ADL HW2 Report

R10922124 林家毅

Q1: Data processing

1. Tokenizer:

   a. Describe in detail about the tokenization algorithm you use. You need to explain what it does in your own ways.

   ### Answer:

   在 tokenization 的時候是使用 `bert-base-chinese` 這個 pre-trained BERT tokenizer，每當我們對一個 (question, context) pair 作 tokenize 的時候，會把它變成一個整數向量 `input_ids`，裡面的每個 element 都對應某一個 subword 的編號，因此如果我們拿 `input_ids` 去 decode，就可以得到原本的 (question, context) pair 以 “[CLS] question [SEP] context [SEP]” 的方式呈現；另外，在做 tokenization 的同時還會得到 `token_type_ids` 和 `attention_mask` 兩個額外的向量，其中，`token_type_ids` 會用 0 來標示 question 部分，用 1 來標示 context 部分，而 `attention_mask` 則是在有做 padding 的時候將 padding 的部分標記為 0，句子原本的 subwords 部分標記為 1

2. Answer Span:

   a. How did you convert the answer span start/end position on characters to position on tokens after BERT tokenization?

   b. After your model predicts the probability of answer span start/end position, what rules did you apply to determine the final start/end position?

   ### Answer:

   a. 先利用 `sequence_ids`在當前的 feature 的 `input_ids` 中找到這段 context 的 start position 和 end position，然後再利用 `offset_mapping` 對應到原本完整 context tokens 中真正的 indices，這樣就可以跟 answer span 的 start/end index 作對應，進而得到 answer span 在目前 feature 的 start/end position (如果 answer 不在或不完全在這段 context 裡面的話就把 start/end position 都設成 0)

b. 在做 preprocess 的時候，會先存下 `example_ids`，可以用來判斷哪些 start_logits 和 end_logits 是屬於某個 example，然後先選出 `n_best = 20` 個最大的 logits，再用這些 start ≤ end 的組合來判斷哪組 `start_logit + end_logit` 的值最大，最後再利用 `offset_mapping` 對應到原本 context 的 indices，就可以得到 final start/end positions

Q2: Modeling with BERTs and their variants

1. Describe
    a. your model (configuration of the transformer model)
    b. performance of your model.
    c. the loss function you used.
    d. The optimization algorithm (e.g. Adam), learning rate and batch size.

Answer:

- Configuration

{

"_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": "4.17.0",

"type_vocab_size": 2,

"use_cache": true,

"vocab_size": 21128

}

- Public score = 0.73236
- Loss function = cross entropy loss
- Optimization algorithm = “adamw_torch”，learning rate = 3e-5，batch size = 2

2. Try another type of pretrained model and describe
   a. your model
   b. performance of your model
   c. the difference between pretrained model (architecture, pretraining loss, etc.)

Answer:

- Configuration

{

"_name_or_path": "hfl/chinese-roberta-wwm-ext",

"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": 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": "4.17.0",

"type_vocab_size": 2,

"use_cache": true,

"vocab_size": 21128

}

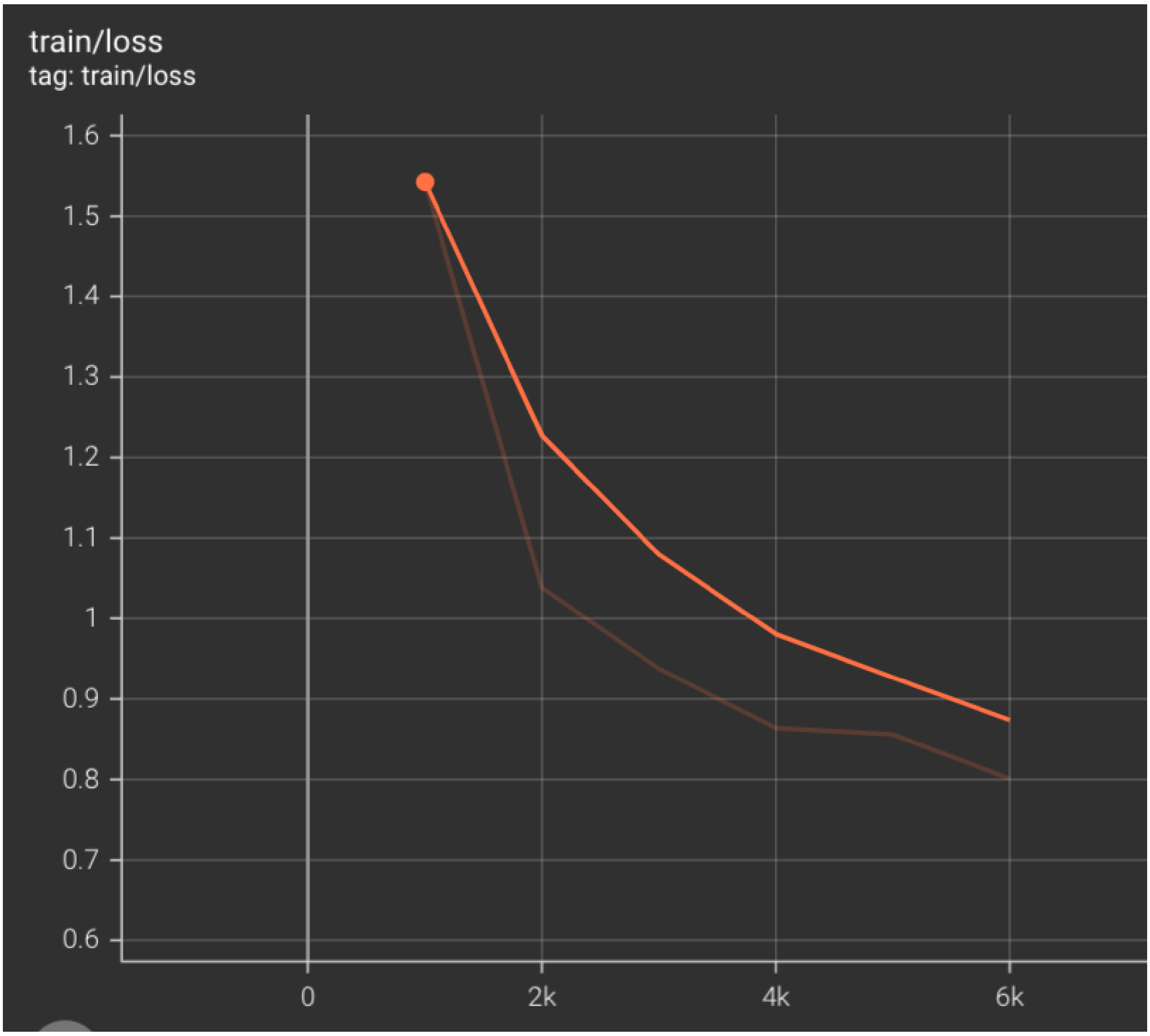
- Public score = 0.77396
- Loss function = cross entropy loss
- Optimization algorithm = “adamw_torch”，learning rate = 3e-5，batch size = 2
- 其中一個跟 bert-base-chinese 不同的地方在於 hfl/chinese-roberta-wwm-ext 做 pretrain 的時候適用到 whole word masking 的技巧，讓機器預測整個 word

Q3: Curves

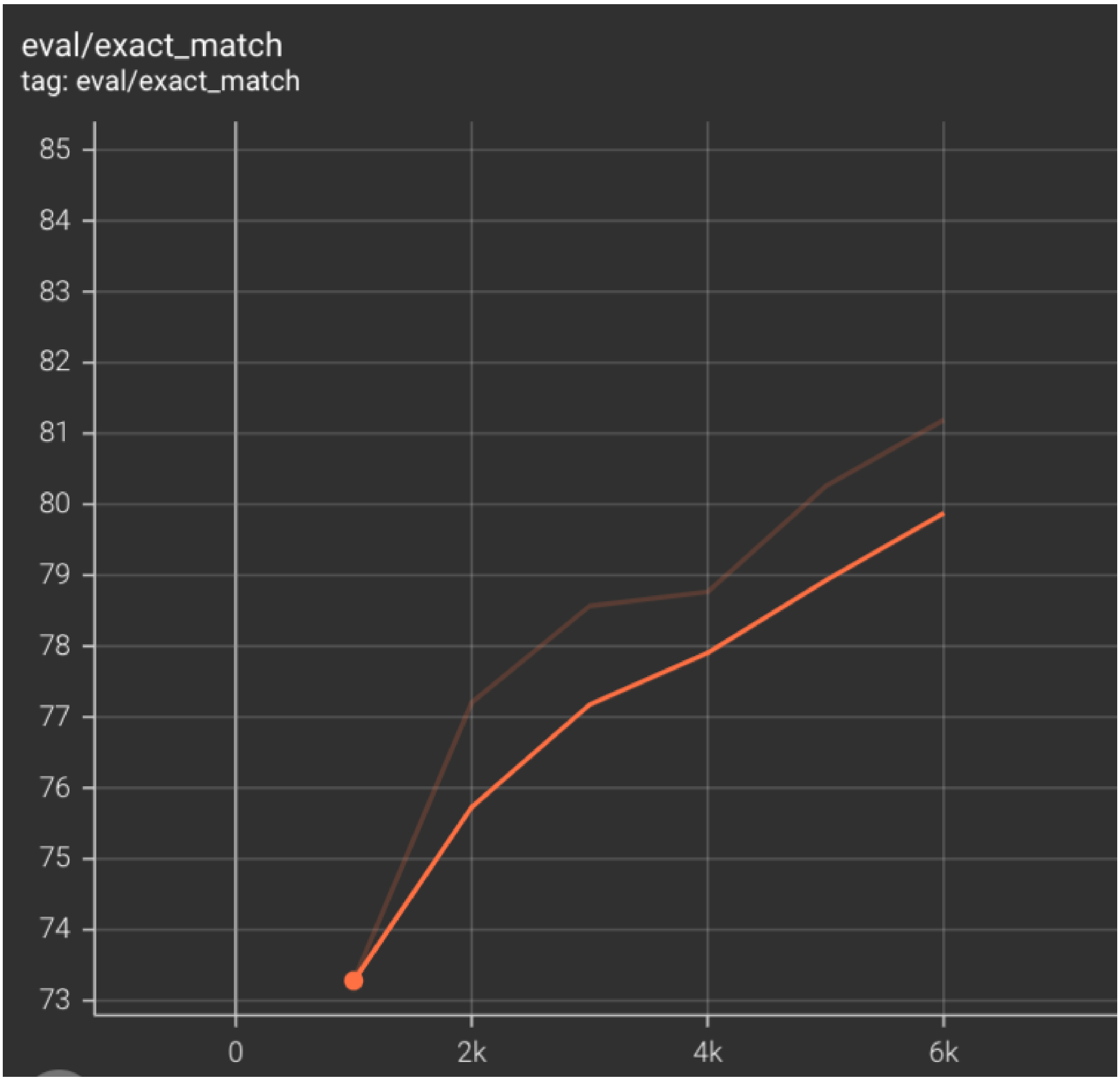
1. Plot the learning curve of your QA model
   a. Learning curve of loss
   b. Learning curve of EM

Answer:

a. Learning curve of loss of "hfl/chinese-roberta-wwm-ext" model (1 point per 1000 steps, total 6 points)



b. Learning curve of EM of "hfl/chinese-roberta-wwm-ext" model (1 point per 1000 steps, total 6 points)



Q4: Pretrained vs Not Pretrained

- Train a transformer model from scratch (without pretrained weights) on the dataset (you can choose either MC or QA)
- Describe
  - The configuration of the model and how do you train this model
  - the performance of this model v.s. BERT

Answer:

- Configuration

{

"architectures": [
"BertForQuestionAnswering"
],
"attention_probs_dropout_prob": 0.1,
"classifier_dropout": null,
"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,
"position_embedding_type": "absolute",
"torch_dtype": "float32",
"transformers_version": "4.17.0",
"type_vocab_size": 2,
"use_cache": true,
"vocab_size": 30522
}

- Public score = 0.04972，相較於 bert-base-chinese 的 0.73236 是非常低的