Exercise No. 1
Exercise Title: Natural Language Processing (NLP)
Name: Farzam Taghipour
Student ID: 95222018
Department of Computer Science

Abstract and Problem:

Since machine learning started ruling over problems, Natural language processing problem became a great major in ml realm. We are surrounded by millions of text book, papers that contain billions of sentences in different languages. But how can we use these data to solve our problems like: essay evaluation, grammar checking, etc. We are familiar with image processing that we have each single pixel valued by 0 to 255, and can be used as feature during a image processing problem. But sentences and words are not valued, So how can we solve this?

Solution:

The solution is simple, give values to words manually!
But there are thousands of millions of words in a data set, Is it possible to manually number them?
Fortunately, Since 2017, a group in Google started working on a project called TensorFlow. TensorFlow is a framework that helps us almost every section of solving our machine leaning or deep learning problem such as: prepossessing data, splitting data to train and test set, modeling, fitting, etc.
Fortunately, There are methods for NLP, which are used in the notebook.(all technical details are noted in main.ipynb).

Evaluation:

Over fitting is a major problem in NLP, So I tried to split my data in a way to reduce over fitting. Also in model I used DropOut layers and for Conv1D layers, I used l2 regularization with Lambda 0.001. But as suspected, model barely could fit on train set and reach 80% accuracy.
I noticed that not only word frequency but also word position in important to predict more accurate.
For example in the sentence:
I lived in Ireland, I went to school there and at school, they teach me to speak …
we know that Irish people speak Gaelic, which can be concluded by looking at the beginning of the sentence that mentioned the writer was living in Ireland. So the key to predict accurately here, is to know writer was living in Ireland. This made me to use recurrent neural networks such as LSTM and GRU. LSTM has a memory cell that can store words in model, So it might be helpful here.

Conclusion:

I tried fitting my LSTM model over training data set but the process was really slow and after 10 epochs it reached 90% accuracy over training set and 70% over validation set. It was a bright symptom of over fitting again. I changed the model construction and used GRU layer. After 20 epochs and 4 and half hour training, it could reach accuracy of 98.9 percent over training set and 80% over validation set. Although still we are over fitting over training set, It is a great outcome.