Bridger Hackworth
7/14/22
CSE 450

Module 07 - Prepare

For my project I have chosen to use an LSTM or GRU to do language translation. My goal is to build a deep neural network with LSTM layers that will accept English text as its input and return the Spanish translation with the highest accuracy possible. To do this, I will be attempting to build upon an example/walkthrough that I found online of a very simple model that does this for French and was only trained on 227 English words and 355 French words. For my own model to reach the scope that I would like it to, I will attempt to implement transfer learning and use a pre-trained embedding layer such as Google's word2vec. I will also use a data set obtained from Anki that contains 118,964 pairs of phrases. Unlike our previous RNN class project, our outputs from our model can be tested against their actual translations, meaning it can be evaluated using accuracy. I currently don't have an idea of what kind of accuracy will be possible, but hopefully I can get something close to 100%. I will also have to implement an LSTM layer on my own, as this is not something that was included in the example that was given (only a GRU layer). Perhaps a LSTM layer will not work as well in this case. This will be something that I will find out. The preprocessing will also be different for my dataset. Although, this example here shows how to preprocess the Anki data for the task I am attempting.

The first article from towardsdatascience.com is super informative and helpful with clearly explaining what I need to do to complete this task. Some of the key things that I learned are the broken down process of preprocessing through tokenization and padding, both things I was unfamiliar with. It also gives a good idea of how the flow of the model should be with an explanation of why for each part. First the input layer must accept the words and then convert each word into an integer. Next, the embedding layer converts each word into a vector so that similar words can be grouped together in a theoretical, n-dimensional space. Next the recurrent layers do the actual grouping of these vectors, using time steps and states to store the positional information. Next the dense layers decode the encoded words into the correct translation sequence. Finally the outputs are mapped to our dataset of Spanish words.

It seems that python, Keras, Google Collab, and its GPU will be the main tools in this project.

Here is a link to my colab notebook:
https://colab.research.google.com/drive/1MYKCl1glAlG6aSMMtRvAcMqjKodHH2Zc?usp=sharing