

УДК 004.85:378.147

ГЕНЕРАЦИЯ И ВЕРИФИКАЦИЯ УЧЕБНОГО КОНТЕНТА ПО ДИСЦИПЛИНЕ «ПРОГРАММИРОВАНИЕ СЕТЕВЫХ ПРИЛОЖЕНИЙ» С ИСПОЛЬЗОВАНИЕМ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ

Шаблинский Д. А.

магистрант, Гомельский государственный технический
университет имени П. О. Сухого

Введение. Создание качественных лекционных курсов и методических материалов представляет собой трудоемкий процесс, требующий глубокой проработки значительного объема информации. Однако с появлением и развитием больших языковых моделей (LLM) открываются принципиально новые возможности для автоматизации данной деятельности [1, 3]. При прямом применении LLM неизбежно возникают проблемы, такие как генерация ошибочных или выдуманных данных, а также склонность моделей к так называемым «галлюцинациям» [2]. Проблема достоверности является критичной именно в сфере технического образования, где точность, надежность и фактическая корректность учебного контента являются определяющими аспектами.

Целью работы является разработка метода автоматизированной генерации учебного контента и алгоритма его верификации на основе архитектуры генерации, дополненной поиском (Retrieval-Augmented Generation, RAG). Данный подход направлен на минимизацию фактологических искажений в материалах по ИТ-дисциплинам путем семантического сопоставления генерируемых данных с эталонными источниками. Для реализации этой задачи используется векторная база данных Qdrant, обеспечивающая строгое обоснование сгенерированного контента через поиск релевантных контекстов в утвержденных учебных пособиях.

Методология и алгоритм работы. Предлагаемая методика базируется на многоступенчатом процессе, направленном на обеспечение верифицируемости контента. Реализация включает четыре ключевых этапа. На первом этапе производится декомпозиция учебной дисциплины на систему структурированных вопросов. Далее выполняется экспертный отбор LLM, исходя из их способности к генерации ответов без смысловых искажений. Основной этап – ге-

нерация – реализуется итерационно: по каждому вопросу в отдельности. В ходе исследования была обоснована нецелесообразность пакетной генерации (batch generation) всего корпуса материалов. Переход к атомарному подходу обусловлен необходимостью минимизации феномена «потери информации в середине контекста» (lost-in-the-middle) и обеспечения высокой гранулярности данных для семантического сопоставления векторов в рамках метода RAG.

Программная реализация и структура комплекса. Разработанная система построена по модульному принципу на основе RAG-архитектуры и включает четыре основных функциональных блока, работающих последовательно, что обеспечивает достоверность итогового контента. При выборе LLM, используемых в программном комплексе, учитывался ряд строгих критериев, таких как стоимость использования, доступность через API, тип лицензии, объем контекстного окна, а также аналитические возможности и требования к локальным ресурсам. Для проведения экспериментов был отобран широкий спектр моделей, включая DeepSeek V3.2, ChatGPT 5, Llama 4, Gemini 3 Flash, Qwen3 и Gemma 3 27B. Каждая из этих моделей является значительным достижением в современной области искусственного интеллекта и доступна для интеграции через API различных провайдеров [4].

Первый модуль системы формулирует запрос. На основе входных данных он собирает детализированный промпт, в котором помимо самого вопроса заданы строгие директивы относительно стиля и структуры ожидаемого ответа, а также ограничения, характерные для конкретной технической дисциплины. Такая детализация критически важна для снижения уровня неопределенности и существенного повышения релевантности ответа языковой модели. Сформированный запрос передается через API во второй модуль – выбранную LLM, будь то облачная модель или локальное решение, что обеспечивает гибкость архитектуры и позволяет контролировать финансовые затраты. Полученные от модели ответы сохраняются в единый унифицированный формат, благодаря чему их последующая обработка становится значительно проще, а включение в учебный процесс – более удобным.

Верификация контента. Ключевым этапом работы является процедура автоматической верификации, за которую отвечает финальный модуль системы. Он сверяет полученный материал с эта-

лонной базой знаний, сформированной из утвержденных учебных пособий. Механизм верификации базируется на алгоритмах семантического сравнения: с помощью модели MiniLM-L6-v2 текстовые фрагменты преобразуются в многомерные числовые векторы (эмбединги). Выбор данной модели обусловлен её эффективностью при работе с русскоязычными техническими текстами и оптимальным балансом между точностью анализа и скоростью обработки данных. Далее система использует векторную базу данных Qdrant для поиска наиболее релевантных совпадений и количественной оценки степени их семантической близости. Достоверность утверждений оценивается на основе коэффициента косинусного сходства векторов: если близость сгенерированного фрагмента к эталонному тексту ниже установленного порогового значения, информация идентифицируется как «галлюцинация». Это позволяет системе не только генерировать текст, но и активно фильтровать его, гарантируя соответствие образовательного контента проверенным источникам.

Результаты эксперимента. Для проверки эффективности решения было проведено исследование на базе курса «Программирование сетевых приложений». Эталонной базой знаний выступали официально утвержденные учебные пособия. Сгенерированные ответы оценивались экспертами по десятибалльной шкале согласно критериям точности, полноты и структурной целостности. Анализ показал, что применение модуля верификации позволило выявить и устранить до 18% фактических неточностей в описании сетевых протоколов и параметров библиотек. Большинство ответов получили оценку не ниже 8 баллов. Тексты отличаются строгой структурой, наличием теоретических выкладок и практическими примерами кода, полностью соответствующими стандартам дисциплины.

Заключение. Разработанный программный комплекс обеспечивает возможность генерации учебных материалов как в стандартном режиме LLM, так и с применением усиленного протокола RAG. В обоих случаях высокая достоверность контента достигается благодаря процедуре финальной верификации через векторную базу Qdrant. Использование LLM в таком формате существенно экономит время преподавателя на подготовку материалов и гарантирует их качество. Модульная организация позволяет адаптировать систему для разных дисциплин и сохранять строгий контроль качества. Интеграция с открытыми API позволяет подключать новые модели

по мере их развития, что обеспечивает возможность постоянной эволюции системы и подбора наиболее точных формулировок для сложных технических текстов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Асенчик, О. Д. Использование больших языковых моделей для создания учебных материалов по дисциплине «Базы данных» / О. Д. Асенчик // Цифровая трансформация. – 2025. – Т. 31, № 4. – С. 5–14.
2. Bender, E. M. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? / E. M. Bender, T. Gebru, A. McMillan-Major // Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. – 2021. – P. 610–623.
3. Kasneci, E. ChatGPT for good? On opportunities and challenges of large language models for education / E. Kasneci, K. Seßler, S. Küchemann // Learning and Individual Differences. – 2023. – Vol. 103. – P. 102274.
4. OpenRouter [Electronic resource]. – 2024. – Mode of access: <https://openrouter.ai/>. – Date of access: 05.12.2025.

УДК 543.544:664

МЕТОДЫ ОПРЕДЕЛЕНИЯ СОДЕРЖАНИЯ ФУРАНА В ПИЩЕВЫХ ПРОДУКТАХ

Шимчук А. А.

аспирант

Введение. Фуран – это бесцветное гетероциклическое соединение с высокой летучестью (температура кипения: 31°C), состоящее из пятичленного ароматического кольца с четырьмя атомами углерода и одним атомом кислорода. Структурная формула фурана представлена на рисунке 1.

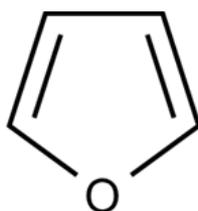


Рисунок 1 – Структурная формула фурана

Известно, что термическая обработка является основной причиной образования фурана. Существует множество механизмов, лежащих в основе его образования, к ним относятся термическая