

Генерация SQL-запросов по текстовому описанию с использованием нейросети GPT-5.1

Саяпин Никита Анатольевич

Приамурский государственный университет им.Шолом-Алейхема

Студент

Аннотация

Целью данного исследования является демонстрация процесса автоматической генерации SQL-запросов к реляционным базам данных на основе текстового описания информационных потребностей пользователя с использованием мультимодальной нейросети GPT-5.1. В работе рассматриваются принципы интерпретации моделью описания схемы базы данных и естественно-языковых запросов, формирование структуры SQL-выражений (SELECT, JOIN, агрегирование, фильтрация) и анализ соответствия сгенерированных запросов исходным требованиям.

Ключевые слова: искусственный интеллект, нейросеть, GPT-5.1, базы данных, SQL, Text-to-SQL, обработка естественного языка, автоматизация.

Generation of SQL queries from textual description using the GPT-5.1 neural network

Sayapin Nikita Anatolyevich

Sholom-Aleichem Priamursky State University

Student

Abstract

The purpose of this study is to demonstrate the process of automatic generation of SQL queries for relational databases from textual user requirements using the GPT-5.1 multimodal neural network. The paper discusses how the model interprets database schema descriptions and natural language questions, constructs SQL statements (SELECT, JOIN, aggregation, filtering), and how well the generated queries match the initial requirements.

Keywords: artificial intelligence, neural network, GPT-5.1, databases, SQL, Text-to-SQL, natural language processing, automation.

1. Введение

1.1 Актуальность

Реляционные базы данных остаются одним из ключевых способов хранения и обработки данных в информационных системах. Язык SQL является стандартным инструментом доступа к данным, однако его использование требует от пользователя знания структуры базы данных,

синтаксиса запросов и особенностей конкретной СУБД. Для специалистов без опыта работы с SQL барьер входа остаётся достаточно высоким.

Одним из подходов к снижению этого барьера является создание естественно-языковых интерфейсов к базам данных, позволяющих формулировать запросы в свободной форме, а затем автоматически преобразовывать их в корректные SQL-выражения. Эта задача в литературе часто обозначается как Text-to-SQL или NL2SQL (Natural Language to SQL).

Появление больших языковых моделей нового поколения, таких как GPT-5.1, создаёт возможность реализовать универсальный механизм преобразования текстового описания в SQL-код без жёсткой привязки к конкретной предметной области. Мультимодальные модели способны одновременно учитывать описание схемы базы данных (таблицы, поля, связи) и естественно-языковой запрос пользователя, формируя итоговый SQL-запрос с учётом контекста.

1.2 Обзор исследований

К. И. Львович рассматривает общие принципы построения систем естественно-языкового интерфейса к информационным системам, выделяя ключевые компоненты таких решений и их роль в обеспечении взаимодействия пользователя с базами данных и знания на естественном языке [1]. Н. В. Соболева анализирует подходы к проектированию когнитивного интерфейса к базам данных, рассматривая естественный язык как основной способ описания предметной области и взаимодействия пользователя с системой. В работе уделяется внимание вопросам отображения фрагментов естественно-языкового описания в элементы концептуальной модели данных, что является важной предпосылкой для автоматического формирования запросов к БД [2]. О. Е. Елисеева предлагает семантический подход к концептуальному проектированию естественно-языкового интерфейса интеллектуальной системы, в котором значительное внимание уделяется моделированию понятий и отношений предметной области [3]. И.С.Постаногов исследует построение сервиса интеллектуализации доступа к данным на базе онтологий, в том числе с использованием естественно-языкового интерфейса. В работе показывается, что онтологическое представление знаний в сочетании с механизмами обработки запросов на естественном языке позволяет реализовать более гибкий и контекстно-зависимый доступ к данным, а также снижает требования к технической подготовке пользователя [4]. В. Н. Кучуганов рассматривает одну из возможных технологий системного анализа, структурирования и частичной формализации описаний сущностей и проблемных ситуаций как попытка уменьшить влияние субъективизма при проектировании систем поддержки принятия решений. Предлагается разграничить понятия семантических и ассоциативных сетей, описываются операции с образами ситуаций для поиска аналогий. Предложена концептуальная схема ассоциативного узнавания текста [5].

1.3 Цель исследования

Цель исследования — продемонстрировать процесс автоматической генерации SQL-запросов по текстовому описанию с использованием нейросети GPT-5.1 и оценить точность интерпретации моделью естественно-языковых требований в трёх типовых сценариях.

2 Материалы и методы

В рамках исследования использовалась мультимодальная нейросеть GPT-5.1, способная преобразовывать текстовые описания требований пользователя и структуры базы данных в формальные SQL-запросы. Работа модели рассматривалась как единый процесс преобразования естественно-языкового текста в структурированный запрос, совместимый с синтаксисом реляционных СУБД.

Проверка работы включала формирование текстового описания схемы базы данных, составление пользовательских запросов на естественном языке, генерацию SQL-выражений при помощи GPT-5.1 и анализ соответствия полученного результата исходным требованиям.

3. Результаты

3.1 Генерация простого запроса выборки по одной таблице

На первом этапе исследовалась способность модели формировать базовые SQL-запросы на основе краткого текстового описания информационной потребности. Задача включала выборку данных из одной таблицы с применением условий фильтрации и сортировки. GPT-5.1 продемонстрировала уверенную интерпретацию условий, корректно выделила необходимые параметры и преобразовала текст в структурированный SQL-запрос.

Промпт:

«Сгенерируй SQL-запрос для PostgreSQL.

Структура таблицы customers:

- id (integer, PK)
- full_name (text)
- city (text)
- created_at (timestamp)

Задача:

Получить id, full_name и created_at всех клиентов из города "Москва", которые зарегистрировались после 1 января 2024 года.

Отсортировать по дате регистрации по убыванию.»

Полученный SQL-код корректно отражал все требования: использовались только указанные поля, применялись условия фильтрации по городу и дате, добавлялась сортировка по убыванию. Модель автоматически учла формат даты и корректное экранирование строковых значений, что

повышает применимость результата. Итоговый запрос был лаконичным и полностью соответствовал текстовому описанию (Рис. 1).

```
sql

SELECT
    id,
    full_name,
    created_at
FROM customers
WHERE city = 'Москва'
    AND created_at > '2024-01-01'
ORDER BY created_at DESC;
```

Рисунок 1 — Генерация простого SQL-запроса по текстовому описанию

3.2 Формирование запроса с объединением таблиц

На следующем этапе анализировалась способность модели формировать SQL-запросы, включающие соединение таблиц по внешнему ключу. Текстовое описание содержало структуру двух взаимосвязанных таблиц и задачу, требующую выборки данных одновременно из обеих. Модель корректно идентифицировала связь между таблицами, применила объединение и сформировала перечень требуемых полей.

Промпт:

«Сгенерируй SQL-запрос для PostgreSQL.

Структура таблицы customers:

- id (integer, PK)
- full_name (text)
- email (text)

Структура таблицы orders:

- id (integer, PK)
- customer_id (integer, FK -> customers.id)
- total_amount (numeric)
- created_at (timestamp)

Задача:

Получить список клиентов и их заказов.

В выборку должны входить: full_name, email, id заказа, total_amount, created_at.

Показать только заказы, созданные в марте 2025 года.

Отсортировать результаты по дате заказа по возрастанию.»

Сформированный SQL-запрос включал корректное использование INNER JOIN, диапазон дат был рассчитан верно, поля таблиц указаны в точном соответствии с описанием задачи. Дополнительно GPT-5.1 использовала алиасы таблиц, что улучшило читаемость результата (Рис. 2).

```
sql

SELECT
    c.full_name,
    c.email,
    o.id AS order_id,
    o.total_amount,
    o.created_at
FROM orders o
JOIN customers c ON o.customer_id = c.id
WHERE o.created_at >= '2025-03-01'
      AND o.created_at < '2025-04-01'
ORDER BY o.created_at ASC;
```

Рисунок 2 — Генерация SQL-запроса с объединением таблиц

3.3 Построение агрегирующего запроса с группировкой

На заключительном этапе проверялась способность модели выполнять обработку текстовых описаний агрегирующих задач: расчёт суммарных метрик, группировка по периодам и фильтрация результатов по агрегатным функциям.

GPT-5.1 корректно интерпретировала понятия «месячная выручка», «сумма оплаченных заказов», «количество заказов», преобразовав их в SQL-конструкции SUM, COUNT, GROUP BY и HAVING.

Промпт:

«Сгенерируй SQL-запрос для PostgreSQL.

Структура таблицы orders:

- id (integer, PK)
- customer_id (integer)
- total_amount (numeric)
- status (text)
- created_at (timestamp)

Задача:

Рассчитать месячную выручку за 2025 год.

Для каждого месяца нужно вывести:

- месяц в формате YYYY-MM,
- сумму total_amount для заказов со статусом 'paid',
- количество таких заказов.

Показать только те месяцы, где сумма превышает 100000.

Отсортировать по месяцу по возрастанию.»

Сформированный SQL-запрос содержал корректное извлечение месяца из даты, использование агрегирующих функций и условие HAVING для фильтрации итогов. Модель чётко соблюдала требования к формату данных и структуры группировки (Рис. 3).

```
SELECT
    to_char(created_at, 'YYYY-MM') AS month,
    SUM(total_amount)              AS total_revenue,
    COUNT(*)                      AS paid_orders_count
FROM orders
WHERE status = 'paid'
    AND created_at >= '2025-01-01'
    AND created_at < '2026-01-01'
GROUP BY to_char(created_at, 'YYYY-MM')
HAVING SUM(total_amount) > 100000
ORDER BY month ASC;
```

Рисунок 3 – Генерация агрегирующего SQL-запроса по описанию метрик

3.4 Сравнение результатов

Анализ трёх исследуемых задач показал, что модель уверенно интерпретирует естественно-языковое описание структуры базы данных и условий выборки, корректно строит SQL-запросы различной сложности и использует типовые практики написания запросов. Основные различия в качестве результата определялись полнотой исходного текстового описания: чем детальнее формулировка, тем точнее итоговый SQL-код.

4 Выводы

Проведённое исследование показало, что использование мультимодальной нейросети GPT-5.1 для автоматической генерации SQL-запросов по текстовому описанию позволяет формировать корректные и структурно понятные выражения для типовых задач работы с реляционными базами данных. Модель продемонстрировала способность учитывать схему базы данных, корректно интерпретировать естественно-языковые

формулировки и строить запросы с фильтрацией, объединением таблиц и агрегированием результатов.

Основным ограничением является зависимость качества результата от полноты и точности исходного текстового описания: при неполной информации о схеме базы данных или неточных формулировках требований возможны логические ошибки, требующие вмешательства разработчика или аналитика. Кроме того, модель не учитывает особенности конкретной СУБД (индексы, ограничения, статистику по данным), а потому не может гарантировать оптимальность сгенерированных запросов.

Тем не менее, GPT-5.1 может рассматриваться как эффективный инструмент поддержки работы с базами данных, позволяющий снизить порог входа для пользователей без глубоких знаний SQL, ускорить подготовку запросов и использовать естественный язык как удобный интерфейс доступа к данным. Перспективным направлением дальнейших исследований является интеграция генерации SQL-запросов с системами визуальной аналитики и средствами автоматической валидации и оптимизации запросов.

Библиографический список

1. Львович К. И., Плотников А. А. О системах естественно-языкового интерфейса // Информационные технологии в управлении, автоматизации и мехатронике. отв. ред. М. С. Разумов. 2022. С. 137–140.
2. Соболева Н. В. Подходы к проектированию когнитивного интерфейса к базам данных // Информатика, вычислительная техника и инженерное образование. 2012. № 1(8). С. 17–26.
3. Елисеева О. Е. Семантическое концептуальное проектирование естественно-языкового интерфейса интеллектуальной системы // Открытые семантические технологии проектирования интеллектуальных систем. 2014. № 4. С. 321–332.
4. Постановов И. С. Построение сервиса интеллектуализации информационных систем как открытой системы // Вестник молодых ученых ПГНИУ. Пермь. 2015. С. 135–143.
5. Кучуганов В. Н. Вербализация реальности и виртуальности. Ассоциативная семантика // Искусственный интеллект и принятие решений. 2011. № 1. С. 55-66.