Построение вопросно-ответной системы с использованием RAG (retrieval-augmented generation)

1. Структура проекта:

```
app/
└── fastapi_back/
        ├── .dockerignore
        ├── Dockerfile -- создание образа из наших файлов приложения
        ├── LICENSE
        ├── README.md
├── requirements.txt  - список зависимостей окружения для fastapi
        └── src/
            ├── __init__.py
            ├── api/
            │   └── v1/
            │       ├── api_route.py  -- Основные эндпоинты для взаимодействия с моделью
            │       ├── schemas.py  -- Схемы вопросов и ответов
            │       └── tfidf_pretrained.joblib  -- Предобученная модель tf-idf
            ├── logger.py  -- Конфигурация инструмента для логирования
            ├── main.py -- Основной файл для запуска приложения
            ├── qdrant/
            │   └── load_qdrant.py  -- Функции для интеграции с векторной БД qdrant
            └── tests/
                ├── conftest.py  -- Конфигурация тестов
                ├── full_dataset.csv  -- Датасет для тестов
                └── test_model.py  -- Тесты
└── grafana/provisioning/datasources/graf_loki.yaml – файл настройки мониторинга приложения и логов
└── streamlit/
    ├── config.toml - конфигурация темы приложения
    ├── .dockerignore
    ├── Dockerfile -- создание образа из наших файлов приложения
    ├── eda.py - функции для отрисовки графиков, препроцессинг текстов
    ├── project_logger.py - модуль логирования, установки и функция-обертка
    ├── st_app.py - основной код приложений
    ├── validat_df.py - модуль, отвечающий за валидацию в загружаемого приложение файла
    ├── requirements.txt  - список зависимостей окружения для streamlit
└── docker-compose.yml
└── promtail.yaml – файл настройки сборщика логов
└── loki.yaml – файл настройки сборщика логов
```

2. Функционал API

- Загрузка датасетов
- Обучение моделей (в настоящее время поддерживается TF-IDF)
- Интеграция с векторной базой данных Qdrant для эффективного поиска по схожести
- Загрузка и выгрузка моделей из оперативной памяти (TODO)
- Поиск контекста для заданных вопросов

- Тестирование качества и скорости модели
- Вывод списка загруженных наборов данных и обученных моделей
- Удаление моделей (по одной или всех сразу)

API Endpoints

- `POST /api/v1/models/load_dataset`: Загрузить датасет
- `POST /api/v1/models/fit_save`: Обучить модель и сохранить в Qdrant
- `POST /api/v1/models/load_model`: Загрузить модель в оперативную память
- `POST /api/v1/models/unload_model`: Выгрузить модели из оперативной памяти
- `POST /api/v1/models/find_context`: Найти контекст для заданного вопроса
- `POST /api/v1/models/quality_test`: Оценить точность и производительность модели
- `GET /api/v1/models/get_datasets`: Получить список загруженных наборов данных
- `GET /api/v1/models/list_models`: Получить список загруженных и обученных моделей
- `DELETE /api/v1/models/remove/{model_id}`: Удалить конкретную модель
- `DELETE /api/v1/models/remove_all`: Удалить все модели

3. Функционал streamlit-приложения

- Загрузка датасета и анализ данных
- Препроцессинг данных
- Построение графиков
- Конфигурирование и обучение модели
- Сравнение моделей
- Получение инференса

4. Запуск приложения

А) Приложение развернуто на VPS:

- http://178.130.43.233:8501/
- http://178.130.43.233:3000/ - Мониторинг и визуализацияя приложения и логов

Сервер очень слабый и полный функционал приложения показать не сможет, максимум допустимый датасет для использования на сервере должен содержать не более 100 записей.

Б) Запуск приложения на локальной машине:

Для запуска требуется

1. Установить по инструкции docker в зависимости от ОС
https://docs.docker.com/compose/install/
2. В папке app прописать команду для сборки и запуска приложения:
docker-compose -p app_rag up
3. Перейте по ссылке http://localhost:8501/ для входа в приложение
4. Для мониторинга приложения и сбора логов требуется:
- Перейте по ссылке http://localhost:3000/
- Подключить datasources loki, по адресу http://loki:3100/
- В разделе explore/loki - ведем мониторинг приложения и логов.

Инструкция использования в файле https://drive.google.com/file/d/1B3RqPt2BVKPCuhqpltTlJFle6sKCiP_t/view?usp=sharing