

BRepMocAutoencoder

Автоэнкодер для B-Rep представлений CAD моделей с поиском похожих объектов по эмбедингам.

Структура проекта

— LICENSE	<- Лицензия с открытым исходным кодом
— Makefile	<- Makefile с командами типа `make data` или `make train`
— README.md	<- Основной README для разработчиков проекта
— .env	<- Переменные окружения (не отслеживаются git)
— .gitignore	<- Файл игнорирования git
— environment.yml	<- Спецификация conda окружения
— pyproject.toml	<- Файл конфигурации проекта с метаданными пакета
— data	
— external	<- Данные из внешних источников
— interim	<- Промежуточные преобразованные данные
— processed	<- Финальные готовые датасеты для моделирования
— raw	<- Исходные неизменяемые данные
— docs	<- Файлы документации
— models	<- Обученные и сериализованные модели (best.ckpt)
— notebooks	<- Jupyter блокноты для экспериментов и тестирования
— notebook_setup.py	<- Скрипт настройки для блокнотов
— test_model.ipynb	<- Блокнот для тестирования модели
— references	<- Словари данных, руководства и пояснительные материалы
— reports	<- Сгенерированные отчёты в HTML, PDF, LaTeX, Excel и т.д.
— occwl	<- Библиотека-обёртка OpenCASCADE для CAD операций
— base.py	<- Базовые классы и утилиты
— compound.py	<- Операции со сложными формами
— edge.py	<- Обработка рёбер
— face.py	<- Обработка граней
— shape.py	<- Общие операции с формами
— viewer.py	<- 3D визуализация
— ...	<- Другие модули обработки CAD
— src	<- Исходный код проекта
— __init__.py	<- Делает src Python модулем
— config.py	<- Хранение полезных переменных и конфигурации
— dataset.py	<- Классы датасетов для обучения
— plots.py	<- Код для создания визуализаций
— metrics.py	<- Метрики оценки для поиска по сходству
— searcher.py	<- Функционал поиска по сходству

```

├── features                <- Модули извлечения признаков
│   ├── brep_features.py    <- Извлечение B-Rep признаков (BRepNetExtractor)
│
├── model                  <- Архитектура модели и обучение
│   ├── brep_autoencoder.py <- Реализация модели автоэнкодера
│
├── pipelines              <- Конвейеры обучения и вывода
│   ├── build_dataset.py    <- Подготовка датасета и статистики
│   ├── build_features.py   <- Конвейер извлечения признаков
│   ├── build_embeddings.py <- Генерация эмбеддингов из обученной модели
│   ├── build_metrics.py    <- Вычисление метрик оценки
│   ├── inference.py        <- Поиск по сходству
│   └── train.py            <- Конвейер обучения модели
│
└── utils                  <- Вспомогательные функции
    ├── file_utils.py       <- Операции с файлами
    ├── running_stats.py    <- Онлайн вычисление статистик
    ├── scale_utils.py      <- Утилиты масштабирования модели
    └── face_index_validator.py <- Валидация CAD моделей

```

Структура датасета

```

├── data
│   ├── external           <- Данные из внешних источников
│   │   ├── s2.0.0_step_all_features.json <- Конфигурация признаков STEP
│   │   ├── segment_names.json           <- Имена сегментов
│   │   ├── feature_lists/               <- Списки признаков
│   │   │   ├── all.json                 <- Все признаки
│   │   │   ├── no_curve_type.json       <- Без типа кривой
│   │   │   ├── no_edge_*.json           <- Без различных признаков рёбер
│   │   │   └── no_face_*.json           <- Без различных признаков граней
│   │   └── kernels/                     <- Конфигурации ядер
│   │       ├── asymmetric*.json         <- Асимметричные ядра
│   │       ├── simple_edge.json         <- Простое ребро
│   │       └── winged_edge*.json        <- Крылатые рёбра
│   ├── interim            <- Промежуточные преобразованные данные
│   ├── processed           <- Финальные готовые датасеты для моделирования
│   │   ├── dataset_stats.json <- Статистики датасета
│   │   ├── features/         <- Извлечённые признаки
│   │   │   ├── brep/         <- B-Rep признаки (.npz файлы)
│   │   │   └── embeddings/    <- Эмбеддинги моделей (.npz файлы)
│   │   └── steps/             <- STEP файлы CAD моделей (.stp)
│   └── raw                   <- Исходные неизменяемые данные

```

В конвейерах обработки уже заложены пути по умолчанию необходимо только либо перенести файлы CAD-моделей в папку steps, либо в `build_features.py` указать `step_path_dir` директорию, где находятся CAD модели

Основные компоненты

Конвейеры обработки (Pipelines)

- `build_features.py` - Извлечение B-Rep признаков из CAD моделей
- `build_dataset.py` - Подготовка датасета и вычисление статистик
- `train.py` - Обучение автоэнкодера
- `build_embeddings.py` - Генерация эмбеддингов из обученной модели
- `inference.py` - Поиск похожих моделей по эмбеддингам
- `build_metrics.py` - Вычисление метрик качества

Запуск конвейеров

```
# Извлечение признаков
python -m src.pipelines.build_features

# Подготовка датасета
python -m src.pipelines.build_dataset

# Обучение модели
python -m src.pipelines.train

# Генерация эмбеддингов
python -m src.pipelines.build_embeddings

# Поиск похожих моделей
python -m src.pipelines.inference

# Вычисление метрик
python -m src.pipelines.build_metrics
```

Для получения справки по параметрам каждого конвейера используйте флаг `--help`.