程式版本

使用 Simple Transformers 框架

Simple Transformers 是一個 Python 的 NLP 套件/框架，旨在減少使用 Transformer 模型時的複雜步驟。能夠讓使用者透過短短幾行的程式碼，快速實現一個 NLP 任務的深度學習模型訓練環境。

而它的簡化工作，顧名思義，當然是基於 Hugging Face 團隊和他們的 Transformers 套件。

安裝 simpletransformers

```
!pip install simpletransformers
```

```
import pandas as pd
```

```
def Convert(data_p):
    data_p = data_p[data_p[’sentiment’].isin([’positive’, ’negative’])]
    data_p[’sentiment’] = data_p[’sentiment’].replace({’positive’:1, ’negative’: 0})
    return data_p
```

```
s_df = pd.read_csv(’/content/drive/MyDrive/Colab Notebooks/IMDB_Dataset.csv’)
```

將資料集 IMDB_Dataset.csv 的 'positive' 改為'1', 'negative'改為 '0'

```
s_df = Convert(s_df)
```

查看資料狀況(正面1,負面0)

```
s_df[’sentiment’].value_counts().plot(kind=’bar’)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f2ac77e0310>
```

詳細資料集狀況

Simple Transformers要求資料必須包含在至少兩列的Pandas DataFrames中。

只需為列的文字和標籤命名，SimpleTransformers就會處理資料。

第一列包含文字，型別為str。 第二列包含標籤，型別為int。

---- | |

```
s_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 50000 entries, 0 to 49999
Data columns (total 2 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   review     50000 non-null  object
 1   sentiment  50000 non-null  int64 
dtypes: int64(1), object(1)
memory usage: 1.1+ MB
```

```
s_df
```

	review	sentiment
0	One of the other reviewers has mentioned that ...	1
1	A wonderful little production. <br /><br />The...	1
2	I thought this was a wonderful way to spend ti...	1
3	Basically there's a family where a little boy ...	0
4	Petter Mattei's "Love in the Time of Money" is...	1
...	...	...
49995	I thought this movie did a down right good job...	1
49996	Bad plot, bad dialogue, bad acting, idiotic di...	0
49997	I am a Catholic taught in parochial elementary...	0
49998	I'm going to have to disagree with the previou...	0
49999	No one expects the Star Trek movies to be high...	0

50000 rows × 2 columns

```
from sklearn.model_selection import train_test_split
```

將資料集拆分為

訓練集train_df 及 測試集test_df

```
train_df, test_df = train_test_split(s_df, test_size = 0.2, random_state = 1027)
```

導入模組(導入想要使用的模型，使用BERT模型的實現套件simpletransformers)

```
from simpletransformers.classification import ClassificationModel
```

創建 ClassificationModel

```
model = ClassificationModel('bert', 'bert-base-uncased',num_labels=None, weight=None)
# model = ClassificationModel("bert","bert-base-uncased")
```

```
Downloading: 100%                                    570/570 [00:00<00:00, 17.1kB/s]

Downloading: 100%                                    420M/420M [00:08<00:00, 56.5MB/s]

Some weights of the model checkpoint at bert-base-uncased were not used when initial
- This IS expected if you are initializing BertForSequenceClassification from the ch
- This IS NOT expected if you are initializing BertForSequenceClassification from th
Some weights of BertForSequenceClassification were not initialized from the model ch
You should probably TRAIN this model on a down-stream task to be able to use it for
Downloading: 100%                                    28.0/28.0 [00:00<00:00, 325B/s]

Downloading: 100%                                    226k/226k [00:00<00:00, 5.65MB/s]

Downloading: 100%                                    455k/455k [00:00<00:00, 8.71MB/s]
```

訓練模組

由於GPU資源限制,Epoch採預設值1

```
model.train_model(train_df, args = {'overwrite_output_dir': True})
```

```
/usr/local/lib/python3.7/dist-packages/simpletransformers/classification/classificat
  "Dataframe headers not specified. Falling back to using column 0 as text and colum

0%                                 80/40000 [00:29<3:51:13, 2.88it/s]

/usr/local/lib/python3.7/dist-packages/transformers/optimization.py:309: FutureWarni
  FutureWarning,

Epoch 1 of 1: 100%                                   1/1 [09:21<00:00, 561.73s/it]

Epochs 0/1. Running Loss: 0.0210: 100%                              5000/5000 [09:1

(5000, 0.3673441009521484)
```

選擇一組好的超參數(hyperparameter)值在開發最先進的模型中起著巨大的作用。

Simple Transformers 原生支持出色的 **W&B Sweeps**(https://docs.wandb.ai/guides/sweeps) 功能，以實現自動超參數優化。

訓練完成後,測試資料集(test_df)放進去測試

```
result, model_output, wrong_predictions = model.eval_model(test_df)
```

```
/usr/local/lib/python3.7/dist-packages/simpletransformers/classification/classificat
  "Dataframe headers not specified. Falling back to using column 0 as text and colum
0%                                                  20/10000 [00:07<56:36, 2.94it/s]

Running Evaluation: 100%                                   1250/1250 [00:28<00:00, 43.75
```

測試結果

```
result
```

```
{'mcc': 0.7655034223410263,
 'tp': 4502,
 'tn': 4323,
 'fp': 678,
 'fn': 497,
 'auroc': 0.9551491182059647,
 'auprc': 0.9534837785696679,
 'eval_loss': 0.3237473171234131}
```

整體預測準確率 (tp+tn)/tp+fp+fn+tn)

```
(4502+4323)/(4502+678+497+4323)
```

```
0.8825
```

✓ 0 秒 完成時間：下午5:40 ● ×