Прогнозирование конверсий на сайте

Модель машинного обучения для увеличения эффективности сайта

Команда № 15

Старинцева Наталья - Team Lead

Власова Ольга
Кобзева Мария
Мюлинг Илья
Стрик Наталья
Халевин Кирилл

Анализ и прогнозирование конверсии пользователей

• Тема: Анализ и прогнозирование конверсии с помощью ML-моделей и развертывание через API

• Цель: Определить ключевые факторы конверсии, построить модель с максимальной точностью (AUC-ROC) и создать продакшн-решение

• Методы: Сравнение моделей (Logistic Regression, CatBoost, XGBoost, LightGBM, нейросети) + FastAPI для развертывания

Старинцева Наталья

Team Lead + EDA Lead

Координация
Очистка данных + основной EDA
Итоговая документация

Власова Ольга

Data Quality Analyst

+ Feature Engineering Lead

Оценка полноты данных (% пропусков)
Выявление дубликатов
Выводы о качестве данных
Создание фичей для ML

Мюлинг Илья

ML Engineer

Самостоятельная разработка модели
Работа с готовыми фичами
Создание предсказательной модели

Халевин Кирилл

API Developer

Создание API для модели

Кобзева Мария

Business Analyst + QA Engineer

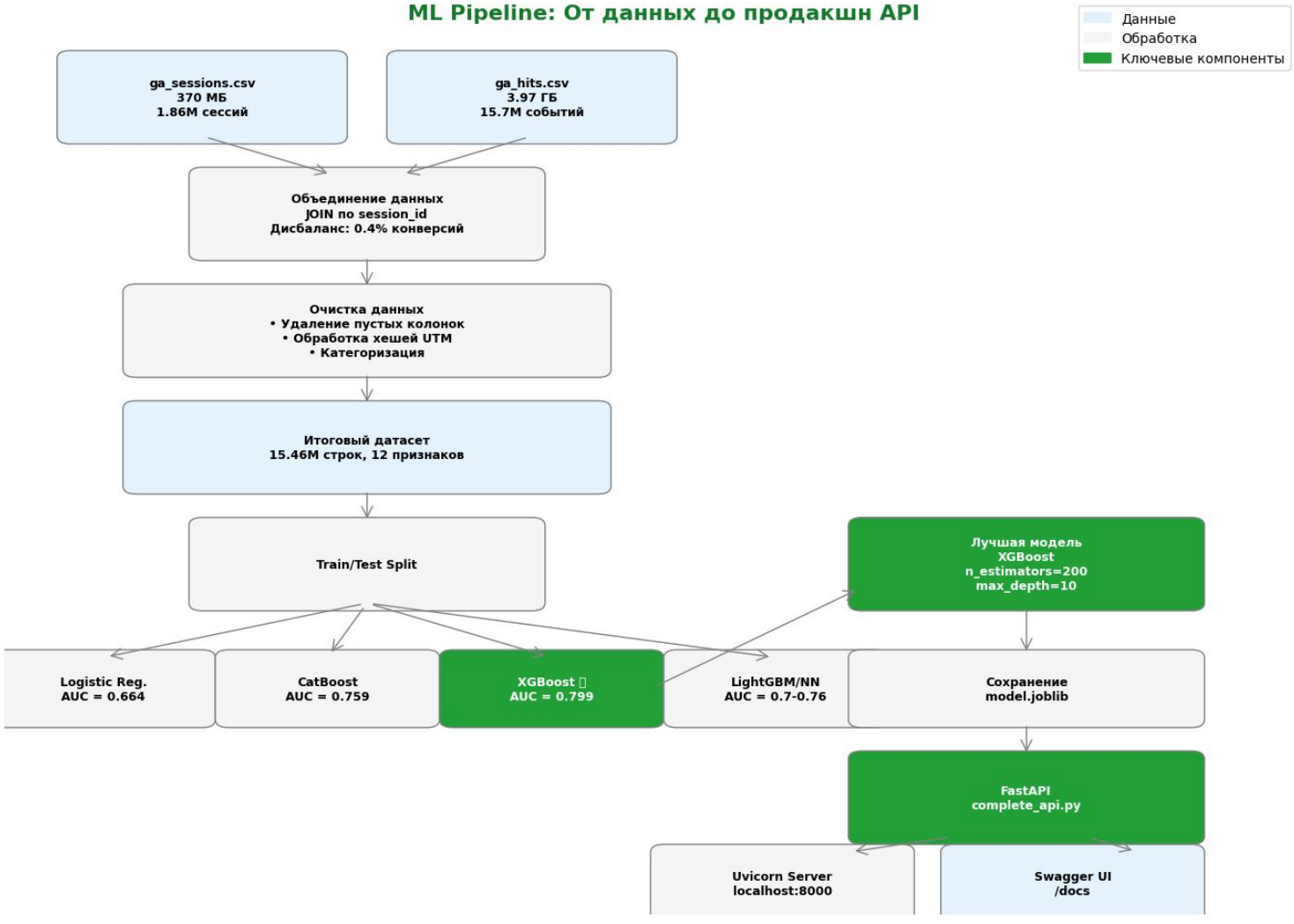
Определение целевых действий
Бизнес-логика и интерпретация
Тестирование всех компонентов

Стрик Наталья

Data Structure Analyst + Documentation Lead

Загрузка датасета и анализ структуры
Описание структуры данных
Анализ типов данных
Определение целевых действий

Команда № 15

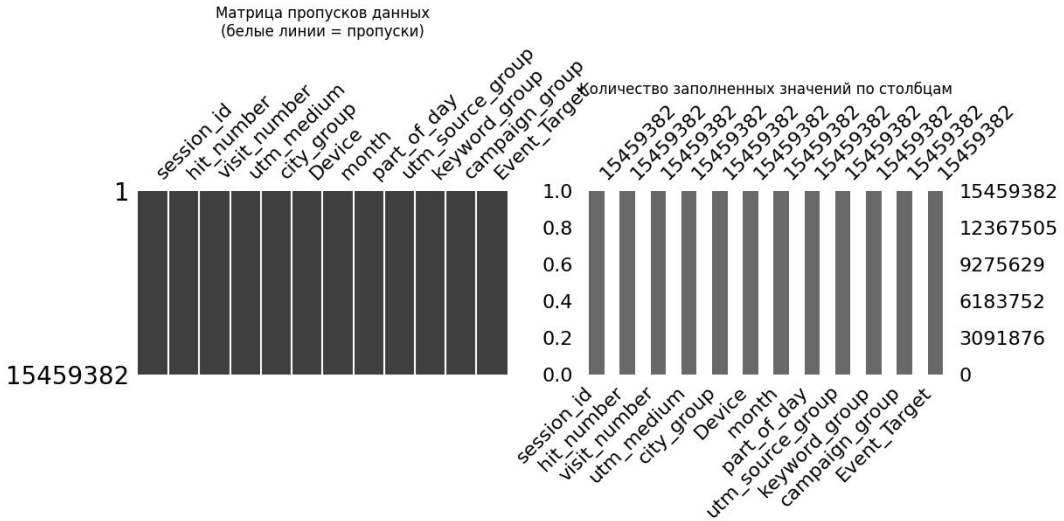

ML Pipeline: От данных до продакшн API
Данные
Обработка
Ключевые компоненты
ga_sessions.csv
370 МБ
1.86M сессий
ga_hits.csv
3.97 ГБ
15.7M событий
Объединение данных
JOIN по session_id
Дисбаланс: 0.4% конверсий
Очистка данных
• Удаление пустых колонок
• Обработка хешей UTM
• Категоризация
Итоговый датасет
15.46M строк, 12 признаков
Train/Test Split
Лучшая модель
XGBoost
n_estimators=200
max_depth=10
Logistic Reg.
AUC = 0.664
CatBoost
AUC = 0.759
XGBoost
AUC = 0.799
LightGBM/NN
AUC = 0.7-0.76
Сохранение
model.joblib
FastAPI
complete_api.py
Uvicorn Server
localhost:8000
Swagger UI
/docs

Обработка и анализ данных

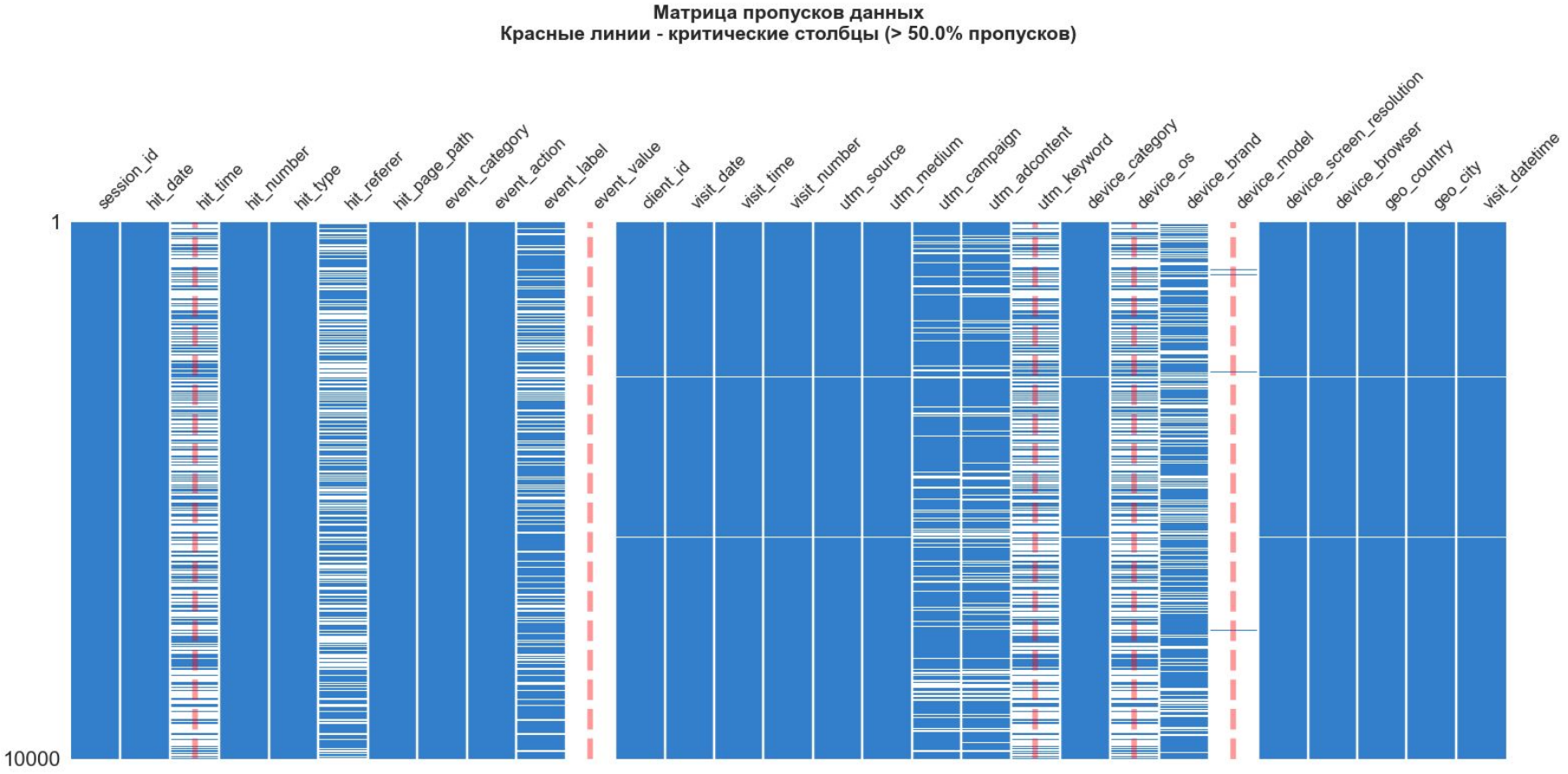
Исходные данные:

• ga_sessions.csv (370 МБ, 1.86M сессий) + ga_hits.csv (3.97 ГБ, 15.7M событий)

• Объединение по session_id → итоговый датасет: 15.46M строк, 12 признаков



Дисбаланс классов: Конверсия — 0.4% (61,473 из 15.46M)



Обработка и анализ данных

Ключевые этапы очистки:

• Удаление пустых колонок (event_value) и малоинформативных (hit_type)

• Обработка хешированных UTM-меток → группировка

• Категоризация устройств, временных периодов, географии

```
Index: 15459382 entries, 0 to 15726469
Data columns (total 12 columns):
 #   Column            Non-Null Count     Dtype
---  ------            --------------     -----
 0   session_id        15459382 non-null  object
 1   hit_number        15459382 non-null  int64
 2   visit_number      15459382 non-null  int64
 3   utm_medium        15459382 non-null  category
 4   city_group        15459382 non-null  category
 5   Device            15459382 non-null  category
 6   month             15459382 non-null  category
 7   part_of_day       15459382 non-null  category
 8   utm_source_group  15459382 non-null  category
 9   keyword_group     15459382 non-null  category
 10  campaign_group    15459382 non-null  category
 11  Event_Target      15459382 non-null  int64
dtypes: category(8), int64(3), object(1)
memory usage: 707.7+ MB
```

Результаты моделирования и сравнение алгоритмов

Сравнение моделей по AUC-ROC:

• Baseline (Logistic Regression): AUC = 0.664

• CatBoost: AUC = 0.759

• Random Forest: AUC = 0.724

• XGBoost (оптимизированный): AUC = 0.81 (лучший)

• Нейросети: AUC ~0.7–0.76

• LightGBM: AUC = 0.766

Ключевые выводы:

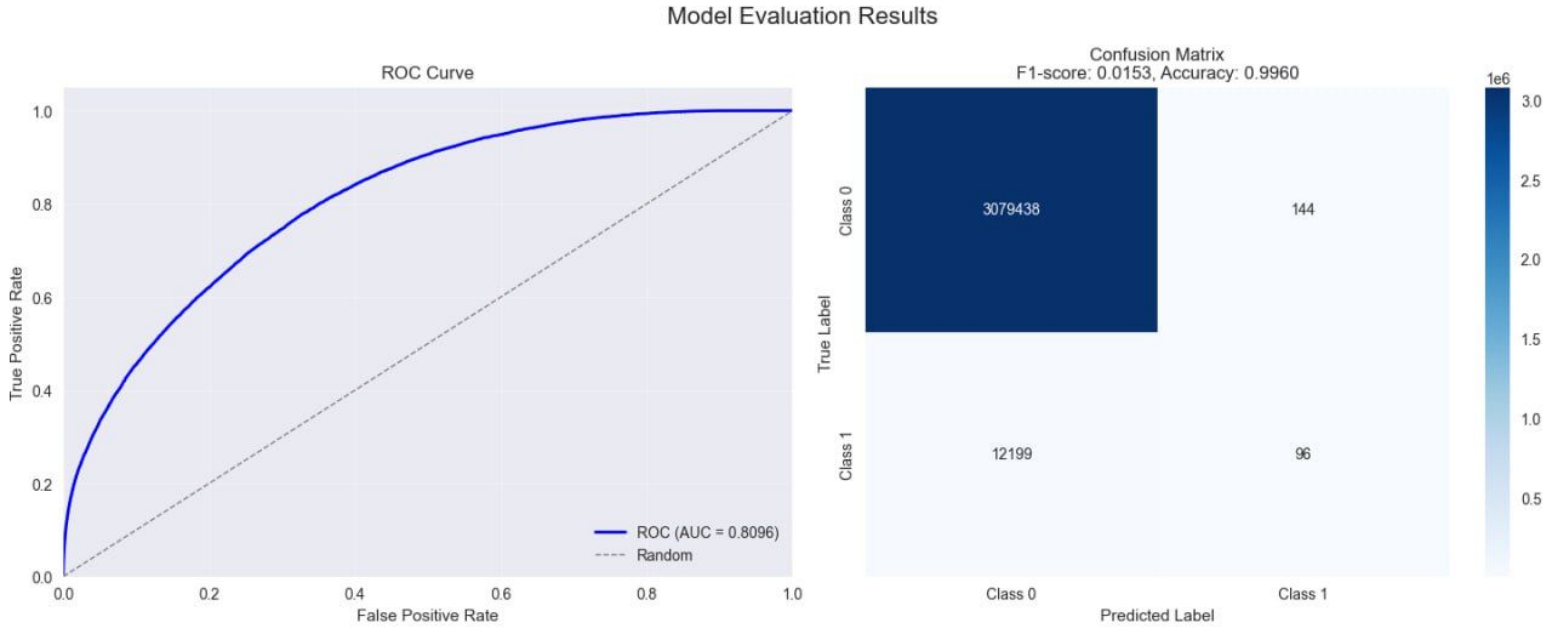
• Градиентный бустинг показывает лучшие результаты

• Оптимизация гипер параметров XG Boost дает прирост +2-4%

• XG Boost эффективен с категориальными признаками

• XG Boost самый быстрый в обучении алгоритм

Лучшая модель: XGBoost (AUC 0.81)



ROC-кривая демонстрирует стабильное превосходство модели над случайным классификатором. Площадь под кривой 0.81указывает на хорошее качество разделения классов и практическую применимость.

Анализ важности признаков и инсайты

Топ-3 признака:

1. hit_number — количество событий в сессии

2. visit_number — повторные визиты увеличивают конверсию

3. part_of_day_day — дневные посещения более результативны

Дополнительные инсайты:

• Поведенческие метрики важнее демографических

• Москвичи и пользователи Apple/десктопов конвертируются чаще

• UTM-источники имеют среднюю предсказательную силу

Анализ важности признаков и инсайты

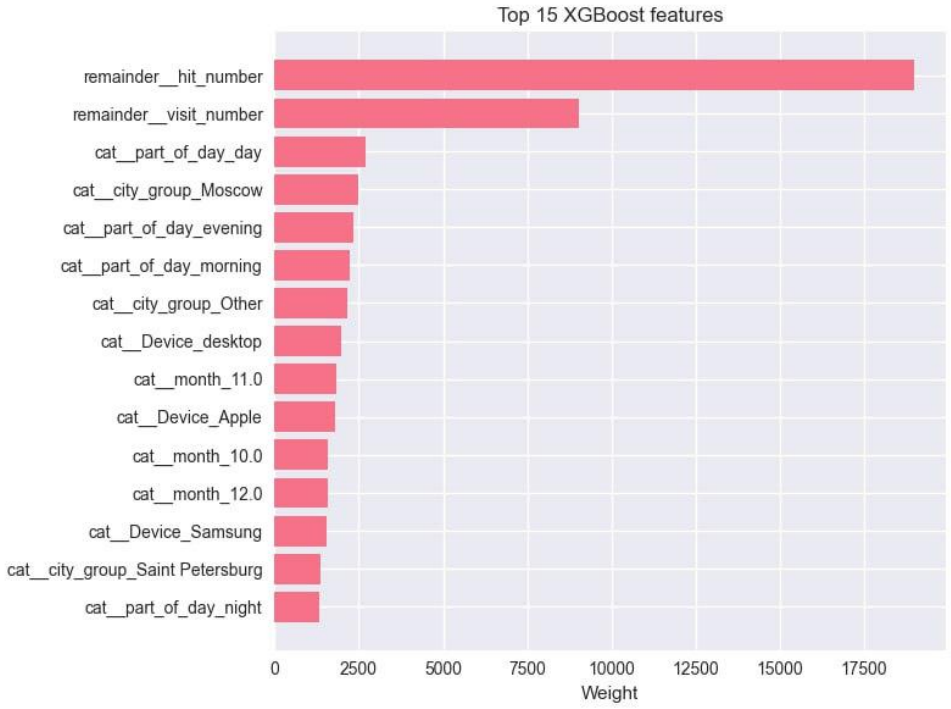
График важности признаков - feature importance из XGBoost

Рекомендации:

• Упростить пользовательский опыт для новых посетителей

• Добавить механизмы удержания после 3-го действия в сессии

• Сосредоточить маркетинговые усилия на дневное и вечернее время



Продакшн-решение

Лучшая модель: XGBoost (AUC 0.81)

• n_estimators=219, max_depth=14, learning_rate=0.12

API-решение (FastAPI):

• REST API для real-time предсказаний

• Swagger UI для тестирования

• 10 входных признаков → бинарное предсказание (0/1)

• Низкая задержка ответа

Технический стек:

• Python, scikit-learn, XGBoost, FastAPI, Uvicorn

• Готово для Docker/Kubernetes

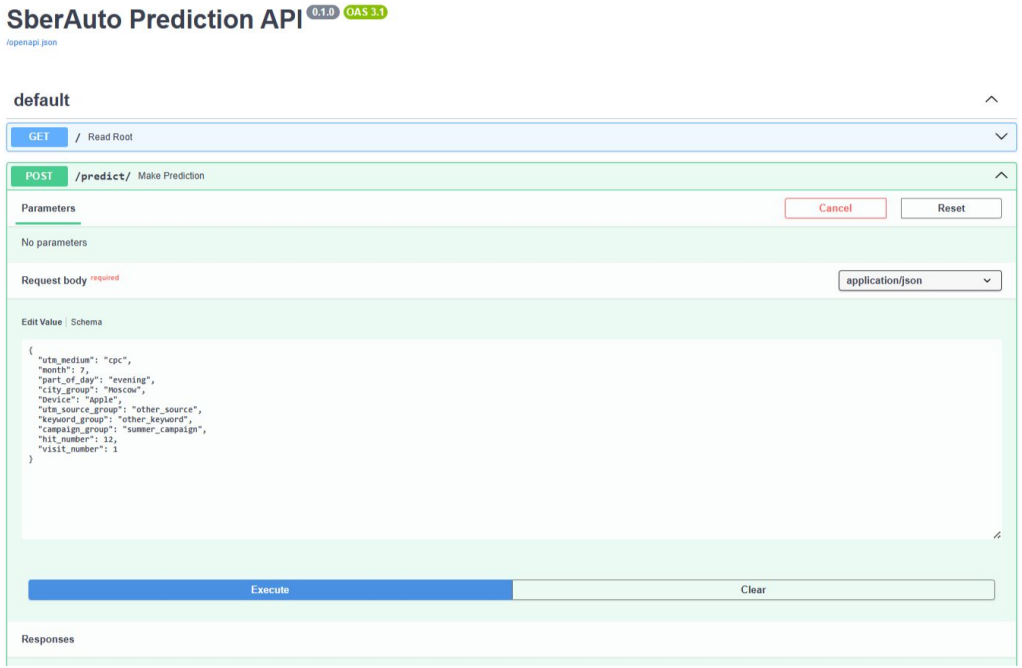
Интерфейс запроса POST /predict/

Поле для ввода JSON с 10 входными параметрами

Кнопки Execute (выполнить) и Clear (очистить)

Готовый пример данных для тестирования

Удобный веб-интерфейс Swagger UI



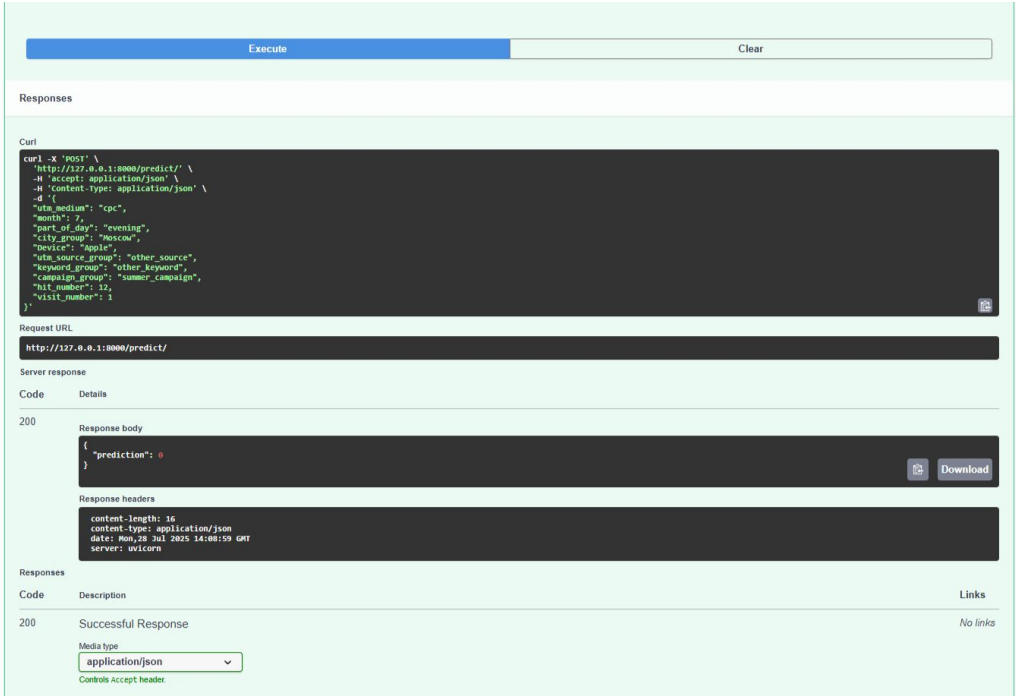
Результат выполнения запроса

HTTP статус 200 OK - успешное выполнение

Response body содержит prediction: 0

cURL команда для программного доступа

Код ответа сервера в формате JSON



Выводы

Дальнейшие шаги:

- Мониторинг качества модели в продакшене
- Переобучение на новых данных
- Оптимизация воронки конверсии на основе выявленных паттернов
- A/B-тестирование рекомендаций