

ADLxMLDS2017 HW1 report

姓名：徐有慶

學號：R05922162

1. Model description

由 fbank 的資料集抽出 feature 進行訓練，將所有句子都 padding 到和最長的句子同樣長度，若該 frame 是經過 padding 所產生的，則將其 label 標示為 0，故總共需分為 49 類。

RNN:

```
Bidirectional(LSTM(256, activation='tanh', dropout=0.5, return_sequences=True))
```

```
Bidirectional(LSTM(256, activation='tanh', dropout=0.5, return_sequences=True))
```

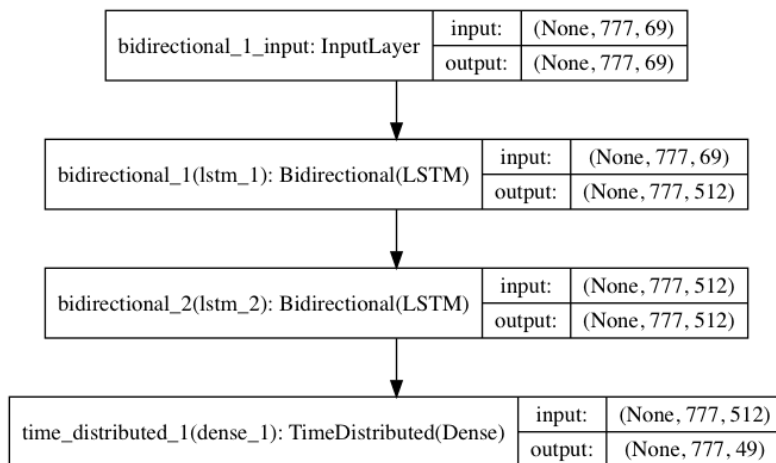
```
TimeDistributed(Dense(49, activation='softmax'))
```

batch size: 128

epoch: 50

loss='categorical_crossentropy'

optimizer='adam'



RNN + CNN:

```
Conv2D(filters=10, kernel_size=[5, 5], activation='tanh', padding='same')
```

```
Conv2D(filters=15, kernel_size=[5, 5], activation='tanh', padding='same')
```

```
Reshape((timesteps, -1))
```

```
Bidirectional(LSTM(256, activation='tanh', dropout=0.5, return_sequences=True))
```

```
Bidirectional(LSTM(256, activation='tanh', dropout=0.5, return_sequences=True))
```

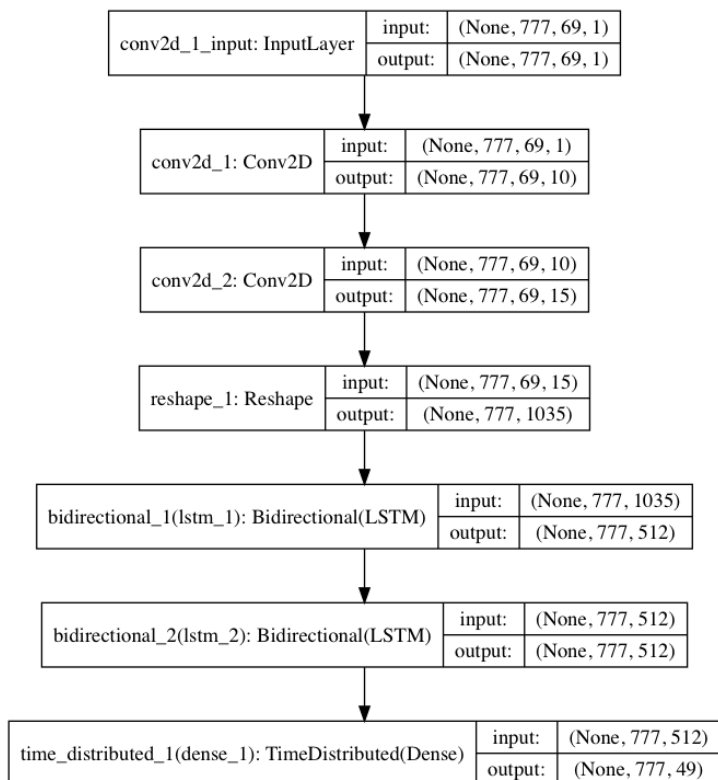
```
TimeDistributed(Dense(49, activation='softmax'))
```

batch size: 128

epoch: 15

loss='categorical_crossentropy'

optimizer='adam'



2. How to improve my performance

Early stopping:

總共有 3696 筆 training data，切出 100 筆當作 validation set，剩餘 3596 筆拿去訓練，觀察 validation 所算出的 loss，若沒有再下降則停止，以防止 overfitting。

Feature normalization:

將每一筆 training data 的 feature 正規化到平均值為 0，標準差為 1，觀察 fbank 的 feature 可以發現，其實每個維度的值，彼此之間差異蠻大的，為了防止有些維度影響較大，做了正規化後可以使得每個維度的影響較平均。

Change wrong prediction:

預測結果時，檢查第 k-1 個 frame 和第 k+1 個 frame 所預測的結果是否相同，如果相同，則第 k 個 frame 的結果也應該要相同，若第 k 個 frame 的結果不相同，則將其改為和第 k-1 個 frame 的值，例：bb**k**bbb -> bb**b**bbb。因為所預測的東西是一句話，所以相同的 phone 應該要是連續的，不可能會出現前後的 phone 相同但中間的 phone 不同。

3. Experimental results and settings

Kaggle private set score:

RNN: 18.02259

CNN+RNN: 10.8531

兩種不同的 model，其所使用的 RNN 架構都是相同的，但是由 kaggle 上的分數可以發現，若在 RNN 前先使用 CNN 去結合不同 frame 之間的 feature，出來的結果會比單純用 RNN 還要好。只使用 RNN，每個 frame 的 feature 只會有 69 維，先用 CNN 去做結合後，每個 frame 的 feature 會成長到 1035 維。

4. Others

使用套件: keras(2.0.7), backend: tensorflow(r1.3.0)