Computational Linguistics Final Project Part 1

1. Speech to text (English vowel phonemes)
2. The problem of this project idea is to translate an audio file of an English vowel phoneme into a text representation in IPA transcription. For example, if I have a dataset of audio recordings of people pronouncing different vowel phonemes, my NLP program will be able to decide what phoneme each recording contains.
3. The algorithm I intend to implement will be a support vector machine. This algorithm is used for a similar research topic in the following papers:
   1. MFCC and SVM Based Recognition of Chinese Vowels (<https://link.springer.com/chapter/10.1007/11596981_118>)
   2. Phoneme Recognition Using Support Vector Machine and Different Features Representations (<https://link.springer.com/chapter/10.1007/978-3-642-28765-7_71>
4. I will evaluate my results with a loss function that helps maximize the margin between data points (English vowel phonemes represented in a high dimensional space) and the hyperplane created by the SVM. During training, the loss function will help generate the optimal hyperplanes to correctly classify our data.
5. I will need a dataset of English vowel phoneme pronunciations from a diverse population of people to help generalize my model. I can obtain this dataset from the University of Western Michigan Vowel Database (<https://homepages.wmich.edu/~hillenbr/voweldata.html>). This dataset was available online and it contains all vowel phoneme pronunciations from 150 individuals (50 of each men, women and children).
6. The potential users of my finalized project will be linguistics students who would like to practice IPA transcription of vowels by using my program as an answer key to their practice, assuming my program reaches sufficient classification accuracy.
7. Binary Sentiment Analysis
   1. The problem of sentiment analysis is to decipher the emotion of a user from a written or spoken corpus. In the case of this project I will accept a body of text as input and output the ‘writer’s’ tone (positive or negative). For example, if the input to my program is, “It seemed mediocre at best.”, my program would probably output “negative”.
   2. Many algorithms (rule based and data driven) are used for sentiment analysis, however, I am deciding to use a Recurrent Neural Network known as an LSTM (Long Short Term Memory). This type of algorithm is described in section 9.4 of our textbook.
   3. I will evaluate my results by using supervised learning to train and evaluate my LSTM. This means I must acquire a labeled dataset, then split it up into a training set (for training) and a validation set (for evaluating the accuracy of my results).
   4. Referencing the above section, I need a dataset or corpus labeled with the following sentiments (positive and negative). I found one available online from a Stanford AI lab (<http://ai.stanford.edu/~amaas/data/sentiment/>). This data is freely usable but requires a citation to the creator (Andrew Maas) if used in a publication.
   5. I find an especially useful application in this program if it is used to read our SMS messages or emails and notifying us beforehand if it has a negative or positive sentiment as a friendly warning.