

Deep Learning

LAB-3

Sudheesha Reddy Musku |16241536

TASK

The main idea of this assignment is, implementation of **Text Classification** with different types of models like CNN, RNN and LSTM and also, illustrating the same with new data set which is not been used in the class. Another task of this implementation is to analyze all the three models and review which one is the best fit among all the models used.

INTRODUCTION

Text Classification is basically a task in Natural Language processing in which documents are represented with sparse lexical features using linear or kernel methods. A few of the recent technological examples of Long Short Term Memory Systems (LSTMS) is in neural systems which is a multiclass classification. CNN, RNN and LSTM models have been applied on the same dataset to compare the accuracy of the model.

CNN- Convolutional Neural Network Systems

If we consider machine language, CNN is a convolutional neural system used in visualizing images. Convolutional refers to as a sliding window connected to an $n \times n$ grid. Convolution and Sub inspecting are the most significant part in CNN.

RNN-Recurrent Neural Networks

It is a class of simulated neural systems whose motive is to establish coordinated cycles with associations between the units which displays dynamic temporal behavior. Unlike feedforward, RNNs utilize the inside memory for grouping the subjective info which results in unsegmented, associated penmanship acknowledgment or discourse acknowledgment.

(LSTMS)-Long Short Term Memory Systems

They are a type of RNN, which works on expansive assortment of issues. The main motive is to recall data for a long stretch of time.

OBJECTIVE

The objective of this task is to apply classification of text into my chosen dataset using CNN, RNN and LSTM. We have discussed below, how we have approached this scenario and what data set with have taken to analyze that. We have also, discussed which model out of them is best and why. Basically, the accuracy of the model is evaluated which helps in choosing the best type pf model. Here, we are taking the case of a crime investigation description of different classes.

APPROACHS/METHODS

1. First, we upload the data set received from the police department crime announcement framework. The data set contains reports of a certain time interval. The pivot in the data set changes each week, like for example, week 1,3,5,7 will be test set and week 2,4,6,8 will be preparation set. The record file contains columns like date, category, description, address, etc.
2. Next , the data is fed to a protocol buffer format for the separation of data preprocessing.

WORKFLOW:

1. Reading the data and generating class of information
2. We build a vocabulary by analyzing the words in the data.
3. A chart needs to be generated after the completing of information preprocessing
4. For the data sources and yields, the placeholders are created.
5. The information received is made non linear by initiation work.
6. Now, the information variable is duplicated according to the weights of the individual variable.
7. The model is now constructed, and accuracy is analyzed
8. Lastly, we generate a map of our concept using tensor board.

DATASET

The dataset used was the criminal record classification for the city of Kansas. It includes all kinds of crimes incident or reported all over the city. The criminal record dataset consists of 10k records which are equally divided into training data and the testing data.

PARAMETERS

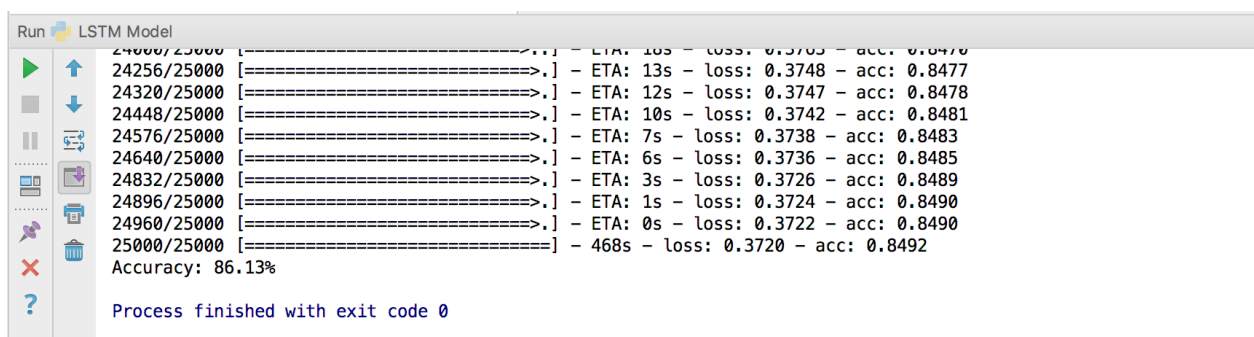
The parameters used for building these three models are:

1. An embedded layer have been applied to the model initially to improve the model fuctionality.
2. Few basic parameters are number of words=5000, vector_length=32 and rev_len=500
3. Sigmoid is the activation fuction used for these models.
4. Calculate the loss generated after each iteration.
5. _Optimizer have been used for text classification.
6. By using all these above parameters accuracy of the text classification has been calculated by using CNN, RNN and LSTM.

EVALUATION & DISCUSSION

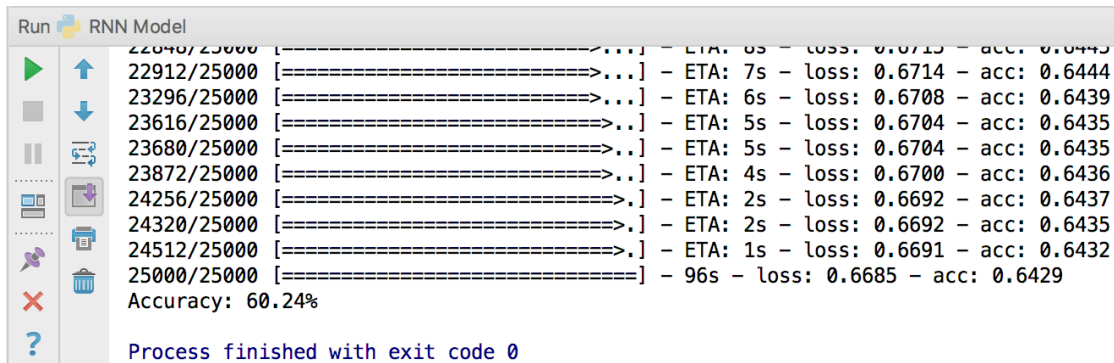
Below are the screenshots for the accuracies using three different models.

LSTM MODEL

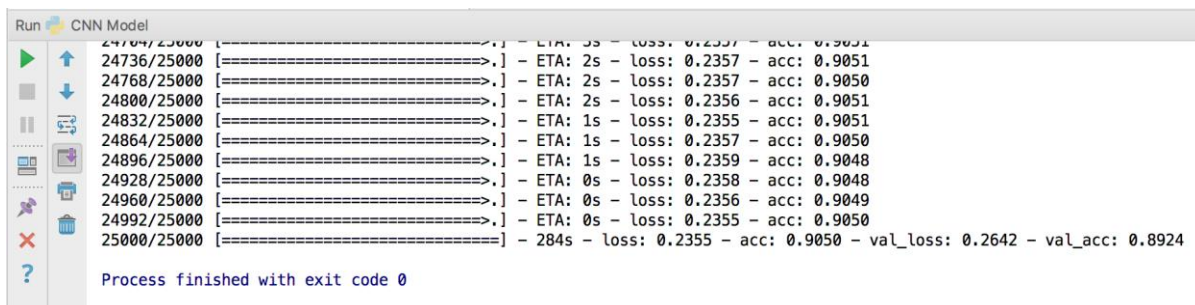


```
Run LSTM Model
24000/25000 [=====] - ETA: 10s - loss: 0.3703 - acc: 0.8476
24256/25000 [=====] - ETA: 13s - loss: 0.3748 - acc: 0.8477
24320/25000 [=====] - ETA: 12s - loss: 0.3747 - acc: 0.8478
24448/25000 [=====] - ETA: 10s - loss: 0.3742 - acc: 0.8481
24576/25000 [=====] - ETA: 7s - loss: 0.3738 - acc: 0.8483
24640/25000 [=====] - ETA: 6s - loss: 0.3736 - acc: 0.8485
24832/25000 [=====] - ETA: 3s - loss: 0.3726 - acc: 0.8489
24896/25000 [=====] - ETA: 1s - loss: 0.3724 - acc: 0.8490
24960/25000 [=====] - ETA: 0s - loss: 0.3722 - acc: 0.8490
25000/25000 [=====] - 468s - loss: 0.3720 - acc: 0.8492
Accuracy: 86.13%
Process finished with exit code 0
```

RNN MODEL



CNN MODEL



From the above screenshots it indicates that CNN model has the highest accuracy among all the three followed by LSTM and then RNN gives the least accuracy.

CONCLUSION

As we get the results of all the models, the data has been classified according to CNN, RNN and LSTMS models. Also, the accuracies of each has been observed. So, we find that CNN model stands out better than the rest of the models. It is more accurate and easy to analyze. LSTMS also is a good model after CNN but a bit slower.