АЛЬФА БУДУЩЕЕ

Альфа Банк

АЛЬФА-БУДУЩЕЕ ХАКАТОН

Настройка RAG для вопросов и ответов

Создайте интеллектуальный pipeline RAG-системы, которая по пользовательскому запросу находит наиболее релевантные фрагменты в корпусе данных.

Постановка задачи

Разработайте pipeline RAG-системы, которая по входному вопросу находит релевантные фрагменты в предоставленном корпусе данных. Для этого:

1

Изучите метод генерации с дополнительной выборкой (RAG).

2

Изучите структуру представленного корпуса текстовых данных с вопросами пользователей и распарсенных с сайта alfabank.ru подстраниц.

3

Постройте retrieval-пайплайн, который возвращает топ-5 документов для каждого вопроса.

4

Отправьте решение на платформу в формате файла .csv и узнайте его качество в виде оценки с помощью метрики Hit@5.

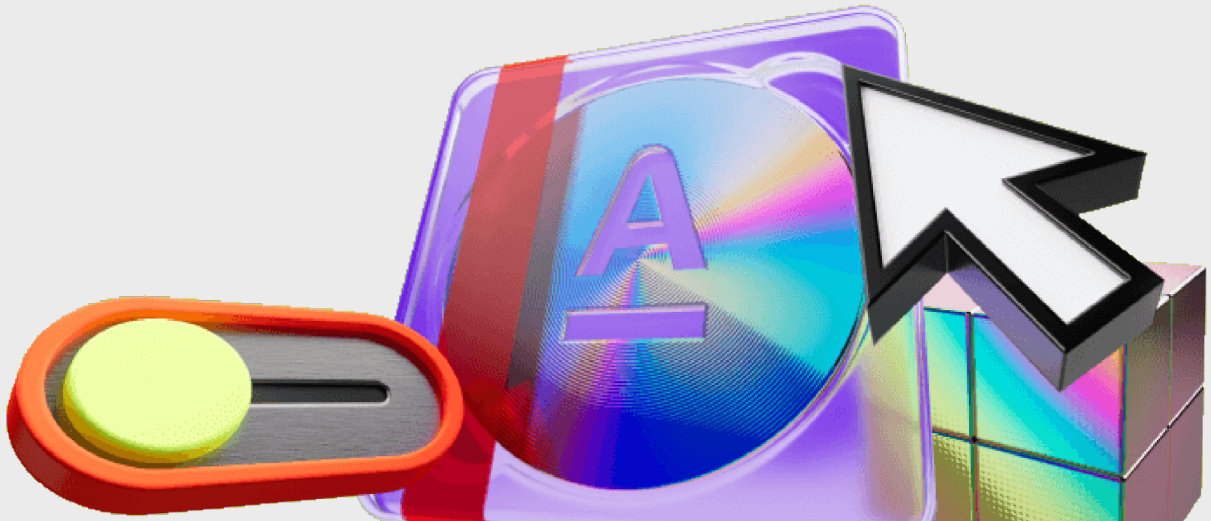
5

Доработайте ваше решение и повторно отправьте его на сайт для повышения результата.

Бизнес-контекст

Вопросно-ответные (Q&A) системы — это прикладные ИИ-сервисы, которые принимают текстовый вопрос и возвращают точный, краткий ответ. В классическом варианте есть два подхода: экстрактивный (фраза-ответ берется из источника) и абстрактивный (ответ генерируется моделью по доступной в ее базе информации). Главные требования к Q&A: точность, опора на источники (evidence), контроль галлюцинаций и понятная трассируемость, возможность проследить, откуда взялся факт.

RAG (Retrieval-Augmented Generation) — это архитектура, которая решает проблему «галлюцинаций» и устаревших знаний у языковых моделей. Сначала retriever — часть пайплайна, которая отвечает за поиск и экстракцию нужного фрагмента текста, — находит релевантные фрагменты в вашей базе знаний. Затем generator — языковая модель — формирует ответ, опираясь именно на эти фрагменты. Тем самым знания подгружаются динамически: вы обновляете корпус — и система отвечает по новым данным без дообучения самой модели.



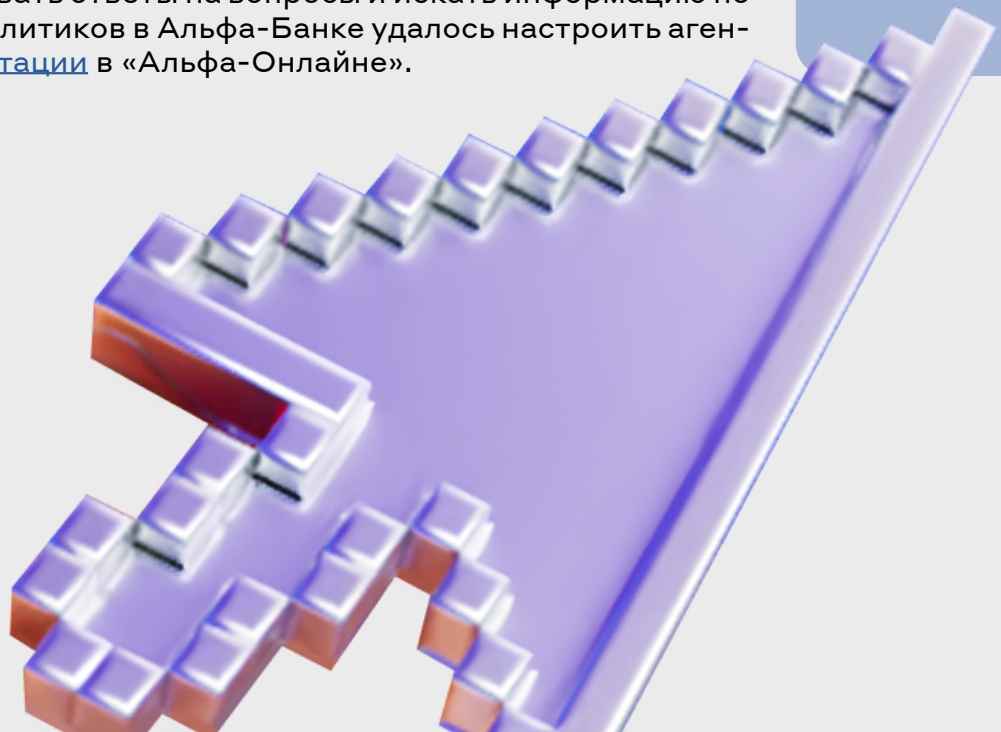
Как это работает по шагам:

- Индексирование. Текстовые данные разбиваются на чанки (chunks), для каждого чанка рассчитываются эмбеддинги, после чего строится векторный индекс для эффективного поиска.
- Извлечение. Для запроса генерируются эмбеддинги, и на основе этого выполняется поиск в индексе, возвращая топ-k ближайших чанков. Для повышения точности результатов часто используется переранжировка с применением cross-encoder.
- Промптование. Отобранные текстовые фрагменты передаются в модель с инструкцией: «Используй только данные из контекста, ссылайся на источники. Если информация не найдена, сообщи, что данных нет».
- Генерация. Языковая модель формирует ответ, сохраняя необходимый стиль, длину и ссылки. Также могут быть добавлены верификаторы, такие как self-check, answerability, и цитирование для повышения достоверности ответа.

Бизнес-контекст

Языковая модель в этой архитектуре появляется только на финальном этапе, так как выступает не источником знаний, а обобщает и связно излагает релевантный контекст, выделяет факты, корректирует неоднозначность формулировок и выдает финальный ответ. Большая часть качества идет не из огромной модели, а из инженерии пайплайна: размер чанков, нормализация и очистка разметки, метрики близости эмбеддингов и параметры поиска, реранжировка, промптинг инструкции, настройка фильтров токсичности.

RAG-системы активно реализуются в чат-ботах, в том числе в банковских приложениях, улучшая качество генерируемого ответа. Так, Альфа-Банк начал использовать ИИ для помощи в выборе дебетовых карт. В 2023 году в Альфа-Банке началась разработка ИИ-платформы, позволяющей ускорить выполнение рутинных задач сотрудников, AlfaGen, объединяющей несколько погруженных в контекст работы в Альфа-Банке благодаря методу RAG языковых моделей. ИИ-агенты на базе RAG позволяют не только успешно генерировать ответы на вопросы и искать информацию по базе знаний. Команде системных аналитиков в Альфа-Банке удалось настроить агента на проверку технической документации в «Альфа-Онлайне».



Критерии успеха

Основной критерий успешности решения — качество извлечения релевантных документов для каждого вопроса. Оценка проводится по метрике Hit@5 на публичной (public, 70%) тестовой части: решение считается успешным, если среди топ-5 найденных документов регулярно присутствует хотя бы один релевантный.

Ограничения

Для создания и настройки модели вы можете использовать язык Python версии 3.10 и выше. Также вы можете использовать любую библиотеку для машинного обучения и предобработки данных, являющуюся OpenSource-ресурсом. Использование закрытых библиотек, частных API и чужого кода, не подпадающего под разрешение о свободном распространении и использовании, запрещено.

Описание данных

ССЫЛКА НА МАТЕРИАЛЫ

Для разработки pipeline RAG-системы вы будете работать с таблицами:

1. Questions.csv — база вопросов, для которой вы настраиваете поисковую систему. Она состоит:
   a) из q_id — индивидуального номера каждого вопроса;
   b) query — содержания вопроса.

2. Websites.csv — база знаний поисковой функции:
   a) web_id — индивидуальный номер каждой веб страницы;
   b) url — адрес веб-страницы;
   c) kind — тип импортированных данных;
   d) title — название страницы, ее заголовок;
   e) text — колонка, содержащая заранее спарсенный со страницы текст.

В результате вашей работы вы должны отправить на платформу submit.csv, файл-пример вашего сабмита лежит на диске.

Пространство решений

Вы имеете право использовать любую модель из любой OpenSource-библиотеки для векторизации текста и чанкования. Выбор и настройка модели не ограничены организаторами.

Приложение 1. Метрика Hit@K

Результат метрики Hit@k

Метрика Hit@k оценивает качество ретривера (поиска фрагментов) в задачах «вопрос-ответ» и RAG. Она показывает, у какой доли вопросов среди первых k найденных результатов присутствует хотя бы один релевантный элемент.

Для запроса q ретривер возвращает упорядоченный список результатов $Rq=[r1,\ldots,rk,\ldots]$. Пусть множество релевантных для q элементов — Gq. Тогда попадание для запроса определяется так:

$$hit@k(q) = \begin{cases} 1, & \text{если } \{r_1,\ldots,r_k\} \cap G_q = \varnothing, \\ 0, & \text{иначе.} \end{cases}$$

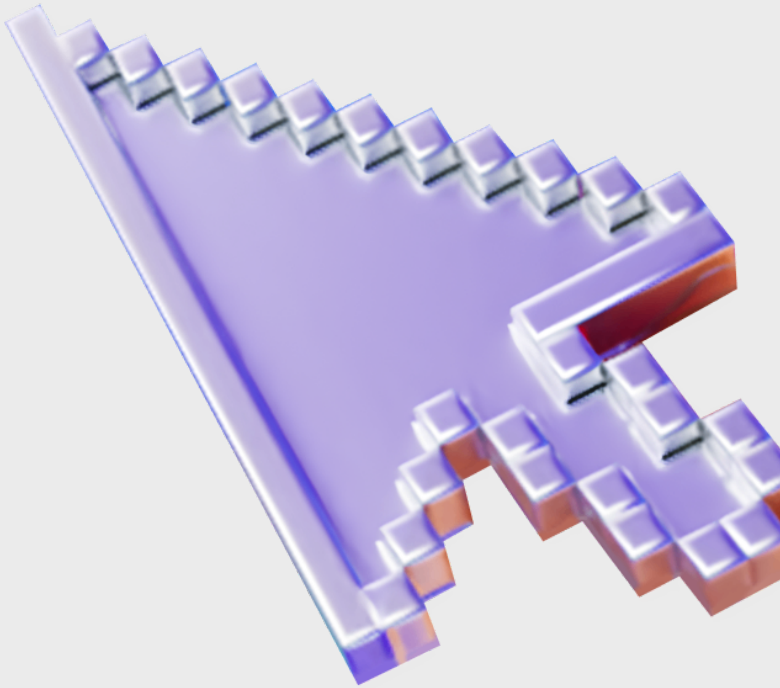
Итоговая метрика по выборке вопросов Q считается как среднее значение по всем вопросам, для которых есть хотя бы одна размеченная релевантная страница (вопросы без разметки в оценку не входят):

$$Hit@k = \frac{1}{|Q|} \sum_{q \in Q} hit@k(q).$$

где Q — подмножество вопросов с непустым набором релевантных страниц.

Интерпретация

Значение Hit@5=0,83 означает, что в среднем система возвращает в топ-5 большую часть релевантных страниц, причём более важные страницы (с большим весом) дают больший вклад в итоговый балл. Чем ближе метрика к 1, тем выше качество работы ретривера.



CHANGELLENGE >>

Альфа Банк

Кейс написан и опубликован Changellenge >> — ведущей организацией по кейсам в России.

www.changellenge.com

Кейс создан совместно с АО «Альфа-Банк»

www.alfabank.ru