R10723050 財金碩二陳韻帆
Q1 Data processing

1.Tokenizer

a. 使用 wordpiece 的 Tokenizer，此 Tokenizer 的原理類似於 BPE Tokenizer，不同的點是 wordpiece 在挑選配對的字詞是用 pair score 來做挑選而不是出現頻率，

$$pairscore = \frac{freq\ of\ pair}{freq\ of\ first\ element * freq\ of\ second\ element}$$

演算法如下:
step1. 設定可容許的 vocabulary 大小
step2. 將詞切成一個一個字元 (含結束符號)，稱其為字元資料庫
step3. 將資料中的字元皆放入 vocabulary 庫中
step4. 選擇 pair score 最高的 pair 放入 vocabulary 庫，並將字元資料庫中的 pair 合併
step5. 反覆做 step4 直到 vocabulary 庫的大小到達設定標準
step6. 將 vocabulary 庫中 pair 好的字元透過 language model pre-training 得到含有意義的 token

2.Answer Span

a. 使用 BatchEncoding 中的 $char\_to\_token()$ 函式，可以將原本使用一個一個字元分割產生的位置編號改成輸入經過 Tokenization 後對應的位置編號

b. 把每對估計出的 $start/end$ 的機率相乘，保留機率最大的 pair，但要跳過 start 位置大於 end 位置的狀況。最後估計出的答案代回 $input\_ids$ 再套用 $tokenizer.decode()$ 含式得到預測目標。

Q2 Modeling with BERTs and their variants

1 a. my model: "$bert-base-chinese$"
MultipleChoiceConfig:


```
{
  "_name_or_path": "bert-base-chinese",
  "architectures": [
    "BertForMultipleChoice"
  ],
  "attention_probs_dropout_prob": 0.1,
  "classifier_dropout": null,
  "directionality": "bidi",
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.1,
  "hidden_size": 768,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "layer_norm_eps": 1e-12,
  "max_position_embeddings": 512,
  "model_type": "bert",
  "num_attention_heads": 12,
  "num_hidden_layers": 12,
  "pad_token_id": 0,
  "pooler_fc_size": 768,
  "pooler_num_attention_heads": 12,
  "pooler_num_fc_layers": 3,
  "pooler_size_per_head": 128,
  "pooler_type": "first_token_transform",
  "position_embedding_type": "absolute",
  "torch_dtype": "float32",
  "transformers_version": "4.23.1",
  "type_vocab_size": 2,
  "use_cache": true,
  "vocab_size": 21128
}
```

QAConfig:


```
{
  "_name_or_path": "bert-base-chinese",
  "architectures": [
    "BertForQuestionAnswering"
  ],
  "attention_probs_dropout_prob": 0.1,
  "classifier_dropout": null,
  "directionality": "bidi",
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.1,
  "hidden_size": 768,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "layer_norm_eps": 1e-12,
  "max_position_embeddings": 512,
  "model_type": "bert",
  "num_attention_heads": 12,
  "num_hidden_layers": 12,
  "pad_token_id": 0,
  "pooler_fc_size": 768,
  "pooler_num_attention_heads": 12,
  "pooler_num_fc_layers": 3,
  "pooler_size_per_head": 128,
  "pooler_type": "first_token_transform",
  "position_embedding_type": "absolute",
  "torch_dtype": "float32",
  "transformers_version": "4.23.1",
  "type_vocab_size": 2,
  "use_cache": true,
  "vocab_size": 21128
}
```


b. performance of my model
Multiple Choice Accuracy:0.9458
QA Accuracy: 0.7438

public score(EM)： 0.6962

c. Loss function:
Cross entropy loss

d. Training argument
Multiple choice:

optimization algorithm: AdamW
learning rate:3e−5(使用 lr_scheduler.StepLR，參數:step_size=200, gamma=0.92)
batch_size:1

Question answering:

optimization algorithm: AdamW
learning rate:3e−5(使用 lr_scheduler.StepLR，參數:step_size=400, gamma=0.9)
batch_size:8

2 a. my model: "$hfl/chinese-roberta-wwm-ext-large$"

MultipleChoiceConfig:


```
"_name_or_path": "hfl/chinese-roberta-wwm-ext",
"architectures": [
  "BertForMultipleChoice"
],
"attention_probs_dropout_prob": 0.1,
"bos_token_id": 0,
"classifier_dropout": null,
"directionality": "bidi",
"eos_token_id": 2,
"hidden_act": "gelu",
"hidden_dropout_prob": 0.1,
"hidden_size": 768,
"initializer_range": 0.02,
"intermediate_size": 3072,
"layer_norm_eps": 1e-12,
"max_position_embeddings": 512,
"model_type": "bert",
"num_attention_heads": 12,
"num_hidden_layers": 12,
"output_past": true,
"pad_token_id": 0,
"pooler_fc_size": 768,
"pooler_num_attention_heads": 12,
"pooler_num_fc_layers": 3,
"pooler_size_per_head": 128,
"pooler_type": "first_token_transform",
"position_embedding_type": "absolute",
"torch_dtype": "float32",
"transformers_version": "4.23.1",
"type_vocab_size": 2,
"use_cache": true,
"vocab_size": 21128
```

QAConfig:


```
"_name_or_path": "hfl/chinese-roberta-wwm-ext-large",
"architectures": [
  "BertForQuestionAnswering"
],
"attention_probs_dropout_prob": 0.1,
"bos_token_id": 0,
"classifier_dropout": null,
"directionality": "bidi",
"eos_token_id": 2,
"hidden_act": "gelu",
"hidden_dropout_prob": 0.1,
"hidden_size": 1024,
"initializer_range": 0.02,
"intermediate_size": 4096,
"layer_norm_eps": 1e-12,
"max_position_embeddings": 512,
"model_type": "bert",
"num_attention_heads": 16,
"num_hidden_layers": 24,
"output_past": true,
"pad_token_id": 0,
"pooler_fc_size": 768,
"pooler_num_attention_heads": 12,
"pooler_num_fc_layers": 3,
"pooler_size_per_head": 128,
"pooler_type": "first_token_transform",
"position_embedding_type": "absolute",
"torch_dtype": "float32",
"transformers_version": "4.23.1",
"type_vocab_size": 2,
"use_cache": true,
"vocab_size": 21128
```

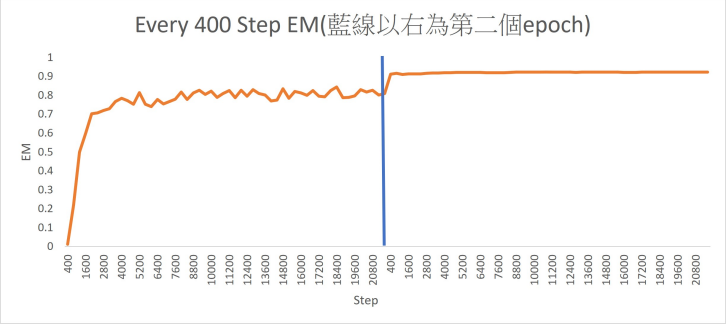

b. performance of my model
Multiple Choice Accuracy:0.953
QA Accuracy: 0.825
public score(EM)： 0.77576

c. RoBERTa 與 BERT 的不同之處在於 RoBERTa 的 Mask 是使用 Dynamic masking 的方式，也就是在訓練的過程會動資料給予不同的 masking，而不像 BERT 在輸入 Data 時就決定唯一的 masking 方式。
wwm 代表的是 whole word masking，是指在做 masking 時不會只對字元做 masking，而是會對整個詞作 Masking，其他不同的部份是對資料量、BatchSize，訓練 Step 數等作修改。

d. BERT -> RoBERTa-wwm-ext-large

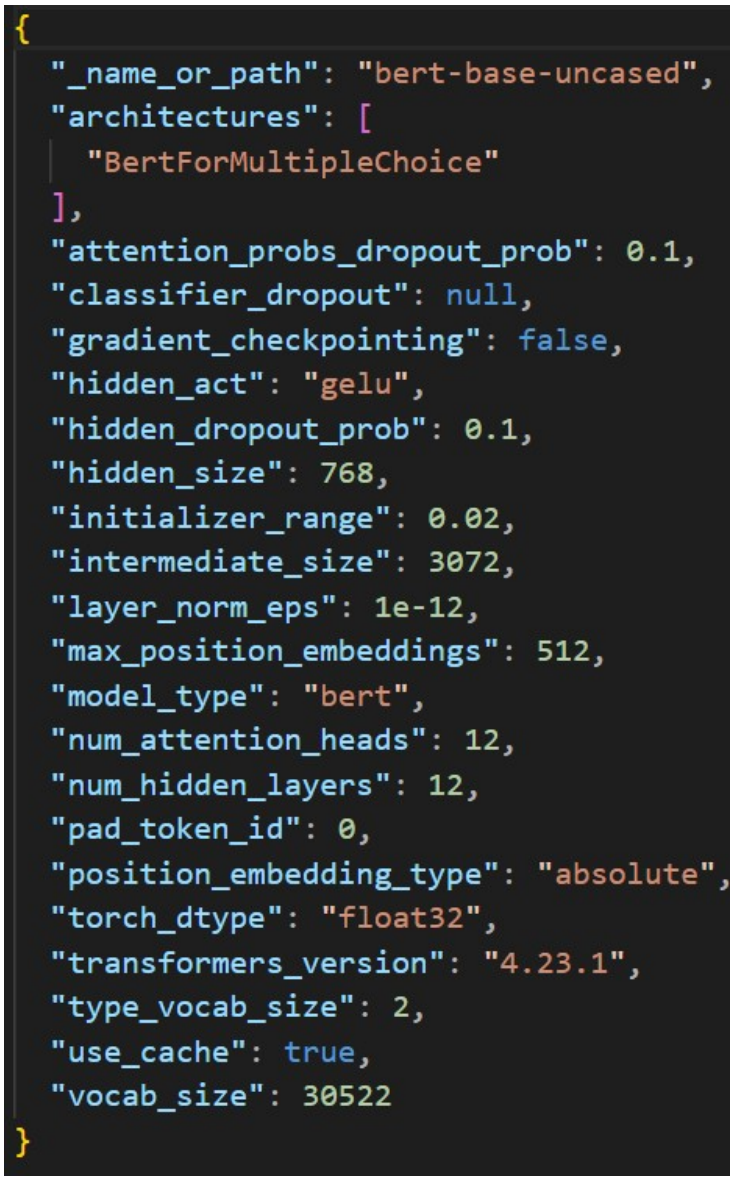
Q3 Curves


Every 400 Step Loss(藍線以右為第二個epoch)
Loss
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
400 1600 2800 4000 5200 6400 7600 8800 10000 11200 12400 13600 14800 16000 17200 18400 19600 20800 400 1600 2800 4000 5200 6400 7600 8800 10000 11200 12400 13600 14800 16000 17200 18400 19600 20800
Step

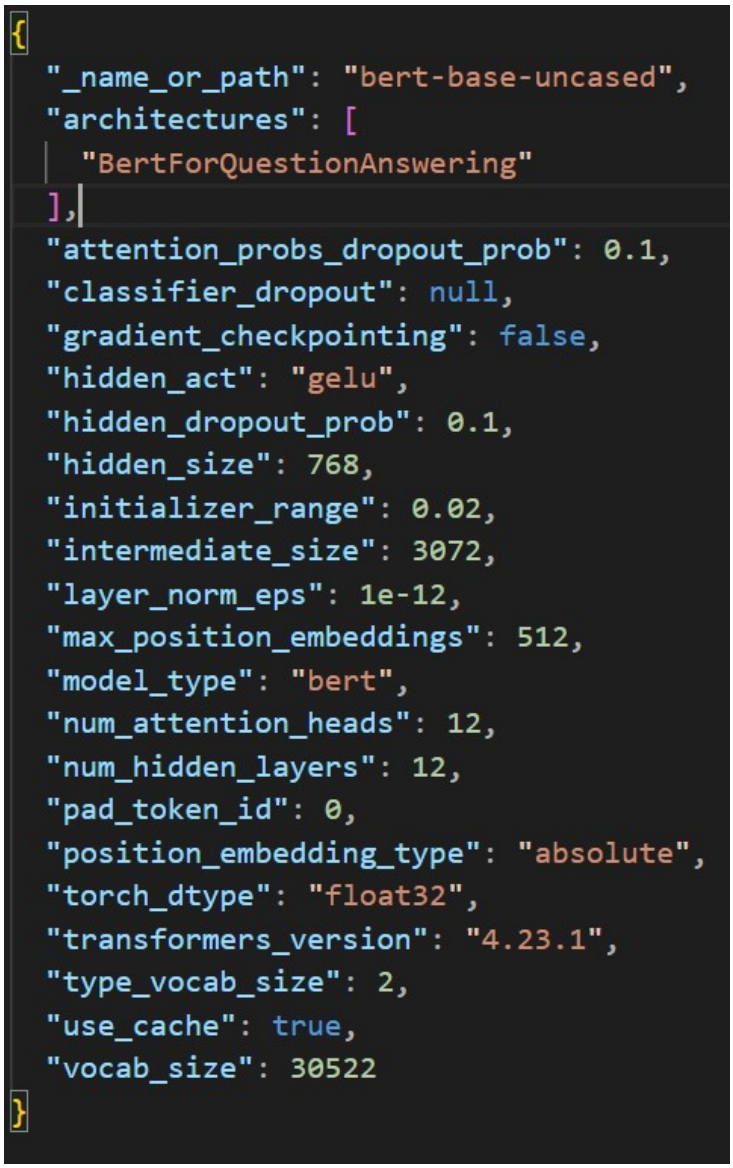

Every 400 Step EM(藍線以右為第二個epoch)
EM
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
400 1600 2800 4000 5200 6400 7600 8800 10000 11200 12400 13600 14800 16000 17200 18400 19600 20800 400 1600 2800 4000 5200 6400 7600 8800 10000 11200 12400 13600 14800 16000 17200 18400 19600 20800
Step

Q4 Pretrained vs Not Pretrained

MultipleChoiceConfig:



QAConfig:



作法: 先使用 init_weight() 函式去除原本的 weight 由於資料量相比 bert 的 pre−train model 少上許多，認為調整方向可調整模型的 configuration，可調整項目包括 hidden_layer, hidden_size 與 num of head 等

performance:
Multiple Choice Accuracy:0.3180
QA Accuracy:0.0458

比較:
相比預訓練好的 BERT 模型，訓練使用的資料量非常少，訓練時間也較不夠，可看出在此架構下直接訓練還是可以訓練到讓模型正確回答少數問題，但若要大幅提升準確率到可實際使用，應需要更大的資料量與時間。