

# ML в FastBoard

- [Модуль проверки качества данных](#)

# Модуль проверки качества данных

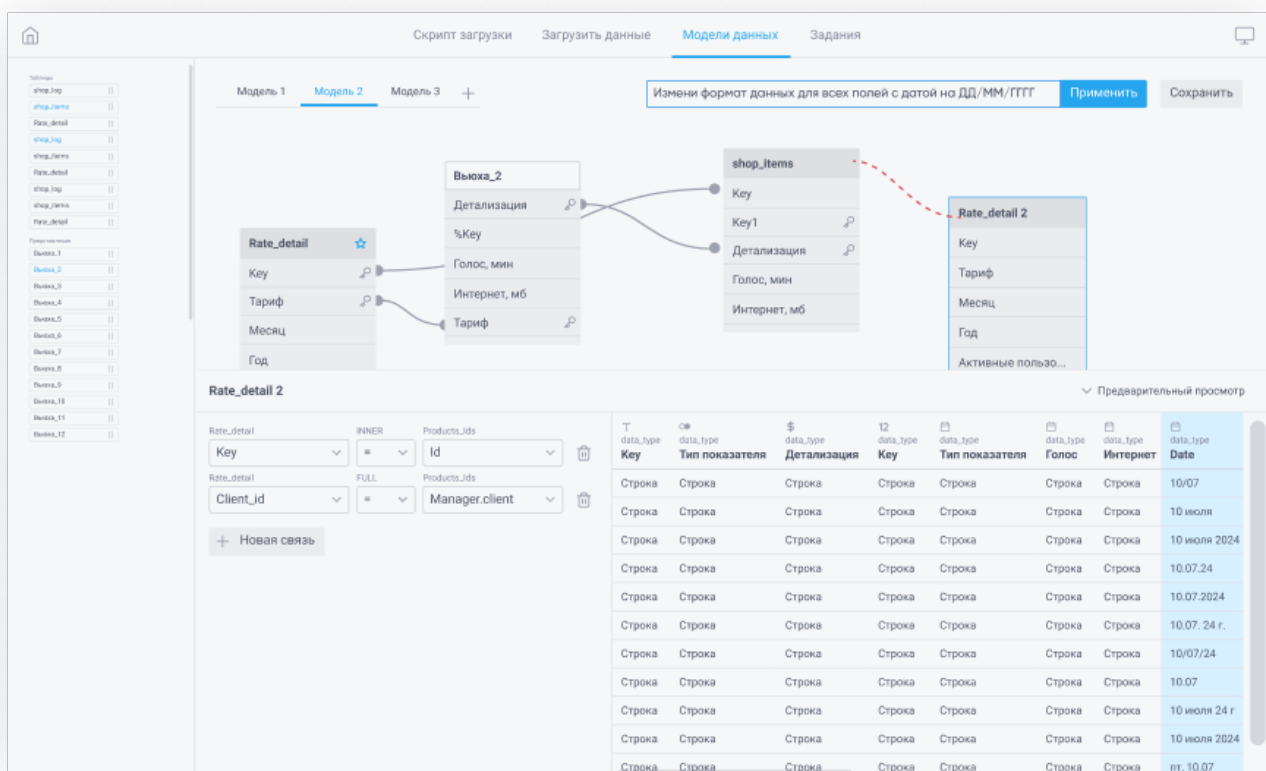
## Задача:

Создание и внедрение в редактор модели данных модуля проверки качества данных с использованием языковой модели нейронной сети для обработки текстовых сообщений от пользователя.

## Концепт:

**Фронт:** текстовая строка для ввода сообщения от пользователя с кнопками «Применить», «Сохранить» и «Отменить». При нажатии на кнопку «Применить» запускается обработка текстового сообщения пользователя языковой моделью. После обработки сообщения и внесения моделью изменений в имеющуюся таблицу в секции «Предварительный просмотр» отображается измененная таблица. При нажатии на кнопку «Сохранить» внесенные изменения вносятся в скрипт и меняют модель данных. При нажатии на кнопку «Отменить» (при условии, что не была нажата кнопка «Сохранить») все предложенные моделью изменения сбрасываются, таблица для предварительного просмотра возвращается в первоначальное состояние.

**ML:** языковая модель для обработки текстовых сообщений от пользователя. Языковая модель обрабатывает 2 вида запросов (и приводит к ним иные): задачи редактирования и задачи поиска. Превращает запрос от пользователя в SQL-код, который используется для создания демонстрационной таблицы или окна с ответом. Демонстрационная таблица хранится временно до тех пор, пока пользователь не подтвердит её использование, не отменит выбор или не выйдет из модели данных. Модель работает только с данными существующих таблиц; не может создавать новые таблицы или удалять имеющиеся.



## Функциональные требования:

**Фронт:** интеграция в модуль работы с моделями данных текстовой строки с 3 управляющими кнопками:

1. Текстовое поле имеет неограниченную длину по числу символов. При достижении видимой границы текстового поля справа оно должно автоматически расширяться вниз с переносом текста на следующую строку. Расширенное поле должно быть видно только при выборе строки запроса нажатием на него. Нажатие на любое пустое место на странице должно приводить к автоматическому скрыванию поля до размеров первоначальной строки.
2. Кнопка «Применить» отправляет POST-запрос через API на бэкенд. В теле запроса находится текст из текстового поля и SQL-код из БД. После нажатия на кнопку «Применить» другие кнопки остаются недоступными до окончания работы модели и формирования таблицы предварительного просмотра. Из бэкенда возвращается SQL-запрос, который формирует временную таблицу для просмотра.
3. Кнопка «Сохранить». По умолчанию недоступна (некликабельна). Активируется после формирования таблицы предварительного просмотра. При нажатии на кнопку SQL-код интегрируется в редактор скрипта и происходит перезапись модели.
4. Кнопка «Отменить» - удаляет временную таблицу, возвращает в таблицу предварительного просмотра результат работы скрипта без изменений от языковой модели.

**ML:** языковая модель принимает текстовый запрос и исходный SQL-код. Возвращает измененный SQL-код. Решает задачи редактирования строк столбцов, столбцов таблицы, а также задачи поиска.

## 1. Обработка строк:

1.1. Ограничение выборки – уменьшение числа строк в столбце по одному из следующих принципов:

1.1.1. Прямое удаление. Пример: "Удали все строки в столбце Расходы, содержащие текстовые значения". ALTER TABLE Таблица DELETE WHERE Столбец Условие

1.1.2. Удаление дубликатов. Пример: "Оставь только неповторяющиеся строки в столбце Доходы". SELECT DISTINCT ... FROM Таблица

1.1.3. Количественные ограничения на число значений с начала / конца . Пример: "Оставь только первые 100 строк в таблице". SELECT ... FROM Таблица LIMIT / OFFSET Значение

1.1.4. Задание условий - ограничение выборки одним из следующих способов:

1.1.4.1. **Условные функции**. Пример: "Если в столбце нет числовых значений, то удали его. Иначе умножь все числовые значения на 100". SELECT IF (Условие, Результат, Иначе)

1.1.4.2. Принадлежность – проверка на наличие значения в массиве / множестве / диапазоне. Использует операторы **IN, EXIST, BETWEEN**. Пример: "Оставь только даты в диапазоне от января 2023 до апреля 2024".

1.1.4.3. **Сравнение**. Использует операторы для сравнения как с фиксированными значениями, так и для сравнения столбцов между собой. Пример: "Оставь только положительные значения в столбце Прибыль"; "Оставь только строки, в которых Доход больше Расхода"

1.1.4.4. **Условия по шаблонам** – поиск по значениям строк в соответствии с заданным шаблоном. Пример: "Оставь только те текстовые строки, в которых содержатся кавычки"

1.2. **Сортировка по столбцу**. Пример: "Отсортируй столбец Количество по убыванию, пустые строки в конце". ORDER BY Столбец Правило

1.3. Добавление / изменение строк – расширение или заполнение таблицы одним из следующих способов:

1.3.1. Генерация строк по условию – добавление новых строк с использованием генератора чисел. Неуказанные столбцы должны заполняться пустыми значениями. Пример: "Добавь в таблицу 50 строк, где Название компании – ООО "Домик", Доходы варьируются от 20000 до 30000 с равномерными шагами между значениями". Должен формироваться массив строк, которые после будут добавлены с помощью INSERT INTO.

1.3.2. Чёткое заполнение строк – пользователь сам вводит все нужные ему значения:

1.3.2.1. Создание строк для всей таблицы. Пример: "Добавь две строки: Вася, апельсины, 5000, Краснодар. Миша, бананы, 10000, Сочи". INSERT INTO TABLE Таблица VALUES (Значение столбца 1, Значение столбца 2, ..., Значение столбца n)

1.3.2.2. Создание строк в одном столбце и назначение их имеющимся значениям. Аналогично предыдущему пункту, но заполняется только одно поле для заполненной строки.

Пример запроса: "Добавь название города Краснодар ко всем строкам, где цена больше 5000". Пример до выполнения: "Москва, 3000, яблоки", "NULL, 6000, бананы", "Ростов, 7000, апельсины". Пример после выполнения: "Москва, 3000, яблоки", "Краснодар, 6000, бананы", "Ростов, 7000, апельсины".

Альтернативный запрос: "Замени на название города Краснодар во всех строках, где цена больше 5000". Пример до выполнения: "Москва, 3000, яблоки", "NULL, 6000, бананы", "Ростов, 7000, апельсины". Пример после выполнения: "Москва, 3000, яблоки", "Краснодар, 6000, бананы", "Краснодар, 7000, апельсины".

