CSE440 Project– Fall 2023

Submit one file per team through this form. **Only include your codes and the report-- do not include the data file or the embedding file.** Put all the codes and the report in one folder, zip it and submit it. Your codes have to be in a notebook. Extra points will be given if you can make your data paths dynamic.
P.S. You will have exactly one chance to submit. So, do not mess it up. If there are multiple submissions from one group using different emails, I will deduct points.

Setting up environment:

- Install Numpy, Scipy and Scikit-Learn if you have not already. Your python installation may already have it, but just make sure they are there.
- This project will require you to build a neural net classifier for the task. For that you will need Tensorflow, Keras or PyTorch. For learning purposes, I prefer Keras. To install Keras (that uses Tensorflow as a backbone), follow this tutorial. You will need this tutorial up until the "Get a version of Python, pre-compiled with Keras and other popular ML Packages" section.

Data download:

- Download this file.
- Unzip the file imdb_440.zip. It will contain the actual data file in .csv format. This is a set of reviews from IMDB.
- Download this file. This will have almost all the words in the English language and their GLoVe word embedding in 100-space (that is, each word is represented by 100-dimensional vectors).

Tutorials and other helpful comments:

- Useful tutorial: for all the questions, you will find this tutorial extremely useful. You can always go online and search for solutions as these questions are really common across the world for teaching NLP, so there's no point stopping you from doing that. What I want you to do is to learn why something is done and how you can make it better-- and for that, I can give you the tutorials I like the most. Do not plagiarize-- use these codes to learn.
- The personal computers you have will probably take a long time to run these codes. You can use Google Colab to run your codes faster.
- Help each other, communicate with each other– no worries. But please do not copy from another group. I am very liberal in terms of what is allowed and what is not, that is, discussion, using online tutorials etc. all are allowed– but, if I find anyone copying, I will give you a straight zero.

Coding Tasks:
We will build a shallow (single hidden layer) recurrent neural network model to classify IMDB reviews. You have to:

1. Load the dataset you downloaded using Pandas.

2. Use a pretrained representation technique to convert the reviews into vectors. You should have already downloaded the GloVe vectors). Use GloVe. The tutorial given will have a section on how to convert the reviews to vectors using GloVe.
3. Split the data into two parts, training and testing. You will train a machine learning model on the training part of the data, and then you will test its performance on the test part of the data. Use an 80-20 split for your data, that is, 80% of the data will be for training and 20% of the data will be for testing.
4. Train it for 20 epochs once and then test. You can play with the hyperparameters like changing the batch size etc.
5. Now, train the shallow model. Use the tutorial given in the preamble. Your model will have three layers: one embedding layer with output shape 100 which will convert your words to a 100-length vector (find GloVe for 100 dimensions), one dense layer with an output shape 10, and the final output layer (another dense layer) with output shape. Do not use any pooling or any other layer.
6. Now, let's improve the model by introducing gated recurrence relation to it. In the previous neural net structure, there were no gates-- it was one dense (fully connected) sequential layer after another. Now, instead of the first dense layer, let's use an LSTM. Replace the first dense layer (that was taking input from the embedding layer and producing a 10-sized output) with an appropriate LSTM layer (you can find hundreds of tutorials on how to do that online, for example, this one is great). We will experiment with two versions of this:
    a. A single unidirectional LSTM layer
    b. A single bidirectional LSTM layer
7. Analyze the performance of these models with the original one. Which one is the best? Why is it the best? Suggest a couple of improvements so it can be better.

Writing tasks:

1. You need to submit a project report with your codes. This report will include what you did, what results did you achieve and how you can improve these results. There is no format– use your own style. I will not guide you through the process of writing a good report– this is not an English class. Use your own judgment. You need to make sure that your report is legible, coherent, informative and useful.
2. Maximum size of the report will be 4 pages.