

Описание кода и файлов в папке airpollpredictor2

В папке airpollpredictor2 сохранен код написанный Шамсутдиновой Аделиной (<https://github.com/Adele-Kim>), а также некоторые промежуточные датасеты.

1) [dataset_creator](#) содержит скрипт для создания датасетов по странам, городам или станциям, из ранее скачанных данных (<https://disk.yandex.ru/d/SVvFER5hMgcKtA>). Модуль запускается в терминале с необходимыми параметрами. Например:

```
python dataset_creator.py DK RS
```

создает датасет, включающий все измерения загрязнителей для Дании и Сербии.

```
python dataset_creator.py BERLIN
```

создает датасет, включающий все измерения загрязнителей для Берлина.

2) В папке [discomap_loader](#) находится альфа версия модуля для загрузки данных с сайта <https://discomap.eea.europa.eu/map/fme/AirQualityExport.htm> и возможностью создания датасета без промежуточного этапа сохранения файлов на диск. Его планируется использовать в виде модуля для автоматической подгрузки новых данных и актуализации train-датасета.

3) В [jupyter](#) хранятся ipynb-файлы этапа EDA, эксперименты с ML-моделями, а также первичные версии некоторых модулей, расположенных в основной папке.

4) В папке [kml_exporter](#) содержится скрипт для выгрузки информации о станциях в kml-формат. Файлы в данном формате можно открывать в Google Earth и проверять расположение станций и ближайших к ним населенных пунктов. Так как в метаданных не для всех станций указаны населенные пункты. Данный скрипт можно использовать для визуализации разных типов станций, а также, после небольшой модификации, и для визуализации индекса AQI на карте с использованием меток разного цвета.

В этой же папке хранится файл с kml-шаблонами и метаданные полученные объединением PanEuropean_metadata.csv и Airbase_v8_stations.csv.

5) Папка [other](#) содержит все что не вошло в предыдущие папки. Это датасет Дании, после этапа EDA, и метаданные Airbase_v8_stations.csv

6) `aqi_calculator.py` – скрипт для создания датасета в формате timeseries, содержащем только колонку с рассчитанным AQI усредненным по дням. Для последующего этапа ML. Для его работы необходим `subindex_calc.py` (содержит формулы для 6 загрязнителей).

7) `data_proc.py` – скрипт для обработки датасета, согласно выявленным на этапе EDA особенностям данных. На выходе получаем копию датасета пригодную для `aqi_calculator`. Запускается из терминала с двумя необходимыми параметрами: `--inputfile --outputfile`

8) `fastapi_starter.py` – простенький FastAPI модуль для загрузки данных и получения предсказаний обученной ранее модели.

9) `mini_ml.py` – минимальный базовый ML-пайплайн на основе простой линейной регрессии с использованием lag-фичей (предыдущих значений AQI) для предсказания текущего AQI.

10) `requirements.txt` содержит список необходимых модулей для работы кода, расположенного в данной папке.

Как все это использовать?

- 1) Создаем датасет с помощью `dataset_creator.py`
- 2) Обрабатываем его `data_proc.py`
- 3) Далее используем `aqi_calculator.py`
- 4) Полученный итоговый датасет используем в `mini_ml.py`

В дальнейшем планируется автоматизировать передачу данных из одного модуля в другой и объединить все это (а также разрабатываемый в данный момент модуль загрузки/подгрузки актуальных данных `discomap_loader`) в один единый сервис на основе FastAPI.