Univ. of Athens, Dept of Informatics & Telecoms
YS 19: Deep Learning for NLP
Fall 2022 - Homework 2

Kyriakopoulos Dimitrios - sdi1900093

Notes to the graders: I provide a fully documented ipynb notebook with a link to Google Colaboratory. In order to run the models on the test dataset

1. Go to the "Load the dataset" section (Second code cell of the Notebook).
2. Change `DATA_PATH` value to the path, where the file with the data for training is located.
3. Change `TESTING_DATA_PATH` value to the path, where the file with the data for testing is located.

1 Introduction

In this project, I perform sentiment analysis on IMDb movies' reviews. The inputs to my model are GloVe word embeddings. I classify the reviews into two classes (Positive and Negative) using FFNN (Feed Forward Neural Networks) and I provide a detailed comparison for each method I experimented with.

2 Text Encoding

In order to convert text into numbers I used GloVe embeddings from https://nlp.stanford.edu/projects/glove/. So as to convert each text of the dataset into numbers I check for every word in the text if appears in the GloVe dictionary. If yes, I get the vector from the dictionary for the word, else the vector for the word contains zeros. In order to get the features vector for each text, I add the vector of each word.

3 Experimenting

I choose a model which consists of one input layer, one hidden layer having the ReLu activation function and one output layer having the sigmoid function. As mentioned in choosing activation function sigmoid function is recommended

as activation function for the output layer for binary classification problems. In addition I choose the model to have one hidden layer as this is considered sufficient for the large majority of problems, as said in choosing no of layers - neurons. In the same report they state that the optimal size of the hidden layer is usually between the size of the input and size of the output layers and so I selected the neurons in the hidden layer to be 150 as the input layer has size 300 and the output has size 1. I mostly experimented with the value of learning rate and weight decay by plotting learning curves for several values combinations, which show that **learning rate: 0.00001, weight decay: 0.01** are the optimal values which is showed in the below Figure 1. Furthemore, in order to define

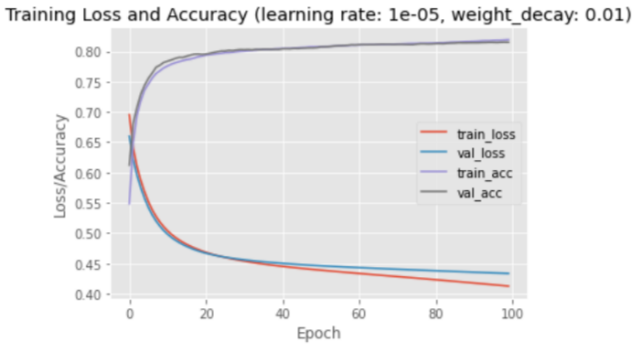

Figure 1: Learning Curves.

the number of features to use from GloVe emdeddings I experimented with all the available values which are: 50, 100, 200, 300 plotting the accuracy for each number of feature. Below are the results (Figure 2) which show that 300 is the optimal number of features to use. What is more, I choose the batch size to be

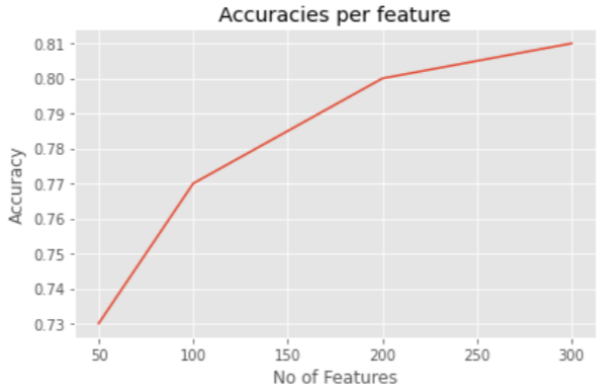

Figure 2: Accuracies per features.

256. Changes to batch size, number of layers and number of neurons did not have any tremendous effect to the performance of the model. Finally, Adam is chossen as loss function. Any other loss function had worse results.

4 Results

As metrics for the performance of the model I use precision, recall and F-measure whose values are shown below (Figure 3). The train ratio of the dataset is

	precision	recall	f1-score	support
0.0	0.82	0.80	0.81	4550
1.0	0.80	0.82	0.81	4452
accuracy			0.81	9002
macro avg	0.81	0.81	0.81	9002
weighted avg	0.81	0.81	0.81	9002

Figure 3: Scores.

80%, while the 20% of the dataset is used for testing. I also plot ROC curve (Figure 4) which is a graph showing the performance of a classification model at all classification thresholds, which shows that the model is quite efficient to distinguish between positive class and negative class.

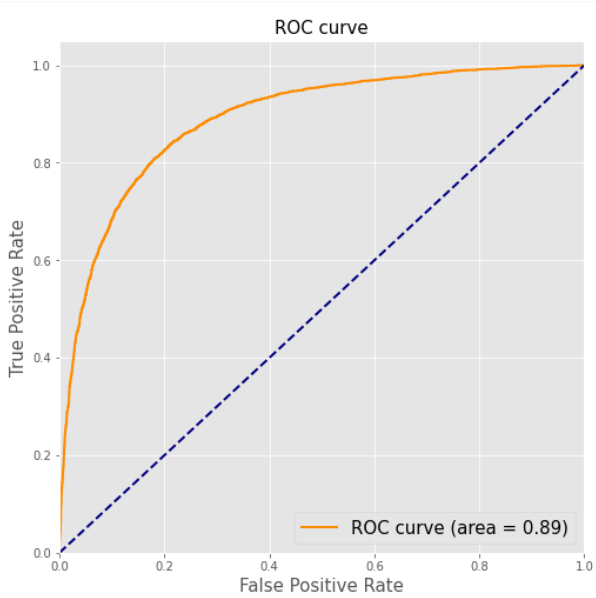

Figure 4: ROC curve.