

RAG-ассистент для подбора игр в Steam

Интеллектуальная система рекомендаций на основе LLM

Команда проекта "Maxsim Crew"

Виталик

Тимофей

Куаныш

Иван

Максим

Цель проекта: Глубокое понимание игровых предпочтений

Понимание запросов

Система должна интерпретировать свободные текстовые запросы игроков, выходя за рамки простых фильтров.

Смысловой подбор

Подбирать игры не по формальным признакам, а по настроению, жанровым предпочтениям и сложным описаниям.

Технологическая основа

Использование связки LLM + RAG и backend-тулзов для извлечения реальных данных об играх из Steam.

Как работает RAG-ассистент?

1

Обработка запроса

Принимает сложный, субъективный текстовый запрос от пользователя.

2

Извлечение данных

Вызывает LLM для получения подробного описания, тегов, жанров на основе запроса.

3

Формирование контекста

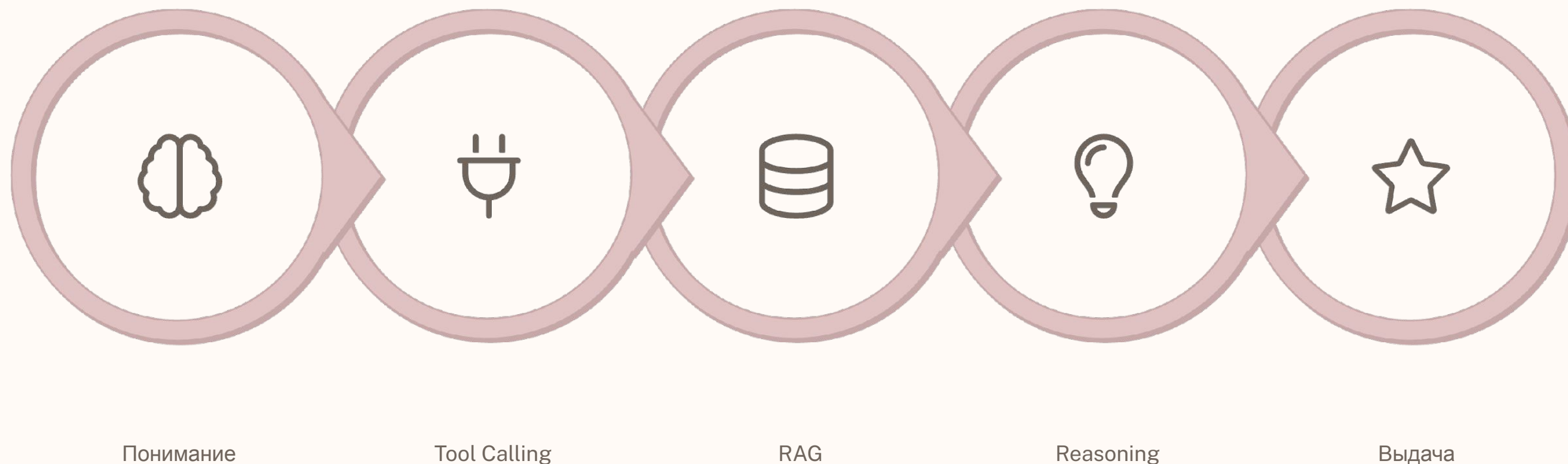
Вызывает backend-инструменты, взаимодействующие с локально развернутой моделью (index + knn), для нахождения подходящих игр на основе эмбедингов.

4

Генерация рекомендаций

Строит финальный список игр, на основе их релевантности по мнению LLM.

Архитектура решения: Интеллектуальный Пайплайн



LLM действует как интеллектуальный агент, исследующий данные и формирующий персонализированные рекомендации.

Локальная модель эмбедингов и поисковый индекс

01

Отбор популярных Steam-игр
По параметрам популярности

02

Создание векторных
представлений
(эмбедингов) для каждой игры

03

Формирование эмбедингов
На основе: описание игры, теги, жанры,
дополнительные игровые атрибуты

04

Создание индекса
С поиском по семантической близости

05

Обеспечение поиска игр
Ближких по смыслу и ощущению, а не только по жанру

Этот индекс играет роль 'локального поиска' в RAG-пайплайне и повышает качество retrieval.

Валидация качества: Строгие Тесты

Набор запросов

Создан набор из **499 реальных пользовательских запросов** различной сложности и типа.

Типы запросов

- Жанровые
- Атмосферные
- Сюжетные
- Сравнения с другими играми
- "Игры под настроение"

Для каждого запроса проводилось сравнение между нашей моделью и baseline-моделью, чтобы объективно оценить преимущества RAG-ассистента. Это позволило нам убедиться в способности системы справляться с нюансами пользовательских предпочтений.

Использованные метрики качества

1

Jaccard Overlap

Измеряет пересечение списков рекомендованных игр.

2

Cosine Similarity

Оценивает семантическую близость смыслового содержания ответов.

3

Semantic RBO

Учитывает порядок следования рекомендаций в списках.

4

Semantic NDCG

Комплексный показатель качества ранжирования рекомендаций.

5

Integrated Score

Общий интегрированный показатель качества работы системы.

Все метрики были рассчитаны и проанализированы в `validation.ipynb`.

Ключевые Результаты

Jaccard Overlap — 0.162

Совпадения редкие (что ожидаемо, т.к. LLM даёт разнообразные рекомендации).

Cosine Similarity — 0.369

- Обе модели дают семантически близкие игры, но с разной вариативностью.
- Списки отличаются по названиям, но схожи по «жанровому смыслу».

Semantic RBO — 0.240

Модель формирует отличающуюся от baseline стратегию ранжирования.

Semantic NDCG — 0.911

- Почти идеальное соответствие «полезности» выдаваемых игр baseline-модели.
- Это практически значит, что модель понимает запрос и выдаёт релевантное.

Integrated Score — 0.485 (48.5%)

Такое значение означает, что модель попадает в смысл, но выдаёт альтернативные рекомендации, а не повторяет baseline, причём порядок также отличается.

Итоговые Выводы: Успех RAG-ассистента

1 Точная Интерпретация

Система правильно интерпретирует сложные пользовательские запросы.

2 Эффективное Извлечение Данных

Успешно использует backend для извлечения релевантных игровых данных.

3 Качественные Рекомендации

Выдает осмысленные и высококачественные рекомендации.

4 Значительное Улучшение

RAG значительно повышает качество подбора игр по сравнению с традиционными методами.