура данных и логика обучения

Проверка пользовательского ответа осуществляется по четырем независимым критериям (от 0 до 10 баллов):

Смысл (соответствие коммуникативной задаче);

Грамматика;

Словарь (лексическая уместность);

Соответствие каналу (стиль и формат).

На основе полученных оценок система автоматически выявляет «слабые» критерии (оценка ниже 7) и инициирует генерацию дополнительного задания (follow-up), направленного на точечную отработку выявленных ошибок. Результаты выполнения сохраняются в базе данных (таблицы users, task_history, courses), что позволяет пользователю отслеживать историю прогресса. Дополнительно реализованы режимы курсового прохождения (с фиксированным набором обязательных каналов) и двухэтапной тренировки лексики.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Описание интерфейса разработанного приложения

Разработанный интерфейс организован по принципу единой layout-структуры: верхняя панель с названием платформы, боковое меню с основными разделами (главная, история, курсы, тренировка, о проекте) и центральная рабочая область.

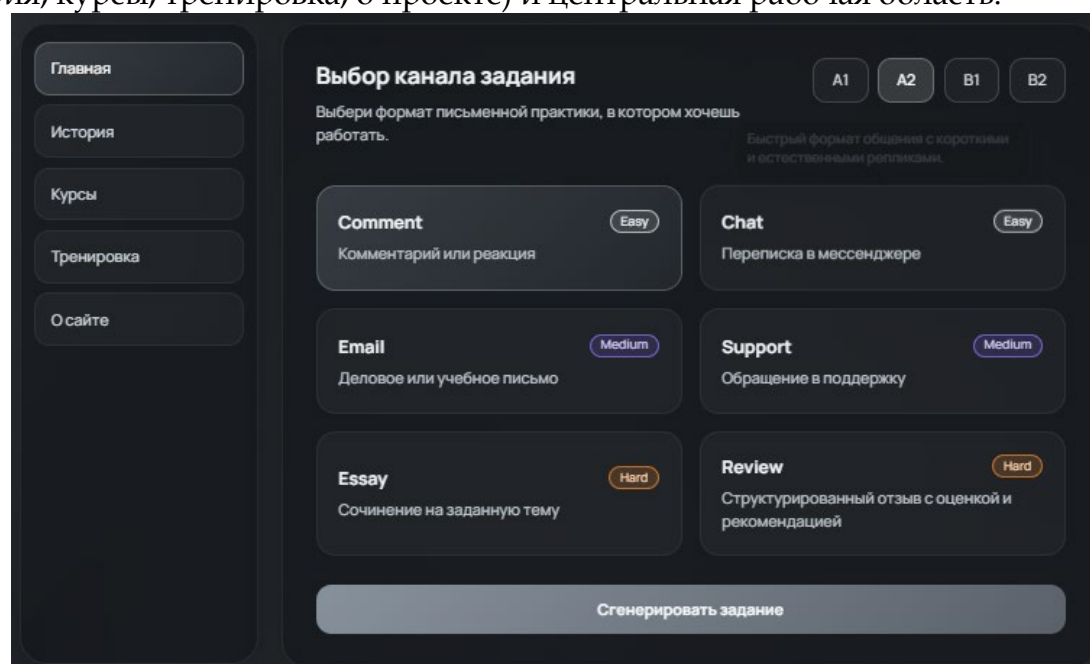


Рисунок 1 — Главная страница выбора канала общения.

На рисунке отображен интерфейс DashboardPage с выпадающими списками выбора уровня языка (A1–B2) и типа канала письменной коммуникации.

ESSAY B2

online education vs traditional education

Situation

You are a student who has experienced both online and traditional classroom learning. Your university is considering expanding its online course offerings, and the student council has asked for opinions.

Task

Write a short essay (about 5-7 sentences) comparing online education and traditional education. Discuss at least one advantage and one disadvantage of each, and give your personal opinion on which you prefer and why.

Helpful words

- flexibility
- face-to-face interaction
- self-discipline
- learning environment
- cost-effective

Пояснение на русском

Ответ

Напиши ответ на английском и отправь его на автоматическую проверку.

Online education offers flexibility and is cost-effective, but it lacks face-to-face interaction. Traditional education has a better learning environment but requires more self-discipline from students. I prefer online because it saves time and money.

Отправить на проверку

Рисунок 2 — Интерфейс проверки ответа и обратной связи.

На рисунке показана страница TaskPage с полем ввода ответа пользователя, числовыми оценками по четырём критериям, текстовыми комментариями на двух языках и кнопкой генерации follow-up задания.

Meaning 8/10 Grammar 9/10 Vocabulary 7/10 Channel 7/10

Feedback in English

The answer addresses the task and gives advantages/disadvantages, but it is a bit short and could be more developed. The structure is clear, but the essay lacks a conclusion connecting ideas. Also, the point about self-discipline is misattributed to traditional education.

Mistakes in English

- Self-discipline is more needed in online education, not traditional.
- Answer is slightly short for an essay; could add a concluding sentence.

Improved answer

Online education offers flexibility and cost-effectiveness, but it lacks face-to-face interaction, which can be a drawback for some learners. Traditional education provides a structured learning environment and direct interaction with peers and teachers, yet it can be less flexible and more expensive. However, online education requires strong self-discipline to stay motivated, while traditional education often provides external structure. In my opinion, I prefer online education because it saves time and money, and with good time management, the lack of face-to-face interaction can be mitigated. Ultimately, the best choice depends on individual learning styles and circumstances.

Пояснение на русском

Сгенерировать задание по ошибкам

Рисунок 3 – Обратная связь по заданию

На рисунке продемонстрирован интерфейс формирования обратной связи с возможностью создать задание по ошибкам, допущенным при ответе.

Методология тестирования

Для подтверждения работоспособности предложенной архитектуры было проведено комплексное тестирование в период с 15 по 17 мая 2026 года. Тестирование выполнялось авторами работы на развернутой версии приложения.

В процессе тестирования использовались инструменты:

Swagger UI (интерактивная документация FastAPI, доступная по адресу /docs) — для прямой проверки REST-API и верификации корректности JSON-ответов от языковой модели DeepSeek;

Ручное тестирование в браузере — для проверки пользовательских сценариев через интерфейс;

Прямой контроль состояния PostgreSQL — для верификации сохранения данных.

Тестирование охватило 24 тестовых сценария, распределенных по трём категориям: функциональное тестирование (12 сценариев) — проверка основных пользовательских сценариев через интерфейс, интеграционное тестирование (6 сценариев) — проверка взаимодействия backend, PostgreSQL и DeepSeek API и тестирование граничных условий (6 сценариев) — проверка обработки некорректных данных и нарушений бизнес-логики.

Результаты тестирования

Среднее время отклика системы от момента нажатия пользователем кнопки «Сгенерировать» до получения готового задания составило 10–12 секунд, что находится в пределах приемлемого диапазона для комфортного учебного процесса. Основную долю времени занимал сетевой обмен с DeepSeek API и обработка JSON-ответа на стороне backend.

Ниже приведены репрезентативные примеры тестовых сценариев по каждой категории с описанием ожидаемого и фактического результатов.

Таблица 1 – Примеры функционального тестирования пользовательских сценариев

ID	Сценарий	Ожидаемый результат	Фактический результат
F-1	Регистрация нового пользователя	Учетная запись будет создана, пользователь получит доступ к системе	Регистрация прошла успешно, переход в защищенную часть системы выполнен
F-3	Генерация свободного задания	Система отобразит новое сгенерированное задание с выбранными параметрами	Задание корректно сформировано и отображено в интерфейсе
F-8	Выполнение курсового задания	Система проверяет ответ и обновляет прогресс курса	Результат проверки получен, прогресс курса изменен корректно

Таблица 2 - Примеры интеграционного тестирования взаимодействия компонентов

ID	Сценарий	Ожидаемый результат	Фактический результат
I-1	Обращение к backend DeepSeek при генерации задания	Backend возвращает корректный JSON с полями channel, level, topic, situation_en, instructions_en, helpful_words_en	JSON-ответ корректен, все обязательные поля присутствуют, задание соответствует каналу и уровню

I-2	Взаимодействие backend, DeepSeek и PostgreSQL при проверке задания	В ответе присутствует history_entry_id, в таблице task_history создается новая запись	Ответ получен корректно, history_entry_id возвращен, запись в task_history создана
I-6	Генерация follow-up задания после слабого результата	Backend возвращает задание с is_follow_up = true, в task_history обновляются follow-up поля	Дополнительное задание создано корректно, данные в истории обновились

Таблица 3 - Примеры тестирования граничных и ошибочных сценариев

ID	Сценарий	Ожидаемый результат	Фактический результат
N-2	Вход с неверным паролем	Система не авторизует пользователя и возвращает ошибку авторизации	Backend вернул HTTP-ошибку 401 Unauthorized, токен не выдан
N-3	Обращение к защищенному маршруту без JWT-токена	Доступ к функции запрещается	Backend вернул HTTP-ошибку 401 Unauthorized, операция не выполнена
N-4	Передача невалидных параметров при генерации задания	Запрос отклоняется на этапе валидации	Backend вернул HTTP-ошибку 422 Unprocessable Entity

В совокупности все 24 тестовых сценария завершились успешно (100% прохождения). Функциональное тестирование подтвердило корректность всех основных пользовательских сценариев: регистрации, генерации заданий, проверки ответов нейросетью, создания follow-up заданий и прохождения курсов. Интеграционное тестирование продемонстрировало согласованность сохранения данных: результаты проверки успешно записывались в таблицу task_history, а прогресс курсов корректно обновлялся в таблице courses. Проверка граничных и ошибочных сценариев показала, что система предсказуемо возвращает соответствующие HTTP-ошибки (401 Unauthorized, 422 Unprocessable Entity, 409 Conflict, 400 Bad Request), не допуская нарушения целостности базы данных и обхода бизнес-логики.

Обсуждение результатов

Полученные экспериментальные результаты подтверждают основные гипотезы данного исследования. Во-первых, интеграция LLM в строгий методический каркас позволяет решить проблему фрагментарности универсальных чат-ботов, обеспечивая пользователя структурированным учебным маршрутом и историей прогресса, что согласуется с выводами Barrot [3] о необходимости педагогического обрамления генеративного ИИ. Во-вторых, использование жесткого JSON-формата для ответов нейросети гарантирует предсказуемость интерфейса и снижает риск ошибок парсинга на стороне клиента. Нормализация и валидация ответов на сервере полностью исключает появление некорректных данных в пользовательском интерфейсе. В-третьих, механизм генерации follow-up заданий на основе многокритериальной оценки обеспечивает

персонализированную работу над ошибками, фокусируя внимание обучающегося на конкретных слабых местах.

Ограничения исследования связаны с зависимостью качества генерации и оценки от ответов внешней языковой модели DeepSeek, а также с отсутствием в текущей версии административной панели для агрегированного анализа ошибок.

ВЫВОДЫ

В данном исследовании разработана комплексная архитектура веб-приложения для тренировки письменной коммуникации на английском языке, интегрирующая симуляцию реальных цифровых каналов с возможностями больших языковых моделей. Методология объединения жестких JSON-промптов, многокритериальной оценки и генерации follow-up заданий позволила создать прозрачный образовательный инструмент, сочетающий свободную практику, структурированный учебный маршрут и элементы персонализированной обратной связи.

Комплексное тестирование подтвердило корректность работы пользовательских сценариев, согласованность взаимодействия backend, базы данных и внешнего AI-сервиса, устойчивость системы к граничным ситуациям, а также приемлемое время отклика. Дальнейшая работа включает расширение перечня поддерживаемых языков, внедрение административной панели для анализа типичных ошибок пользователей и проведение педагогической апробации платформы.

Список литературы:

1. Kasneci E., Sessler K., Küchemann S., et al. ChatGPT for good? On opportunities and challenges of large language models for education // Learning and Individual Differences. 2023. Vol. 103. P. 102274. DOI: 10.1016/j.lindif.2023.102274
2. Mollick E.R., Mollick L. Instructors as Innovators: A Future-focused Approach to New AI Learning Opportunities // The Wharton School Research Paper. 2023. DOI: 10.2139/ssrn.4802463
3. Barrot J.S. ChatGPT as a Language Learning Tool: An Emerging Technology Report // Technology, Knowledge and Learning. 2024. Vol. 29. No. 2. P. 1151–1156. DOI: 10.1007/s10758-023-09711-4
4. Lo C.K., Yu P.L.H., Xu S., Ng D.T.K., Jong M.S.Y. Exploring the application of ChatGPT in ESL/EFL education and related research issues: A systematic review of empirical studies // Smart Learning Environments. 2024. Vol. 11. Article 27. DOI: 10.1186/s40561-024-00342-5
5. DeepSeek-AI. DeepSeek-V3 Technical Report. 2024. URL: <https://arxiv.org/abs/2412.19437> (дата обращения: 22.07.2026).
6. Mizumoto A., Eguchi M. Exploring the potential of using an AI language model for automated essay scoring // Research Methods in Applied Linguistics. 2023. Vol. 2. No. 2. Article 100050. DOI: 10.1016/j.rmal.2023.100050
7. Xue Y. Towards automated writing evaluation: A comprehensive review with bibliometric, scientometric, and meta-analytic approaches // Education and Information Technologies. 2024. DOI: 10.1007/s10639-024-12596-0
8. DeepSeek API Documentation: Models & Pricing. URL: https://api-docs.deepseek.com/quick_start/pricing/ (дата обращения: 22.07.2026).

References:

1. Kasneci E., Sessler K., Küchemann S., et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 2023, vol. 103, p. 102274. DOI: 10.1016/j.lindif.2023.102274 (accessed 22.07.2026).
2. Mollick E.R., Mollick L. Instructors as Innovators: A Future-focused Approach to New AI Learning Opportunities. *The Wharton School Research Paper*, 2023. DOI: 10.2139/ssrn.4802463 (accessed 22.07.2026).
3. Barrot J.S. ChatGPT as a Language Learning Tool: An Emerging Technology Report. *Technology, Knowledge and Learning*, 2024, vol. 29, no. 2, pp. 1151–1156. DOI: 10.1007/s10758-023-09711-4 (accessed 22.07.2026).
4. Lo C.K., Yu P.L.H., Xu S., Ng D.T.K., Jong M.S.Y. Exploring the application of ChatGPT in ESL/EFL education and related research issues: A systematic review of empirical studies. *Smart Learning Environments*, 2024, vol. 11, art. 27. DOI: 10.1186/s40561-024-00342-5 (accessed 22.07.2026).
5. DeepSeek-AI. DeepSeek-V3 Technical Report. 2024. Available at: <https://arxiv.org/abs/2412.19437> (accessed 22.07.2026).
6. Mizumoto A., Eguchi M. Exploring the potential of using an AI language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2023, vol. 2, no. 2, art. 100050. DOI: 10.1016/j.rmal.2023.100050 (accessed 22.07.2026).
7. Xue Y. Towards automated writing evaluation: A comprehensive review with bibliometric, scientometric, and meta-analytic approaches. *Education and Information Technologies*, 2024. DOI: 10.1007/s10639-024-12596-0 (accessed 22.07.2026).
8. DeepSeek API Documentation: Models & Pricing. Available at: https://api-docs.deepseek.com/quick_start/pricing/ (accessed 22.07.2026).