

Homework 2 ADL NTU 2022

Environment

```
./download.sh
```

Preprocessing

```
python preprocess.py --do_train --do_eval --do_predict --context <context file> --train  
<train file> --valid <valid file> --test <test file>
```

Context Selection

重現Training過程

```
python run_mc.py --model_name_or_path <model name or path> --output_dir out_MC --  
per_gpu_train_batch_size 1 --save_steps 10000 --gradient_accumulation_steps 2 --  
max_seq_length 384 --learning_rate 3e-5 --num_train_epochs 1 --do_train --do_eval --  
train_file train_pro.json --validation_file valid_pro.json --overwrite_output_dir --  
do_predict --test_file test_pro.json
```

不load pretrained weights

```
python run_no_pre.py --model_name_or_path <model name or path> --output_dir out_MC_NO -  
-per_gpu_train_batch_size 1 --save_steps 2000 --gradient_accumulation_steps 2 --  
max_seq_length 384 --learning_rate 3e-5 --num_train_epochs 1 --do_train --do_eval --  
overwrite_output_dir --train_file train_pro.json --validation_file valid_pro.json --  
do_predict --test_file test_pro.json
```

Question Answering

重現Training過程

```
python run_qa.py --model_name_or_path <model name or path> --output_dir out_QA --  
per_gpu_train_batch_size 1 --save_steps 10000 --gradient_accumulation_steps 2 --  
max_seq_length 384 --learning_rate 3e-5 --num_train_epochs 1 --do_train --do_eval --  
train_file train_pro.json --validation_file valid_pro.json --overwrite_output_dir --  
do_predict --test_file test_pro.json
```

Postprocessing

將test的結果轉換成可以上傳到kaggle的格式

```
python postprocess.py --file ./out_QA/predict_predictions.json --out <predict result file>
```

Results

Model	Public Score	Private Score	Rank
MC+QA	0.78390	0.78139	92/170

Bonus

Preprocessing

```
python preprocess-slot.py --do_train --do_predict --train <train file> --valid <valid file> --test <test file>
```

Intent Classification Training

重現Training的結果(需要先將code裡354行開始的路徑改成本地的bert檔案路徑跟配合上次的code)

```
python train_intent.py --do_train --do_eval --do_predict --cuda=<cuda device> --num_epoch=<number of epoch>
```

Slot Tagging Training

重現Training的結果(需要配合上次的code)

```
python run_ner.py --model_name_or_path <model name or path> --output_dir ./slot_tag --per_gpu_train_batch_size 1 --save_steps 10000 --gradient_accumulation_steps 2 --max_seq_length 384 --learning_rate 3e-5 --num_train_epochs 1 --do_train --do_eval --overwrite_output_dir --label_column_name ner --text_column_name words --train_file train_slot_pro.json --validation_file valid_slot_pro.json --do_predict --test_file test_slot_pro.json

python postprocess-slot.py --file ./slot_tag/predictions.txt
```

Results

Model	Public Score	Private Score
Intent Classification	0.94133	0.93377
Slot Tagging	0.81233	0.80385