Natural Language Processing Final Project Report

Group 14

B07902037 蔡沛勳　B07902115 陳致元　R10922105 鄧遠祥

Task 1: Aspect Category Detection

1. Method

Preprocess

每筆 review 有18個 aspect, 所以會有18個 label。依照 task 的需求, 將 label -2, -1, 0, 1 修改為兩個 label : 0, 1, 將 -2 修改為 0, 代表沒有評論到此 aspect, 將 -1, 0, 1 修改為 1, 代表有評論到此 aspect。

將 review 用 tokenizer編碼, 若長度超過512會截斷, 不夠則做 padding，即為餵入 model 的 input。

Pretrained model

皆使用 huggingface 上的 pretrained model 和 tokenizer 來 fine-tuning。

使用bert-base-chinese和hfl/chinese-roberta-wwm-ext。

Model architecture


```
"_name_or_path": "hfl/chinese-roberta-wwm-ext",
"architectures": ["BertForMaskedLM"],
"attention_probs_dropout_prob": 0.1,
"bos_token_id": 2,
"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_heada": 12,
"pooler_num_fc_layers": 3,
"pooler_size_per_head": 128,
```

```
"pooler_type": "first_token_transform",
"position_embedding_type": "absolute",
"problem_type": "multi_label_classification",
"transformer_version": "4.19.2",
"type_vocab_size": 2,
"use_cache": true,
"vocab_size": 21128,
```


2. Experiment

Environment setting

GPU: GTX 1080 Ti

Library:

- accelerate
- datasets >= 1.8.0
- sentencepiece != 0.1.92
- scipy
- scikit-learn
- protobuftorch >= 1.3

Hyperparameters

optimizer: AdamW

learning rate = 3e-5

batch size = 8

Gradient accumulation steps = 2

Result

Bert v.s. Roberta

Model	Bert	Roberta
Kaggle Score	0.74210	0.79813

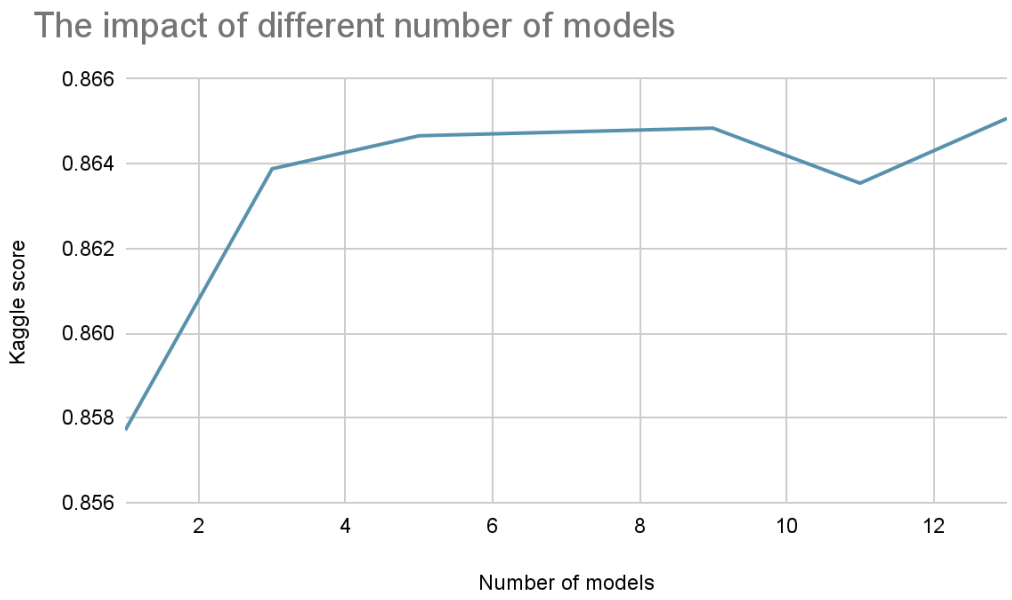
Max length 384 v.s. 512

Max Length	384	512
Kaggle Score	0.79813	0.85823

Single Model v.s. Ensemble (different random seed)

Random Seed	seed1	seed2	seed3	ensemble
Kaggle Score	0.85444	0.85771	0.86391	0.86388

Different number of models



Task 2: Aspect Category Sentiment Classification

1. Method

我們測試了三種不同的 pretrained model 及三種不同的 model architecture 來得到所有分類的 sentiment classification。

Pretrained Model:

皆使用 huggingface 上的 pretrained model 和 tokenizer 來 fine-tuning。

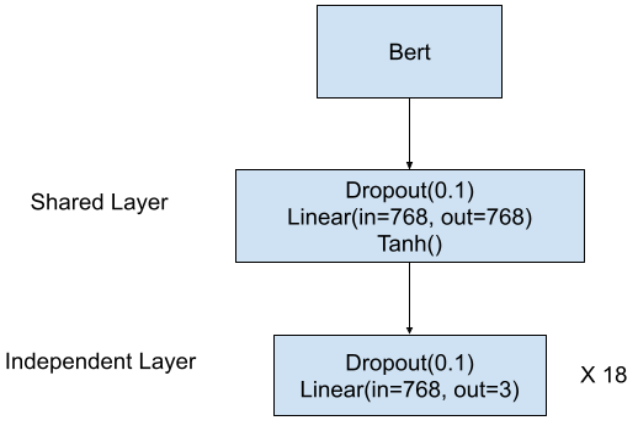
model	Huggingface link	hidden_layers	Output size
BERT	bert-base-chinese	12	(768)
XLNet	hfl/chinese-xlnet-base	12	(input_size * 768)
Roberta	hfl/chinese-roberta-wwm-ext	12	(input_size * 768)

由於 XLNet 及 Roberta 對每個 input 產生的 representation 為 input size*768，故取 mean 使得所有 representation 維度皆為 (768)。

Model Structure:

以下皆以 BERT 作範例，可替換為 XLNet, Roberta。

(1) 訓練單一 model 對所有 labels 進行 sentiment classification



(2) 對所有 labels 皆訓練一個 model 來做 sentiment classification

對於每個 label 的 train 和 dev 資料, 我們先將值為 -2 的 sample 移除以節省訓練 model 的時間:

- 原本 train 共有 33647 筆 sample, 但經過前述操作後平均剩下 10804 筆 sample
- 原本 dev 共有 4265 筆 sample, 但經過前述操作後平均剩下 1394 筆 sample

Model 結構與第一個類似, 差別為 num_label 為 1, 總共訓練18個 model。

(3) 將所有 labels 分成五群, 各自訓練 model 來做 sentiment classification

根據每個 label 的 prefix (Ambience, Food, Location, Price Service) 分群, 各自訓練一個 model。

同樣地, 對於每個 label 的 train 和 dev 資料, 我們先將值為 -2 的 sample 移除以節省訓練 model 的時間:

- 原本 train 共有 33647 筆 sample, 但經過前述操作後平均剩下 22267 筆 sample
- 原本 dev 共有 4265 筆 sample, 但經過前述操作後平均剩下 2866 筆 sample

Model 結構與第一個類似, 差別為 num_label 為不同大類的 label 數, 總共訓練 5 個 model。

Loss Function: CrossEntropyLoss(ignore_index = -1)

忽略原先 dataset 標記為 -2 (the aspect is not mentioned in the text) 的 data

Optimizer: ADAM, learning rate = 2e-5, batch size = 8

2. Experiment

以下實驗設定:

- GPU: RTX 3070
- Model: 如 (1) 中提到的 model structure, 將 pretrained model 替換成 BERT, Roberta 及 XLNet
- 每筆 input padding 或 truncate 至長度 512
- 訓練跑 3 個 epoch、batch size 8

a. 使用不同的 Pretrain Model (Bert v.s. Roberta v.s. XLNet)
*這邊使用的是 evaluation accuracy

Categories	BERT		Roberta		XLNet	
	Time	Accuracy	Time	Accuracy	Time	Accuracy
All	0:53:05	0.83	1:07:22	0.84	1:36:06	0.8246
Ambience	0:36:08	0.8790	0:36:36	0.8840	1:04:01	0.8812
Food	1:00:38	0.8237	0:55:49	0.8264	1:37:25	0.8377
Location	0:26:39	0.9309	0:23:51	0.9256	0:41:51	0.9268
Price	0:41:20	0.7994	0:39:12	0.7931	1:09:19	0.8028
Service	0:47:16	0.8217	0:40:31	0.8358	1:11:36	0.8336
Total	3:32:01	0.8509	3:15:59	0.8464	5:44:06	0.8506

b. 單一 model v.s. 分群訓練 v.s. 每個 label 各自訓練
*這邊使用的 pretrained model 為 BERT

Method	Total Time	Average Accuracy	Kaggle (public)
單一 model	0:53:05	0.8353	0.83152
分五大類的 label	3:32:01	0.8509	0.84499
1-to-1 model	7:28:23	0.8443	0.84190

c. Results:

從實驗 a 可以看出使用不同 pretrained model 在訓練時間及正確率上的表現及差異。

從訓練時間來看, XLNet 的訓練時間為 BERT 及 Roberta 的 1.5 倍以上, 而 BERT 在只訓練一個 model 分類時訓練速度比 Roberta 快, 但在分五大類處理時則反過來。

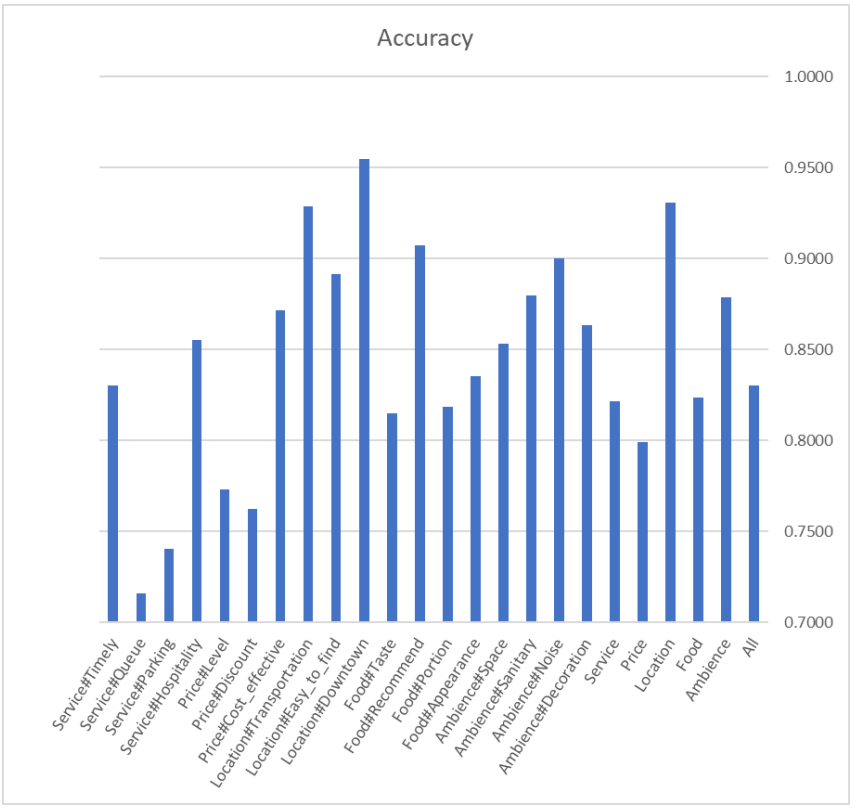
而正確率上訓練單一 model 分類時 Roberta 的正確率最高, 有 84 %的正確率, 但是在分五大類時 BERT 的正確率最高, 有 85.09% 的正確率。

從實驗 b 可看出 3 種 model structure 的訓練時間及正確率。

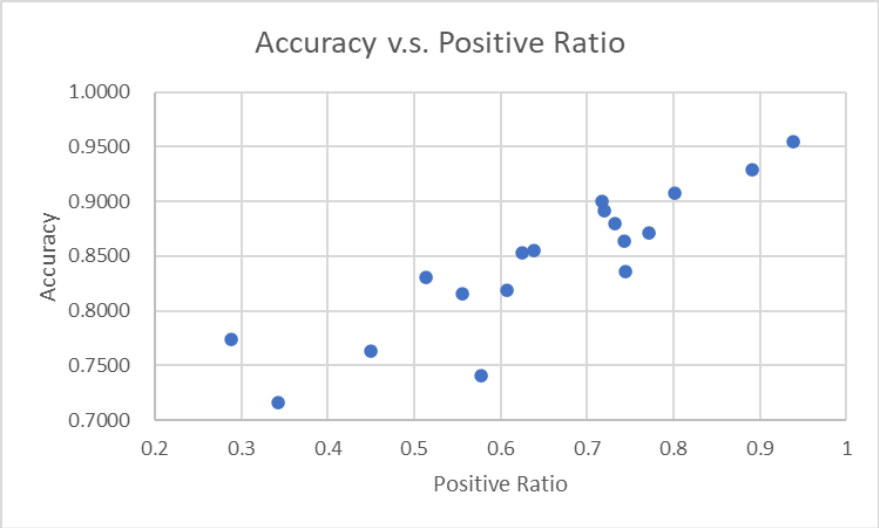
訓練時間上，由於要訓練的 model 數分別為1/5/18，即使能透過 preprocess 先砍掉部分資料，三者的訓練時間的差異依然是巨大的，訓練時間比約為 1 : 4 : 10.5。

而正確率則為分五大類的 model 效果最好 (84.5 %)；1-to-1 model 次之 (84.2 %)，單一 model 最差 (83.2 %)。

d. 圖表



圖一、category、分群等 eval accuracy 的並排比較。這裡沒觀察到比較明顯的規律。



圖二、Category 各自訓練時，Accuracy 和 label=1 的比例之間的關係。我們可以從圖中觀察到兩者似乎有強烈的正相關。

Workload Distribution

蔡沛勳	Task 2, report
陳致元	Task 2, report
鄧遠祥	Task 1, report