

電子商務技術-作業2

110423078何仲霖 資管碩一

Colab連結：

TextGeneration :<https://colab.research.google.com/drive/1ihekj5STEcMFI9LrYHFy8sNbcQFucBir?authuser=1#scrollTo=QGp-TR08Jzps>
StockENN : <https://colab.research.google.com/drive/148X9I6GkU3J39fNzCxBrSvU8MKvnLns?usp=sharing>

一.TextGeneration

- 選擇文章

這個作業使用「哈利波特1-神秘的魔法石」作為練習

- 文章經過訓練後的判別差異

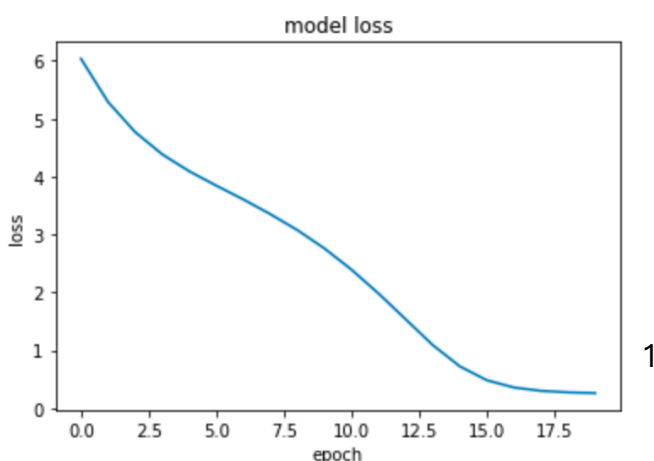
一 訓練前：

```
原本的中文字序列：
，大喊大叫，連帶跳往後退去。
哈利坐起來
-----
輸入尚未訓練的model後獲得：
冰冰冰冰讀讀讀讀讀讀讀讀妙妙答佛父佛佛
```

一 訓練後：

```
原本的中文字序列：
，大喊大叫，連帶跳往後退去。
哈利坐起來
-----
輸入進訓練後的model後獲得：
「喊大叫，連帶跳往後退去。
哈利坐起來，
```

- 評估模型



- 預測文章結果

```
init_seq = "哈利波特"
init_seq_ind = [word_2_index[w] for w in init_seq]
input = init_seq_ind[-seq_len:]

generateWords(input,500)
```

哈利波特！」

他的心突然往下一沉，比他剛才出來了，他的生日裡還有一個再躲。」

「我就把你留在這兒吧，」哈利說，「我們必須充當棋子。」

「我不知道，先生。」

「我不知道它叫這個名字，先生。」海格說。

「我不是弗雷，我不知道你是誰了！」

「我想，我們一定離來了。」

「不管怎麼說……哈利，」巨蟒說，「你們可以肯定一定是學校裡的學院。我們需要學習的東西太多，就把隱形斗篷送給你的。」

「我……不想……穿……」

「別大聲嚷嚷，」他說，「我們必須充當棋子。」

「我不知道，先生。」

「我不知道它叫這個名字，先生。」海格說著，舉起望遠鏡，巨蟒從他們身邊走過去。

「好，」海格說，「裏邊是魁地奇球的是爆，」他們把隱形斗篷拉在身上。

「我們還是回家去比好，」他對著一隻子往前，佩妮阿姨說服達力坐到哈利身邊，想看這裡就是個盼望。

哈利覺得自己臉上頓時失去了血跡。

「……他還是有一個秘密。」

「我們只能猜測，」哈利說，「幸吧，我們現在必須這麼做，」鄧不利多說，「我們不知道這個洞有多深。把笛子給你們，想要在霍格華茲魔法學校就讀。隨信很強，我們需要一個人，這是一個好。」

「哦，」男孩說，「我聽說過這個洞有不同的動物，可是你們在

二. StockRNN

- 參數設定
 - Seq_len設為3
 - BATCH SIZE設為10
- 模型設計

```
# todo
input_shape = (seq_len,1)
output_shape = (seq_len,1)

model = tf.keras.Sequential()

model.add(tf.keras.layers.LSTM(units=50, return_sequences=True, input_shape = input_shape))
model.add(Dropout(0.2))

model.add(tf.keras.layers.LSTM(units=50, return_sequences=True))
model.add(Dropout(0.2))

model.add(tf.keras.layers.Dense(1))

model.summary()
```

Output為一個數值，所以Output的維度為1

- 制定訓練計畫並訓練

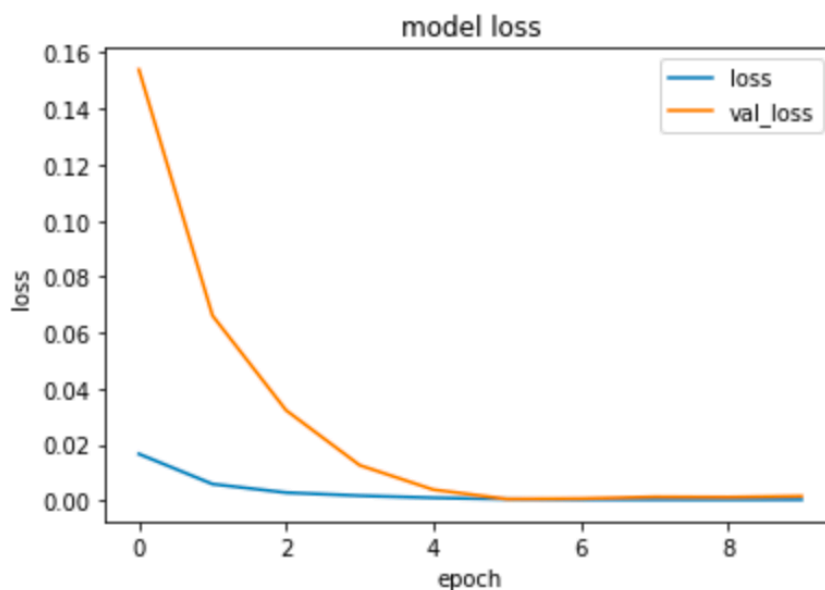
```
# todo
epochs = 10

# todo
model.compile(loss="mean_squared_error",optimizer="adam",metrics=["accuracy"])

# todo
# 前面使用 tf.data 建構的資料集
history = model.fit(train_ds, epochs = epochs, validation_data = val_ds, batch_size=BATCH_SIZE)
print(history.history.keys())
```

Loss function 採用MSE

- 評估模型



- 進行預測

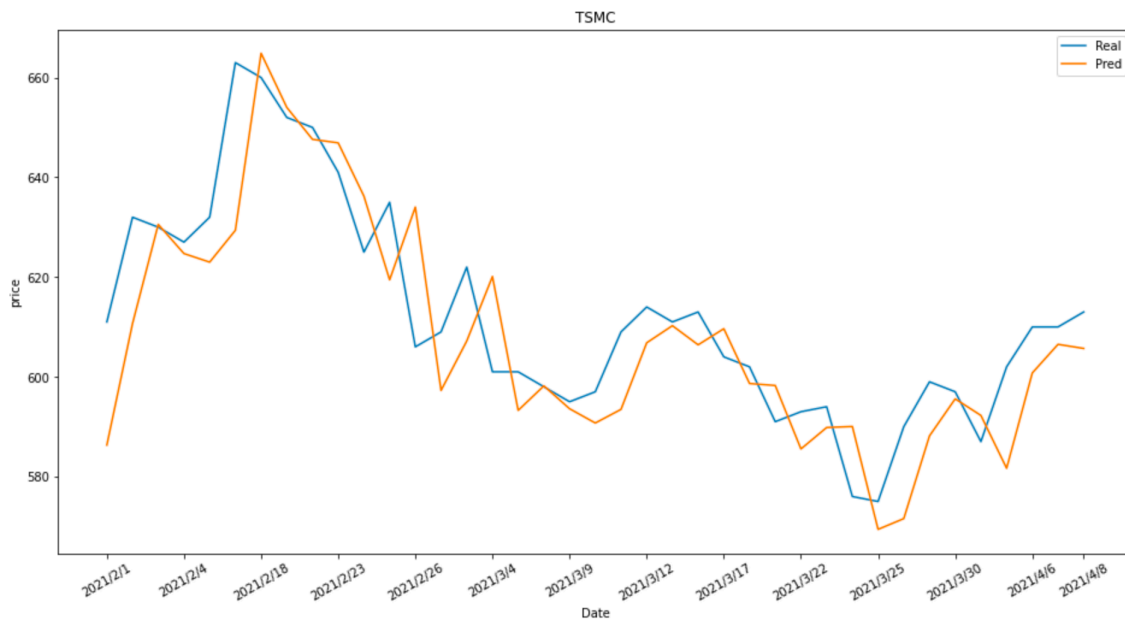
我們要透過seq_len的天數，來預測seq+1天的股價

```
# 參考步驟（在for迴圈中）：
# 1. 把input轉**維度**成一個新變數next_input丟入model
# 2. 獲得的pred取出seq中最後一個時間點的output數值（提示：[0,-1,0]）
# 3. 把pred丟入pred_prices中
# 4. 把real_price經過transform_one轉換成小數點並加在input後面
# 5. 保持input長度為seq_len長（也就是把最前面的數字踢掉）
#####
# todo #
#####
predict = model(tf.expand_dims(input,axis=0))
predict = predict[:,-1,:]
pred_prices.append(predict.numpy()[0][0])
output = scalar.transform_one(real_price)
input.append(output)
input=input[-seq_len:]
```

```
print(f"dates : {dates[:10]}")
print(f"real_prices : {real_prices[:10]}")
print(f"pred_prices : {pred_prices[:10]}")
```

```
dates : ['2021/2/1', '2021/2/2', '2021/2/3', '2021/2/4', '2021/2/5', '2021/2/17', '2021/2/18', '2021/2/19', '2021/2/22', '2021/2/23']
real_prices : [611.0, 632.0, 630.0, 627.0, 632.0, 663.0, 660.0, 652.0, 650.0, 641.0]
pred_prices : [0.8551973, 0.89592737, 0.9291046, 0.91934013, 0.9164719, 0.9271335, 0.9864275, 0.9682866, 0.9575881, 0.95641553]
```

● 預測結果



可以看到這個模型的預測值與實際股價算蠻接近，且整體趨勢都是相同的，代表這個訓練模型如果加上其他因素近來考量的話，是可以拿來當作未來股價的參考的。

● 作業心得：

這次的作業主要著重在模型的設計上，因為RNN跟LSTM模型本身比較複雜，花了比較多時間在思考維度的設計及模型的架構，對於這兩個模型的實際運作有更深入的了解。這次的作業將兩個模型應用在比較實務上面，像是股價預測的這個作業，雖然股價的影響因素不單純只是看前幾天的股價趨勢就能預測，還有很多不確定的因素，但是這還是可以作為一個參考的依據，未來甚至可以考慮更多變數，因為現在很多人在做程式交易，除了可以做到自動化交易外，我認為還是有機會可以做到預測股市的趨勢，不追求一定要準確預測股價值，這樣可以給使用者多點參考的資訊，所以這次作業我覺得蠻有趣的，也學到這些模型實際的運用。