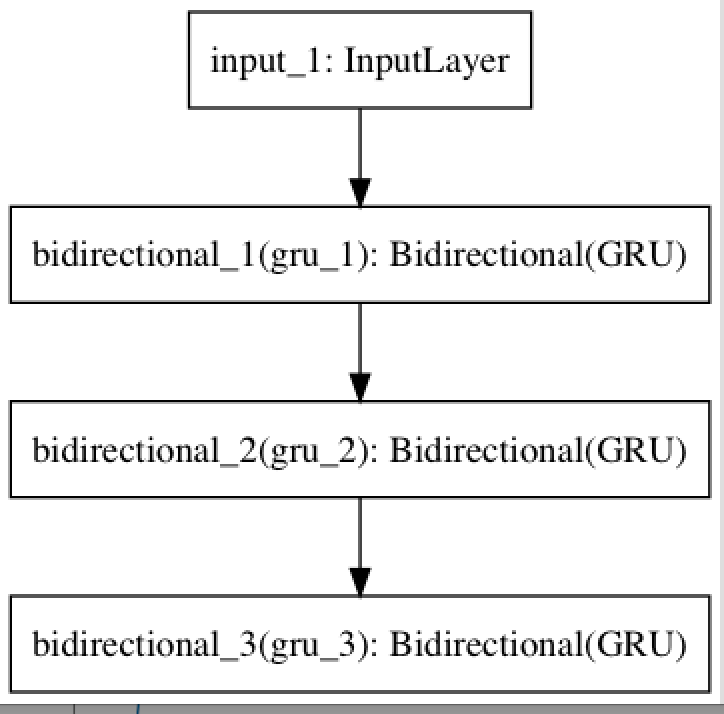
R05922131 祝子軒

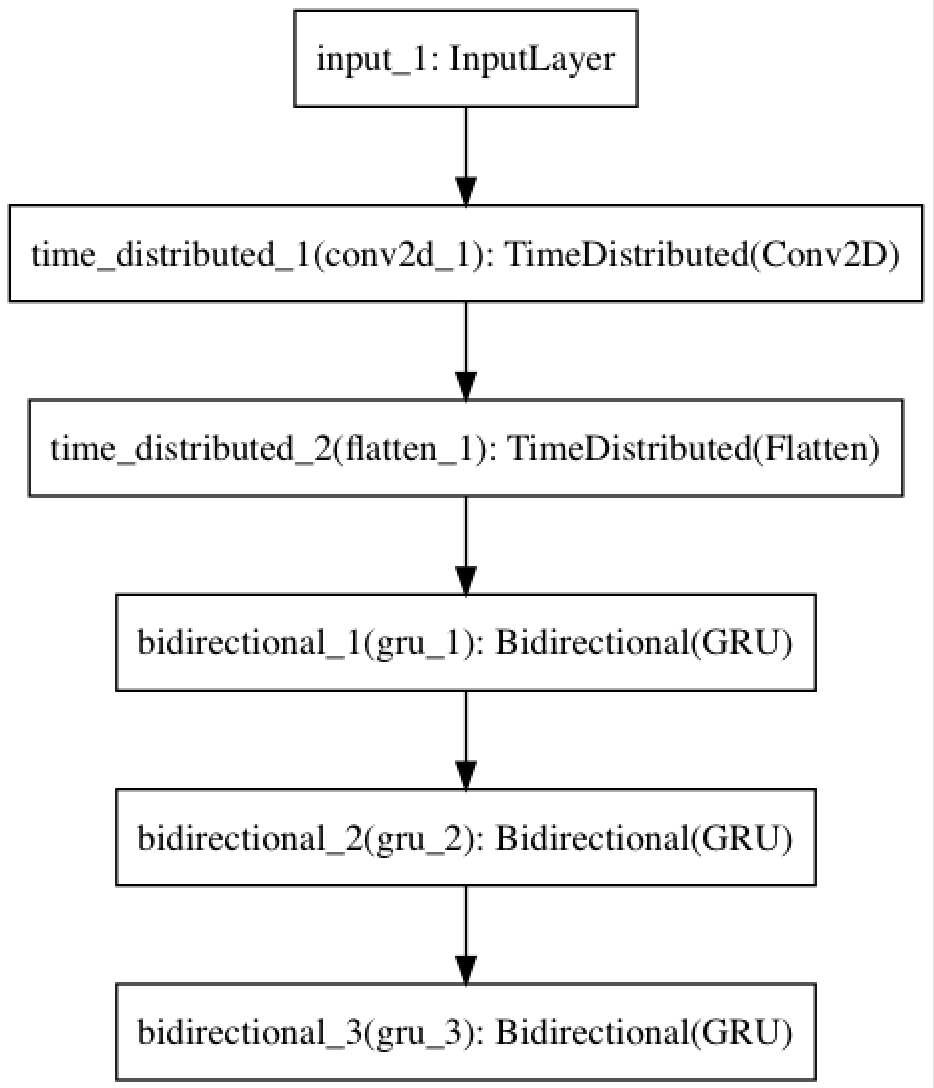
ADLxMLDS hw1 report

# Model description

* RNN  
  將每一段聲音切成有重複的數個等長小段，例如：’12345’會切成’123’,’234’,’345’，透過分別的trainging 和predict後，將結果在重疊的部分作投票多數決得到最後答案。經過實驗測試，在取長度為13時效果較好，這樣作的好處是可以維持RNN輸入的一致性，不會有不同長度的問題，且可以保留較多的training data。  
  Model如下圖：



* RNN＋CNN  
  將每一段聲音切成有重複的數個等長小段，例如：’1234567’會切成’12345’,’23456’,’34567’，在將每個小段切成等長小小段，例如：’12345’會切成’123’,’234’,’345’，將小小段送入CNN後得到的值再送入小段做RNN，因為前處理的時間有所限制，因此只有取3作為CNN的window。並因為mfcc是13\*3 dims，所以CNN的channel我設為3，並作Conv2D。實驗結果在RNN sequence 取13時表現也較好。Best model也是此model。  
  Model如下圖：



# How to improve your performance

* Method 1  
  作雜訊或是錯誤預測處理，例如：預測的結果為’aaabaa’，則有很大的可能b是雜訊或是錯誤判斷的結果，因此應該把它拿掉，處理的方法為開一個window，slide為1，作投票多數決處理，則假設window size 為5，則原預測’aaabaa’經由’aaaba’和’aabaa’的投票都是a為多數，因此預測改為’aa’ 。經此處理後的RNN model會有明顯的performance提升如下表：（在window size 設為7的時候表現最好）

|  |  |
| --- | --- |
| Window size | Public error |
| 3 | 12.830 |
| 5 | 11.367 |
| 7 | 10.819 |
| 9 | 11.406 |
| 11 | 13.045 |

* Method 2  
  對RNN不做padding而選擇作固定長度的切割的好處除了可以維持RNN輸入的一致性，不會有不同長度的問題以外，且可以保留較多的training data，此外還可以避免太長的sequence訊息會被遺忘等問題。  
  透過不同長度的RNN設置也會有不同的performance表現：

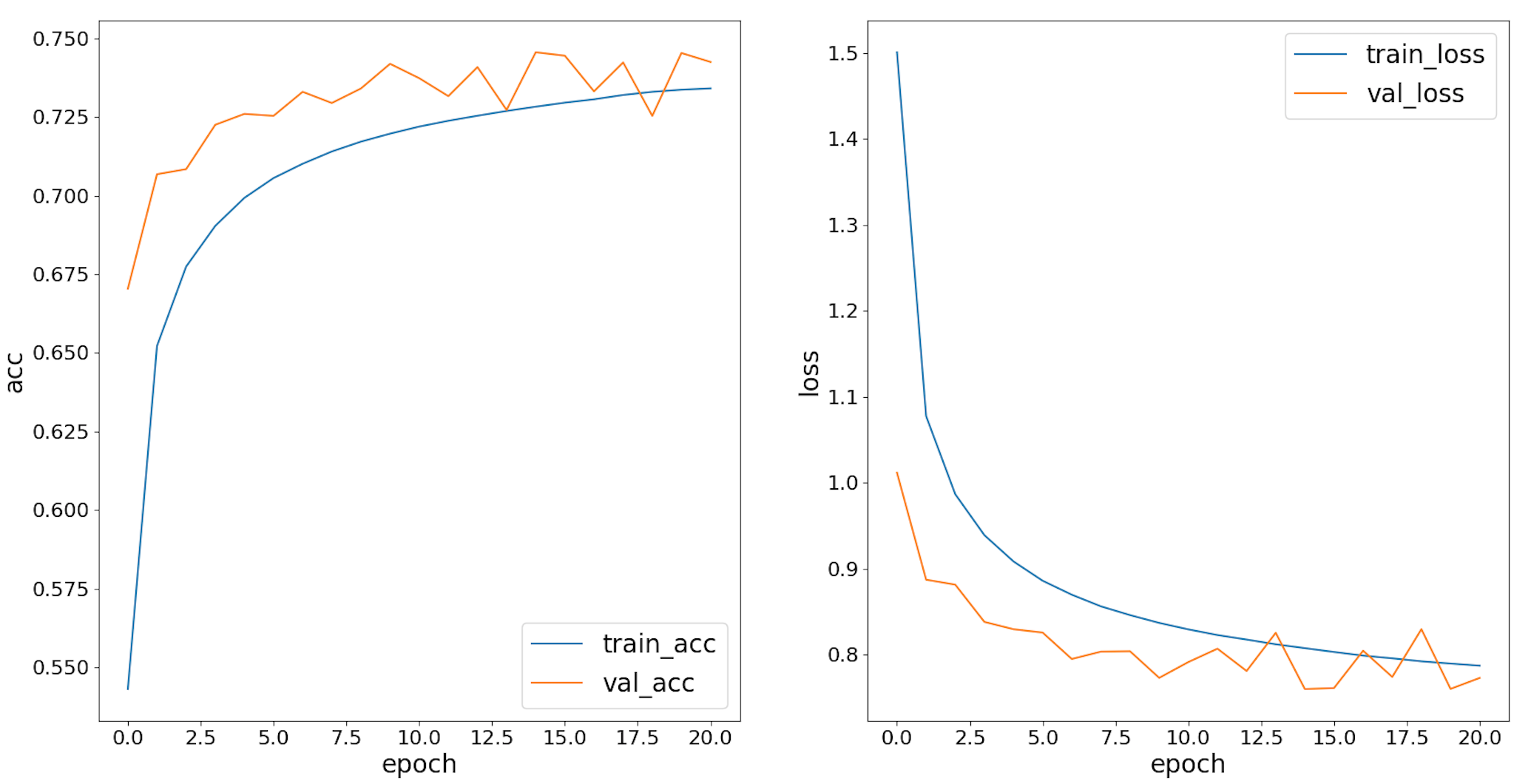
|  |  |
| --- | --- |
| Length of RNN | Public error |
| 9 | 13.045 |
| 11 | 10.548 |
| 13 | 9.875 |
| 15 | 11.152 |

# Experimental results and settings

* 比較RNN和RNN+CNN，由於我的前處理為了保留較多的data，因此RNN+CNN的資料量會是CNN的k倍，k為CNN的window size，因此前處理時間和訓練時間都增加不少，不過都能在十分鐘內，performance表現如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Model name | Sequence length | Public error | Private error |
| RNN | 9 | 10.819 | 10.698 |
| RNN | 13 | 9.627 | 9.460 |
| RNN+CNN | 9 | 9.700 | 9.946 |
| RNN+CNN | 13 | 8.723 | 8.696 |
| RNN+CNN+BN | 13 | 8.610 | 8.301 |

* 下圖紀錄了CNN+RNN的訓練過程，有設置earlystopping、dropout、batchnormalization。



* 比較GRU 以及 LSTM，實務上結果差不多，但因為GRU的參數較少訓練較快，因此都使用GRU來實作，下圖為相同設置測試結果。

|  |  |
| --- | --- |
| Model | Public error |
| GRU | 15.276 |
| LSTM | 16.525 |