建筑领域（融合知识的）预训练语言模型

依赖库

- Python v3.8
- Numpy v1.21.2
- CUDA v10.1
- PyTorch v1.8.0
- Transformers v4.7.0
- LTP v4.1.5.post2

文件结构

```
├── data # 数据目录
│   ├── raw_text.txt # 原始建筑文本数据
│   └── triple.json # 原始知识三元组数据
├── model # 存放预训练所需的各种初始模型的目录
├── output # 存放输出结果的目录
├── config.json # 模型预训练的配置文件
├── config.schema # 模型预训练配置文件的标准格式与字段含义
├── dataset.py # 数据集类定义、数据加载
├── data_utils.py # 数据预处理工具函数
├── model.py # 模型定义
├── preprocess.py # 数据预处理脚本，生成结果存放在 data 中
├── train.py # 预训练脚本
├── unit_test.py # 各子功能的单元测试函数
└── utils.py # 通用工具函数、类
```

模型下载

- LTP 分词模型：http://39.96.43.154/ltp/v3/small.tgz
- 初始模型：https://cloud.tsinghua.edu.cn/f/0888c8802dab41c89366/?dl=1
- 预训练好的模型：https://cloud.tsinghua.edu.cn/f/0748856014124860ab2c/?dl=1

预训练流程

1. 下载 LTP 分词模型和初始模型，存放至 `model` 目录，并全部解压为单独目录 `LTP` 和 `chinese_roberta_wwm_ext`。
2. `python preprocess.py`，以在 `data` 目录中生成 `raw_text_seg.txt` 和 `seg_len_to_lm_ids_dict.pkl` 两个中间数据文件。
3. `python train.py --config config.json` 开始预训练。

直接使用方法

1. 下载预训练好的模型，解压至任意路径 `$PATH$`
2. 在 python 代码中插入以下片段：

```python
from transformers import BertTokenizer, BertModel
tokenizer = BertTokenizer.from_pretrained($PATH$)
model = BertModel.from_pretrained($PATH$)
```