R10943181 電子所碩一 黃詩瑜

1. After your model predicts the probability of answer span start/end position, what rules did you apply to determine the final start/end position? (the rules you applied must be different from the sample code)

Ans:

當預測出的end\_index 小於 start\_index時，會得不到答案。在evaluate funciton，加入一個條件，只有在start\_index 小於等於 end\_index時，且機率更大時，才會更新答案。

2. Try another type of pretrained model which can be found in huggingface’s Model Hub (e.g. BERT -> BERT-wwm-ext, or BERT -> RoBERTa ), and describe

● the pretrained model you used

● performance of the pretrained model you used

● the difference between BERT and the pretrained model you used (architecture,

pretraining loss, etc.)

Ans:

the pretrained model you used：hfl/chinese-macbert-large

performance of the pretrained model you used：

epoch = 4

validation accuracy = 0.790

Kaggle public score = 0.80798

the difference between BERT and the pretrained model you used：

BERT’s architecture:

BERT主要包括兩個pre-training tasks：mask語言模型(MLM)和下一句預測(NSP)。

MLM：從輸入中隨機masked某些tokens，目的是僅根據其上下文預測原始單詞。

NSP：預測句子B是否為A的下一個句子 。

hfl/chinese-macbert-large’s architecture:

MacBERT保留了BERT的一些pre-training tasks，再做些修改。改使用Whole Word Masking、N-gram Masking：single token、2-gram、3-gram、4-gram，分別對應比例為0.4、0.3、0.2、0.1。因為finetuning時從未見過[MASK]token，因此使用相似的word進行替換，使用工具Synonyms toolkit 獲得相似的詞。如果被選中的N-gram存在相似的詞，則隨機選擇相似的詞進行替換，否則隨機選擇任意詞替換。對於一個輸入文本，15%的詞進行masking。其中80%的使用相似的詞進行替換，10%使用完全隨機替換，10%保持不變。

BERT’s pretraining loss：

epoch = 4

validation accuracy = 0.746

Kaggle public score = 0.76079

hfl/chinese-macbert-large’s pretraining loss：

epoch = 4

validation accuracy = 0.790

Kaggle public score = 0.80798