

Также изучаются подходы к организации уровней с применением *Tilemap* (в *Unity*), что дает возможность эффективно формировать большие сцены и оптимизировать производительность. При создании визуальной части проекта используются спрайтовые ресурсы, позволяющие детализировать окружение и подчеркнуть художественный стиль игры. Важным элементом разработки является создание интерфейса пользователя (*HUD*), отображающего здоровье, способности и элементы взаимодействия.

В процессе проектирования особое внимание уделялось вопросам оптимизации производительности, структурирования кода и возможности масштабирования проекта. Разработка производилась с использованием языка *C#*, что позволяет гибко настраивать поведение игровых объектов, реализовывать логику взаимодействий и сценарные элементы [1–3].

Предполагается, что результатом работы станет прототип интерактивного *2D*-приложения в жанре метроидвания, включающий основные элементы жанра: исследование взаимосвязанных локаций, получение новых умений и постепенное открытие мира. Реализуемые подходы обеспечат возможность дальнейшего расширения функционала, добавления новых механик, персонажей и визуальных элементов.

Создаваемое приложение может использоваться в качестве учебного примера при изучении принципов разработки игр на *Unity*, а также как основа для последующих экспериментальных и коммерческих проектов. Применяемые методы проектирования, архитектурные решения и использование современных инструментов обеспечивают гибкость, расширяемость и потенциал для развития интерактивных *2D*-приложений данного жанра.

Литература

1. Бабори́ко Р. Д. Особенности разработки гиперказуальных игр / Р. Д. Бабори́ко, И. В. Дорошенко // Материалы XXVII Республиканской научной конференции студентов и аспирантов «Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях», Гомель, 18–20 марта 2024 г. – С. 158.
2. Дорошенко, И. В. Создание коллекционной карточной игры с использованием среды разработки *Unity* / И. В. Дорошенко, А. А. Копач // Сборник статей VII Международной научно-практической конференции «Научные исследования 2023», Пенза, 17 июня 2023 г. – С. 20–23.
3. Тикоски, С. Современная разработка игр на *Unity* / С. Тикоски. – СПб. : БХВ-Петербург, 2024. – 496 с.

АРХИТЕКТУРА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ СОЗДАНИЯ ДОСТОВЕРНЫХ УЧЕБНЫХ МАТЕРИАЛОВ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКОВЫХ МОДЕЛЕЙ

Д. А. Шаблинский, О. Д. Асенчик

*Гомельский государственный технический университет
имени П. О. Сухого, Республика Беларусь*

Данная работа посвящена разработке методики и программного комплекса для автоматизации создания достоверных учебных материалов по техническим дисциплинам с использованием больших языковых моделей.

Ключевые слова: большие языковые модели, верификация, технические дисциплины, учебные материалы, автоматизация, достоверность, генерация контента.

ARCHITECTURE OF A SOFTWARE COMPLEX FOR CREATING RELIABLE EDUCATIONAL MATERIALS USING LANGUAGE MODELS

D. A. Shablinski, O. D. Asenchik

Sukhoi State Technical University of Gomel, Republic of Belarus

This work is devoted to the development of a methodology and a software complex for automating the creation of reliable educational materials for technical disciplines using large language models.

Keywords: large language models, verification, technical disciplines, educational materials, automation, reliability, content generation.

Создание качественных лекций, методических материалов – трудоемкий процесс, требующий проработки большого объема информации. Однако с появлением больших языковых моделей (LLM) появляются новые возможности для автоматизации данной деятельности [1]. При прямом применении LLM возникают проблемы, такие как ошибочные и выдуманные данные генерируемых текстов и склонность к галлюцинациям [2]. Проблема недостоверности является критичной в техническом образовании, где точность и надежность учебного контента являются важными аспектами.

Целью работы является описание архитектуры и алгоритма работы программного комплекса, позволяющего генерировать учебные материалы на основе наборов экзаменационных вопросов с высокой достоверностью. Разработанная система поддерживает работу как с OpenAI API языковых моделей, так и с локальными языковыми моделями.

Основными этапами реализации предлагаемой методики являются: разработка наборов вопросов, конкретизирующих и детализирующих изучение тем учебной дисциплины; выбор набора реализаций LLM и предварительная оценка их возможностей ответить на эти вопросы без галлюцинаций; многоступенчатая генерация ответов на каждый из полученных вопросов с предварительной верификацией; и оценочная финальная верификация текстов полученных ответов.

Подход, предполагающий генерацию ответов на все вопросы одновременно в рамках одного запроса к LLM, оказался неэффективным из-за недостаточного размера контекстного окна большинства моделей для передачи больших объемов информации, большой нагрузки на серверы и неустойчивости сетевого соединения. В результате, нами применялась методика ответа на каждый вопрос по отдельности.

Система построена по модульному принципу и включает три основных блока, которые работают последовательно, что гарантирует надежную достоверность контента.

Требования к выбору LLM. При выборе LLM, используемых в комплексе, учитывались следующие критерии: стоимость использования, доступность через API, лицензия, объем контекстного окна, а также аналитические возможности и требования к локальным ресурсам. В качестве используемых для проведения экспериментов LLM были выбраны DeepSeek V3, DeepSeek R1, Llama 4, Gemini 2.5 Flash, Qwen3 и Gemma 3 27B. Каждая из этих моделей представляет значительное достижение в области искусственного интеллекта и доступна через API различных провайдеров [3].

Первый модуль занимается подготовкой запроса. На основе входных вопросов он собирает детализированный запрос, в котором помимо самого вопроса заданы директивы по стилю и структуре ответа, а также ограничения, характерные для конкретной дисциплины. Такая детализация важна для снижения неопределенности и повышения релевантности ответа LLM. Сформированный запрос передается через

API во второй модуль – выбранную LLM, будь то облачная модель или локальная, что обеспечивает гибкость и позволяет контролировать затраты. Полученные от LLM ответ сохраняются в единый формат, благодаря чему их последующая обработка становится проще, а включение в учебный процесс – более удобным.

В центре всей архитектуры стоит финальный модуль, чья задача – верифицировать сгенерированный текст. Он автоматически сверяет полученный материал с проверенными источниками – утвержденными учебными пособиями, формирующими эталонную базу знаний. Механизм верификации основан на алгоритмах семантического сравнения фрагментов текста. Для этой цели в качестве эмбединговой модели выбрана MiniLM-L6-v2. Данная модель преобразует как эталонные тексты из базы знаний, так и сгенерированные ответы LLM в высоко размерные числовые векторы. Далее система использует векторную базу данных FAISS для эффективного поиска и количественной оценки семантической близости сгенерированного контента к источникам. Это позволяет выявлять и маркировать так называемые галлюцинации – сведения, не найденные в эталонных ресурсах. Таким образом, система не ограничивается лишь генерацией, а также активно отбирает контент, поддерживая высокий стандарт образовательных материалов.

Для проверки эффективности разработанного решения было проведено экспериментальное исследование. В качестве тестовых данных использовались экзаменационные вопросы по курсу «Основы Web-дизайна и HTML». Эталонной базой знаний являются утвержденные учебные пособия. Сгенерированные ответы оценивались по десятибалльной шкале согласно трем критериям: точность, полнота раскрытия темы и структурная целостность. Полученный анализ показал, что большинство ответов получили высокую оценку (8 баллов и выше), что подтверждает высокую эффективность модуля верификации. При этом ответы обладают четкой структурой, сопровождаются теоретическим обоснованием и практическими примерами.

Разработанная архитектура способствует созданию качественных лекционных и методических материалов, предлагая системный подход к автоматизации учебного процесса. Использование LLM позволяет экономить время и поддерживать высокий стандарт образовательных материалов. Модульная организация позволяет адаптировать систему для разных дисциплин и сохранять контроль качества. Возможность использования различных языковых моделей обеспечивает гибкость выбора оптимального решения для конкретных учебных задач и предметных областей. Интеграция с открытыми API позволяет подключать новые модели по мере их появления, что обеспечивает постоянное развитие системы. Сравнительный анализ результатов позволяет подобрать наиболее точные формулировки, необходимые для технических дисциплин.

Литература

1. ChatGPT for good? On opportunities and challenges of large language models for education / E. Kasneci, K. Seßler, S. Küchemann [et al.] // Learning and Individual Differences. – 2023. – Vol. 103. – P. 95. – DOI 10.1016/j.lindif.2023.102274
2. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? / E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell // Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. – 2021. – P. 610–623.
3. OpenRouter. – URL: <https://openrouter.ai/> (дата обращения: 05.10.2025).