

Документация проекта

I. API

1. Описание структуры проекта

Проект состоит из нескольких файлов, каждый из которых выполняет свою уникальную функцию:

- `main.py`: Основной файл приложения, реализующий API на базе FastAPI. Он содержит маршруты для обучения модели и предсказания заработной платы на основе данных о вакансиях.
- `preprocessing.py`: Файл, содержащий функции для предобработки данных. Он включает в себя функции для преобразования названий вакансий в категории и подготовки данных для обучения модели.
- `schemas.py`: Файл, описывающий структуры данных, используемые в API. Он определяет модели для входных и выходных данных, включая вакансии, запросы на обучение модели и предсказания.
- `jobmarket_model.pkl`: Сохраненная модель, используемая для предсказания заработной платы. Этот файл необходим для работы эндпоинта `/predict`, так как он содержит обученные параметры модели.

2. Описание функционала API

1. `/fit`

Обучает модель на основе предоставленных данных о вакансиях.

- `FitRequest`: включает в себя данные о вакансиях и конфигурацию модели.
- `FitResponse`: сообщение о статусе обучения модели.

2. `/predict`

Выполняет предсказание заработной платы на основе данных о вакансиях с использованием обученной модели.

- `PredictRequest`: включает в себя идентификатор модели и данные о вакансиях.
- `PredictionResponse`: список предсказанных значений заработной платы.

3. Инструкция по использованию API

1. Запуск приложения:

- Убедитесь, что у вас установлен Python и необходимые библиотеки.

2. Обучение модели:

- Отправьте POST-запрос на эндпоинт `/fit` с данными о вакансиях и конфигурацией модели в формате JSON.

3. Предсказание заработной платы:

- Отправьте POST-запрос на эндпоинт `/predict` с идентификатором модели и данными о вакансиях в формате JSON.
- Получите предсказанные значения заработной платы.

II. Streamlit

1. Описание структуры проекта

Проект включает в себя Streamlit-приложение, которое взаимодействует с API для анализа вакансий и предсказания заработной платы. Основной файл приложения отвечает за пользовательский интерфейс, обработку взаимодействий и визуализацию данных.

Файлы проекта:

- requirements.txt: Список необходимых библиотек для установки.
- config.toml: Конфигурационный файл для Streamlit.
- streamline_service.py: Главный файл Streamlit-приложения.

2. Описание функционала Streamlit-приложения

1. Загрузка данных:

- Пользователь может загрузить CSV-файл с данными о вакансиях.

2. Анализ данных:

- Приложение предоставляет возможность проводить исследовательский анализ данных, включая:

- Вывод информации о загруженных данных.
- Анализ пропусков и дубликатов и визуализация распределения зарплат.
- Анализ профессиональных ролей и навыков.

3. Обучение модели:

- Пользователь может настроить гиперпараметры для обучения модели и отправить запрос на обучение модели через API.

- Возможность отображения кривых обучения.

4. Предсказание заработной платы:

- Пользователь может загрузить новый CSV-файл для предсказания заработной платы с использованием обученной модели.

3. Инструкция по использованию

1. Установка зависимостей:

- Убедитесь, что у вас установлен Python и необходимые библиотеки.

2. Запуск приложения:

- Запустите приложение

3. Использование приложения:

- Загрузите CSV-файл с данными о вакансиях.
- Выберите нужный раздел в меню (EDA, Обучение модели, Предсказание).
- Вводите параметры и нажимайте соответствующие кнопки для выполнения действий.

Для дальнейшего улучшения документации будут рассмотрены добавления следующих элементов:

- Примеры запросов: Включение примеров JSON-запросов и ответов для API.
- Обработка ошибок: Описание возможных ошибок и их решений для API и Streamlit-приложения.