Final Project: Sentiment Analysis **CSCE 633**
Due: 11.59pm on December 12, 2023

Instructions for final project submission

a) You will need to provide your code for your solution
b) You will need to provide an IEEE-style written report
c) You will need to prepare a 10 minute recorded presentation of your project
d) Create a **single zip** and submit it on **CANVAS**. Please note – you may need to be careful with how the presentation is recorded and submitted – please check early with us on alternatives, if you need one.
e) You can run your code based on Google Colab if you wish.

We will use the Yelp review dataset, which comprises around 174000 reviews with stars. We will be using only a subset of this dataset for experiments. Our goal is to implement the powerful *Transformer* model for sentiment analysis based on the text review and stars.

Inside the "Final Project" on CANVAS, you can find two files, named 'yelp review train.csv' and 'yelp review test.csv'. Each file contains a set of reviews posted by users on Yelp.

15% of your grade: Presentation – a 10 minute presentation in which you provide an introduction and background, a clear description of your methods, how you tested your project and how you evaluated the final results, and concluded that your project is successful

15% of your grade: Written Report – at least 5 pages in the format of the scribe notes. The paper must include an introduction (with citations towards related work and a review of state of the art), your methods (including citations you used to come up with your approach), your experiments and results, including how you tested your model and have confidence it is correct, and conclusion.

70% of your grade: Code – broken down by the following % (adding up to 100)

(a) (15%) Data pre-processing: Pre-process the data by removing the punctuation and stopwords and converting all words to lowercase. Moreover, converting the stars into three levels: Positive > 3, negative <= 2, and neutral = 3. Finally,
Note: You can use the *nltk* library from here: https://www.nltk.org/ to remove stop words. The regular expression may be helpful.

(b) (20%) Input data preparation: The input of the Transformer model is a fixedlength review sequence where integer numbers represent words. In this part, you need to build vocabulary for the dataset and pad the review data to a fixed length.

(c) (40%) Transformer implementation: Implement a Transformer model which is composed of an encoder network (i.e., multi-head self-attention layers) and a prediction head mapping the hidden representation of input sequence into the label space (i.e., three classes). *Note:* You can find more details about Transformer at here https://arxiv.org/pdf/1706. 03762.pdf. You may need to implement positional embeddings, a vocabulary embedding table, and mask indicators for padded tokens. Pytorch is recommended for model implementation.

(d) (15%) Model training: Train the model with stochastic gradient descent using mini-batch fashion based on the 'yelp review train.csv' dataset. Print the training curve, where the x-axis is the training epochs, and the y-axis is the training accuracy.
Note: You can randomly sample a small set of training data as the validation set and save the best model with the highest validation accuracy.

(e) (10%) Result analysis: Load the best model saved during training and report the accuracy of the model on the test set (i.e., 'yelp review test.csv'). What are the impacts of hyper-parameters, such as the hidden dimension and the number of attention layers, on the Transformer?