Report

1. Data

The labeled data set consists of 50,000 IMDB movie reviews, specially selected for sentiment analysis. The sentiment of reviews is binary, meaning the IMDB rating < 5 results in a sentiment score of 0, and rating >=7 have a sentiment score of 1. No individual movie has more than 30 reviews. The 25,000 review labeled training set does not include any of the same movies as the 25,000 review test set. In addition, there are another 50,000 IMDB reviews provided without any rating labels.

2. Data preprocessing

2.1 Tokenize sentence. Remove stop words and punctuation.

2.2 split data: 70% for training, 30% for testing

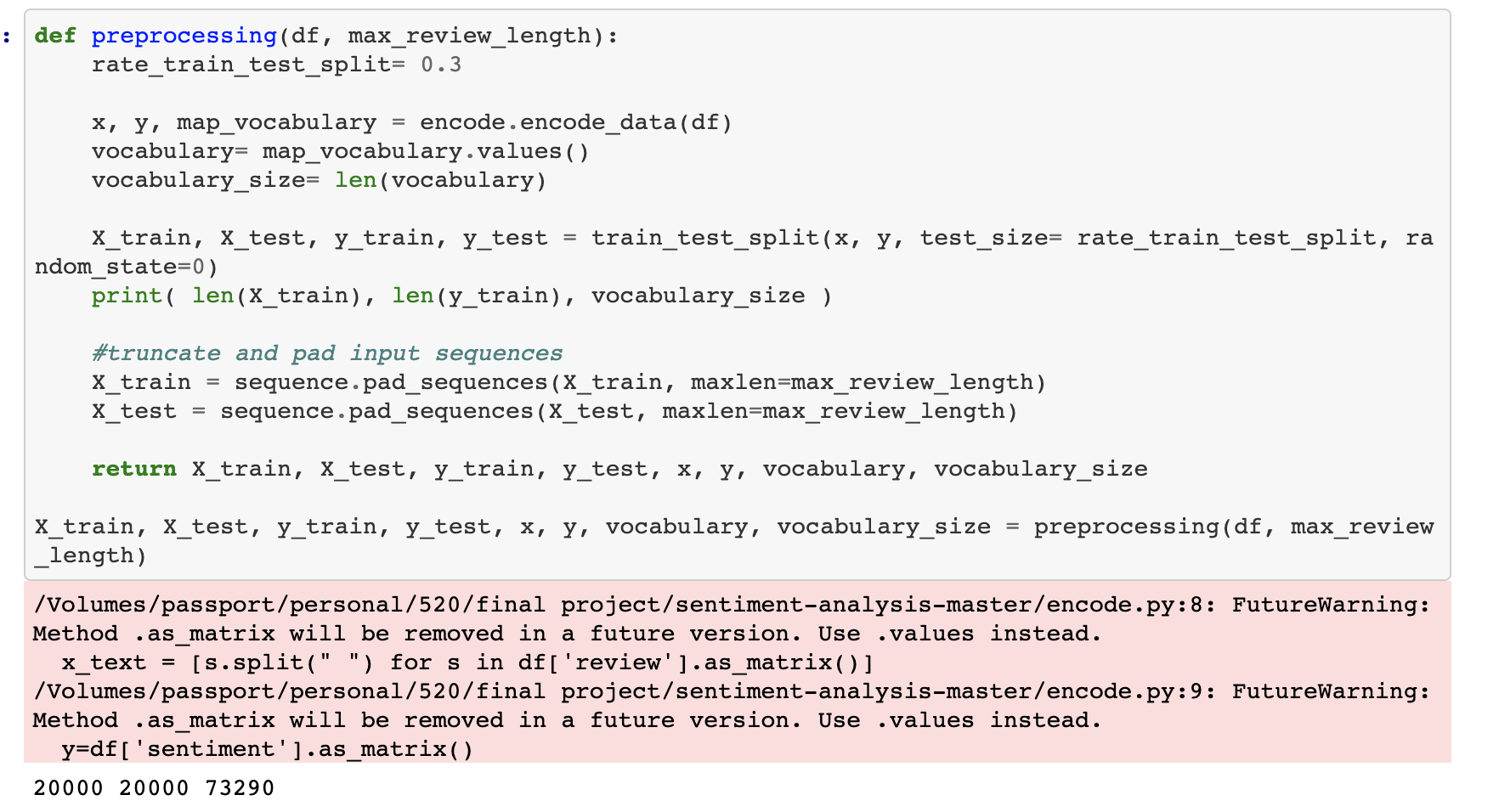
3. Word embedding

<https://code.google.com/archive/p/word2vec/>

word2vec-GoogleNews-vectors with 300 dimensional….

4. CNN

<http://www.joshuakim.io/understanding-how-convolutional-neural-network-cnn-perform-text-classification-with-word-embeddings/>

In CNN model, input layer is token words with trained google news word embedding. Then we choose filter with 1, 2,3,4. After that, dense layer and dropout layer….

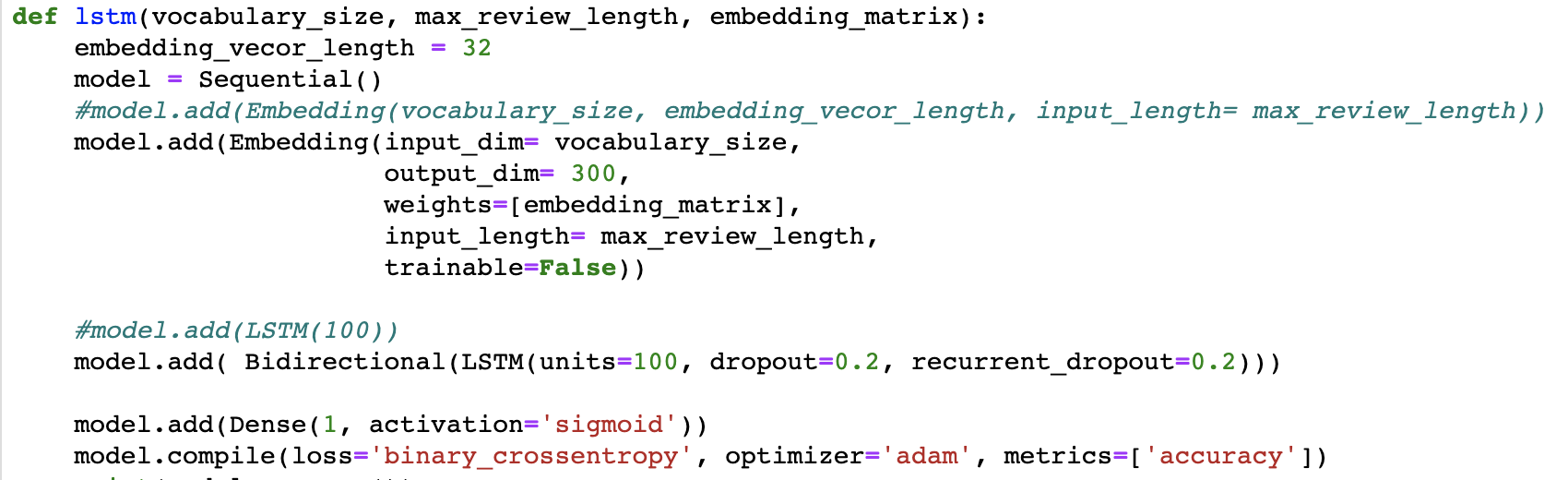
4. LSTM

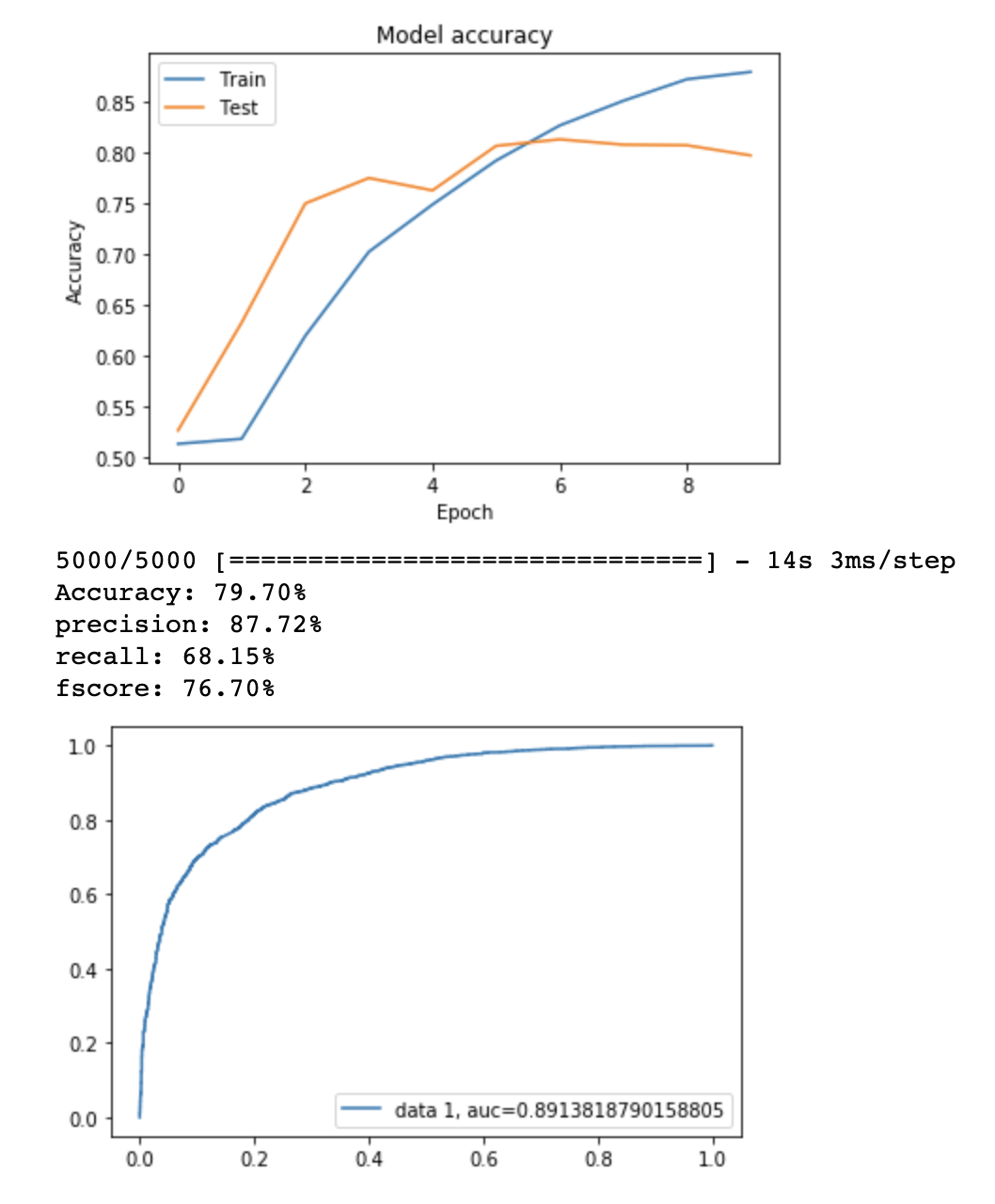
In RNN model, LSTM is chose. Bidirectional LSTM ….

<https://machinelearningmastery.com/develop-bidirectional-lstm-sequence-classification-python-keras/>

5. Training and Metrics

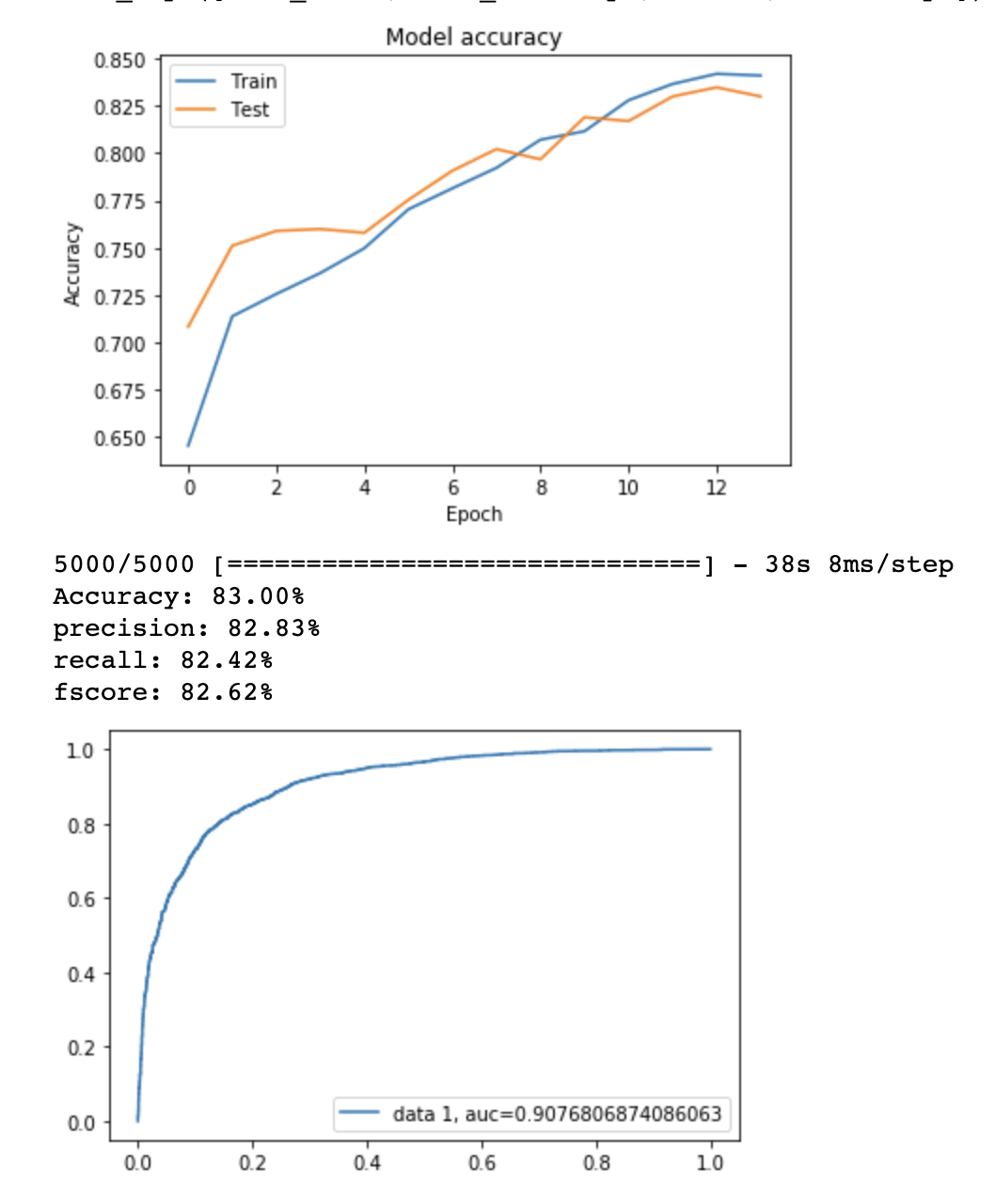
5.2 cnn

In Epoch 6, testing accuracy begin going down but training accuracy is still going up. This is overfitting. Before overfitting happen, we need stop it and save trained model. Here accuracy is 79.70%



5.2 LSTM

In LSTM metrics, do the same thing as CNN for training process. Accuracy is 83%. It is higher than CNN.



5.3 source code

<https://github.com/taixingbi/hu_520_final_project>