Q1. Data processing

(1) Tokenizer

使用 Hugging Face 的 Bert Tokenizer。Bert Tokenizer 會將 word 轉成 basic token，然後再把這些 basic token 轉成 word-piece token 用於 sub-word，這些 token 會用 id(integer)記錄下來。Tokenizer 也新增 special token 如[CLS] ,[MASK], [SEP] 到要轉換的句子裡。

(2) Answer Span

- How did you convert the answer span start/end position on characters to position on tokens after BERT tokenization?
  BertTokenizer 的回傳 tokenized_examples 中，有 offset_mapping 的 attribute 可以將 token 轉回成原本 char 的 position 以及 sequence_id 可以得到 sequence。
  train.json 中可以得到 answer 的 start_char position 以及 text 的長度，相加後可得到 end_char position。
  接者初始化 start_token_index 及 end_token_index 在此 sequence 的頭跟尾，一直迴圈到 offset_mapping[start_token_index]為 start_char。此時 start_token 對應到的就是 answer span 中的開始，而用相同的做法也可找到結尾。
- After your model predicts the probability of answer span start/end position, what rules did you apply to determine the final start/end position?
  找到該 example 可能 start logits 與 end logits 記錄前 20 個機率最高的 span，20 個裡面機率最高的就是輸出的 answer span

Q2. Modeling with BERTs and their variants

(1) Describe

- Model
  - Paragraph selection: bert-base-chinese
  - Span selection: hfl/chinese-roberta-wwm-ext
- Performance
  - Paragraph selection Accuracy : 0.9557
  - Span selection EM: 81.887
- Loss function
  - Paragraph selection: Cross-Entropy
  - Span selection: Cross-Entropy
- Optimizer
  - Paragraph selection: AdamW
  - Span selection: AdamW

- Learning Rate
  - Paragraph selection: 3e-5
  - Span selection: 3e-5
- Batch Size
  - Paragraph selection: 2
  - Span selection: 2

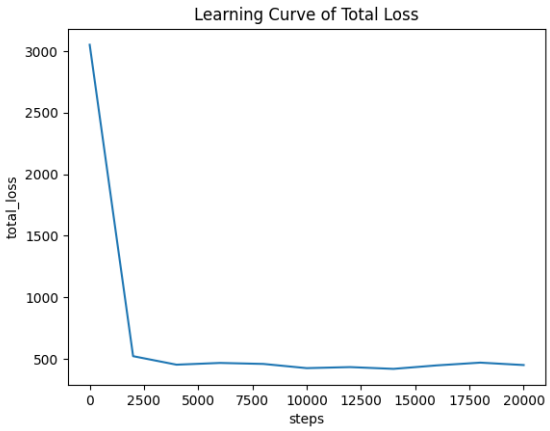
(2) Another Type(Span Selection)

- Model: hfl/chinese-bert-wwm-ext
- Peformance EM: 78.531
-

Q3. Curves

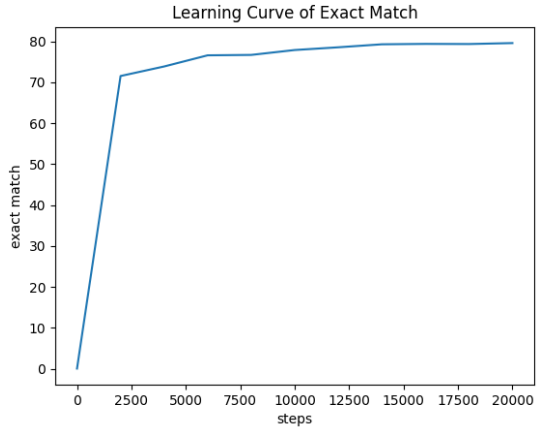
(1) Loss Curve

每 2000 步紀錄一次 validation 的 total loss



(2) EM Curve

每 2000 步紀錄一次 validation 的 Exact Match



Q4. Pre-trained vs Not Pre-trained