Steve Su, steven36, captain

Pierson Wodarz, wodarz2

Our group will participate in the text classification competition.

From an initial web search, many types of methods have been used to perform text classification, including decision trees, support vector machines, K nearest neighbors and artificial neural networks (ANN). Current state of the art algorithms involves ANNs which includes both convolutional neural networks (CNN) and recurrent neural networks (RNN). CNNs and RNNs differ in that RNN’s architecture allows it to recognize chronological sequences, such as words in sentence, better than CNNs. CNNs are typically better recognizing adjacent regions of input data such as adjacent pixels in an image. As a result, they are common in image classification. For our task of sarcasm classification, the RNN might be a better choice. However, there is [recent literature which shows success with CNNs](https://www.researchgate.net/publication/333319181_Deep_CNN-LSTM_with_Word_Embeddings_for_News_Headline_Sarcasm_Detection) for text classification tasks. We plan to utilize one of these two neural networks for our project.

Additionally, this project involves understanding and utilizing the libraries which perform the NN analysis. In order to successfully classify the text data, we expect we will need to perform several steps:

Data Loading: Transform the raw data into acceptable inputs for the tools and models.

Example tools/data structures: NumPy arrays, TensorFlow Dataset objects

Considerations: Best strategy for transformation; how to handle hashtags, punctuation, etc.

Data Preprocessing: Tokenize the data, normalization, etc

Example tools/data structures: Keras preprocessing layers

Model Building: Create a trainable framework

Example tools/data structures: Keras, Tensor Flow, Pytorch

Training: Train the model on the training datasets and tune the parameters for optimization.

Testing: Run the model over the test dataset.

We plan to use python as the programming language as we have prior experience with this language, and it has many libraries for neural network modeling (Keras, Pytorch, Tensor Flow).