

**ЕСТЕСТВЕННОЯЗЫКОВЫЙ ИНТЕРФЕЙС ДЛЯ ГЕНЕРАЦИИ  
ЗАПРОСОВ К БАЗЕ ДАННЫХ ПО ДОСТУПНЫМ  
ТЕХНИЧЕСКИМ МЕТОДАМ ИСПОЛЬЗОВАНИЯ  
ПРОИЗВОДСТВЕННЫХ ОТХОДОВ**

***Аннотация.** В работе рассматривается система, обеспечивающая естественно-языковой интерфейс для взаимодействия с базой данных. Система использует большие языковые модели для обработки запросов пользователей. Показано, что использование внешних утилит при работе с большими языковыми моделями точность ответов.*

**A.V. Tachyla, V.V. Smelov**

Belarusian State Technological University  
Minsk, Belarus

**NATURAL LANGUAGE INTERFACE FOR GENERATING  
QUERIES TO A DATABASE ON AVAILABLE TECHNICAL  
METHODS FOR USING INDUSTRIAL WASTE**

***Abstract.** The paper considers a system that provides a natural language interface for interacting with a database. The system uses large language models to process user requests. It is shown that the use of external utilities when working with large language models is the accuracy of responses.*

Современные промышленные предприятия генерируют значительные объемы отходов, что требует эффективных методов их утилизации и переработки. Базы данных, содержащие информацию о технических методах использования производственных отходов, часто бывают сложными для неспециалистов: запросы требуют знания SQL или специфических интерфейсов. Разработка естественно-языкового интерфейса позволяет пользователям формулировать запросы на естественном языке, делая систему доступной для менеджеров, инженеров и аналитиков. В рамках магистерской диссертации предлагается система, интегрирующая большие языковые модели (LLM – Large Language Models – такие как OpenAI GPT, Anthropic Claude и другие) с инструментами для доступа к базам данных и внешним серверам. Это решает задачи как простого извлечения данных (например, "Какие товары производит производитель А?"), так и сложных расчетов (например, "Возможно ли сократить выпуск продукта Б за счет повышения выпуска других продуктов?"), где LLM

делегировать вычисления внешнему RPC-серверу для решения системы линейных уравнений.

Проблема галлюцинаций в LLM, когда модели генерируют неверные ответы из-за статистической природы обучения, особенно актуальна в технических областях, где ошибки могут привести к неверным решениям с серьезными последствиями. Для минимизации этого риска система использует внешние утилиты: LLM не генерирует данные самостоятельно, а вызывает специализированные инструменты. В частности, применяется MCP (Model Context Protocol) для обогащения контекста модели релевантными данными, находящимися в базе данных, где хранится информация о кластере производителей, их продуктах, предпосылках (включая отходы как входные ресурсы) и планах производства. База моделирует сценарии, где продукты одного производителя могут стать сырьем для другого, выстраиваясь в производственные цепочки с планами по выпуску продукции за пределы кластера.

Аналитический обзор технологий показал, что фреймворки LangChain и LangGraph подходят для создания и управления последовательностями инструментов в LLM, позволяя строить графы потока операций для обработки запросов. LangChain обеспечивает модульность в интеграции LLM с внешними API, а LangGraph – управление состоянием в многошаговых взаимодействиях. MCP выбран для работы с базами данных, так как предоставляет стандартизированный интерфейс для инструментов, включая получение схем базы данных, описание объектов базы данных, таких как таблицы со столбцами, ограничениями целостности и индексами и выполнение произвольных запросов к базе данных.

Разработанный прототип приложения использует Python с библиотеками `asyncio` для обеспечения асинхронности обработки пользовательских запросов, `postgres-mcp` для доступа к базе данных и `OpenAI` для выполнения запросов к большим языковым моделям. Система инициализирует соединение MCP-сервера с базой данных и устанавливает соединение с сервером. Пользовательский запрос передается LLM с системным сообщением, содержащим математическую модель, которую должна использовать языковая модель: определение сущностей и связей между ними, инструкции для модели, набор аксиом и перечисление внешних утилит, доступных модели.

Процесс обработки запроса цикличен: LLM анализирует запрос, вызывает инструменты (если нужно), получает результаты и генерирует финальный ответ. Например, для запроса о продуктах

производителя LLM сначала получает схему и описание объектов в ней, при необходимости выполняя дополнительные запросы для определения объектов, необходимых для обработки запроса (например, ищет нужную таблицу из имеющихся в схеме, если запрос содержит не точное, а приблизительное название сущности), формирует SQL-запрос и выполняет его через MCP. Для оптимизационных задач, таких как изменение плана производства, LLM вызывает RPC-сервер, передавая необходимые параметры, где сервер решает систему линейных уравнений (баланс исходных и исходящих продуктов). Это обеспечивает повышенную точность ответа, т.к. LLM делегирует расчеты специализированному модулю.

Прототип тестировался на начальной базе данных с данными о кластере: производители А, В, С; продукты с зависимостями (отход продукта А – сырье для В); планы производства. Пример запроса: "Какие товары производит производитель А?" — LLM генерирует SQL "SELECT \* FROM products WHERE producer\_id = (SELECT id FROM producers WHERE name = 'A')", выполняет через MCP и возвращает список. Для "Возможно ли сократить продукт В на 20% за счет других?" – LLM вызывает HTTP-сервер с JSON {"product": "В", "reduction": 20, "optimize": true}, получает решение уравнений и интерпретирует результат.

Эксперименты показали снижение ошибок на 50-70% по сравнению с чистыми LLM благодаря инструментам. Система справляется с синонимами и неоднозначностями, запрашивая уточнения. Гибридный подход сочетает генеративные возможности LLM с формализованным доступом к данным, делая систему надежной для задач утилизации отходов. Это открывает перспективы для приложений с повышенными требованиями к точности и достоверности генерируемых ответов.

### **Список использованных источников**

1. Бхуян Ш., Исаченко Т. – Генеративный ИИ с обучением больших языковых моделей (LLM) для джунов. – Москва: Литрес, 2025.
2. Дхамани Н., Энглер М. – Генеративный искусственный интеллект. Как ИИ меняет нашу жизнь и работу. – Москва: Литрес, 2025.
3. Берриман Д., Циглер. А. – Промт-инжиниринг для LLM. Искусство построения приложений на основе больших языковых моделей. – Санкт-Петербург: Питер, 2025.