

1.使用model:

右圖是我使用的model，是keras的cnn，是一層Convolution(64,4,4)，activation為relu，加上三層dense，dense的out_dim分別為256 256 2，前面兩層activation為relu，最後一層為softmax。

我嘗試過Convolution兩層三層四層，效果都不是很好。也有嘗試過convolution加maxpooling交錯各兩層三層，效果不彰。

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 18, 37, 64)	1088
Flatten_1 (Flatten)	(None, 42624)	0
dense_1 (Dense) (Flatten_1)	(None, 256)	10912000
activation_1 (Activation)	(None, 256)	0
dense_2 (Dense) (Activation_1)	(None, 256)	65792
activation_2 (Activation)	(None, 256)	0
dense_3 (Dense) (Activation_2)	(None, 2)	514
activation_3 (Activation)	(None, 2)	0

2.其他細節

首先，我有調class_weight，是使用compute_class_weight所產生的。

再來是因為過了softmax之後，predict出來的值會是該筆資料為0 or 1的概率，我有設threshold，這個threshold為training data中label為1的資料，經過model.predict後為1的機

率，取最小的那個，並除以50，只要testing data predict出來的機率大於這個機率，我們就判定他為1，否則為0。

除以50是調出來的，有試過不除，除以10除以100，最好的是除以50。

```
class_weight = class_weight.compute_class_weight('balanced',
                                                  np.unique(y_train),
                                                  y_train)
```

```
label_array=np.where(y_train==1)
thresh=min(model.predict(x_train)[label_array][1])/50
```

3.不足

這個model對三份的predict結果都可以蠻高的（recall precision都0.8 up），但偶爾會突然跑出一個偏低的（recall precision都突然降至0.4），所以我還在繼續調參數跟試不同的model。

4.model

```
model.add(Convolution2D(64,4,4, input_shape=(x_train.shape[1:]), activation='relu'))
model.add(Flatten())
model.add(Dense(output_dim=256))
model.add(Activation('relu'))
model.add(Dense(output_dim=256))
model.add(Activation('relu'))
model.add(Dense(output_dim=2))
model.add(Activation('softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=[recall, precision])
model.summary()
model.fit(x_train, y_train, nb_epoch=5, batch_size=100, class_weight = class_weight)
```