CIS4930 Natural Language Processing with Python Project Proposal

Abinai Pasunuri, Ammar Syed

Problem Description

With so much going on in the world, the explosion of news sources and social media, and the ability to access the news at your fingertips, we have seen a rise in fake and unreliable news articles/sources. When people cannot detect whether a news article is spreading false or misleading information, there can be potentially dangerous consequences. It has become more difficult for individuals to distinguish between “real” and “fake” news. In this project, we hope to develop a machine learning model using natural language processing that attempts to classify and distinguish between fake and real news articles accurately.

Proposed Method

Learning Goals

One of the goals we have is understanding how we can process text and extract useful features. Another goal is learning how various machine learning models work and how we can use them to generate predictions from extracted text features. Ultimately, we hope to learn how to combine the power of natural language processing and machine learning to tackle problems such as this one.

Solution

For our solution, we first plan to obtain relevant news articles with ranging topics either through a publicly available labeled dataset or scraping news articles from both reliable and unreliable news sources. Following obtaining the data, we plan to clean and preprocess the corpus by removing stop words, lemmatization, and tokenization. Once all the data is cleaned, we plan to use methods, such as term frequency-inverse document frequency (tf-idf) or word2vec, to get the corpus in a vectorized form. Following all the preprocessing, we plan to take this data and train our machine learning models. For this project, we plan to use two different types of machine learning models. Since the data we will be using will be labeled, we will be using supervised machine learning techniques.

The first model to be used will be a support vector classifier (SVC). Since there are two categories, fake and real, for the news articles, this is a binary classification problem. Support vector classifiers use a decision boundary to classify data into two groups and will likely perform well for this purpose.

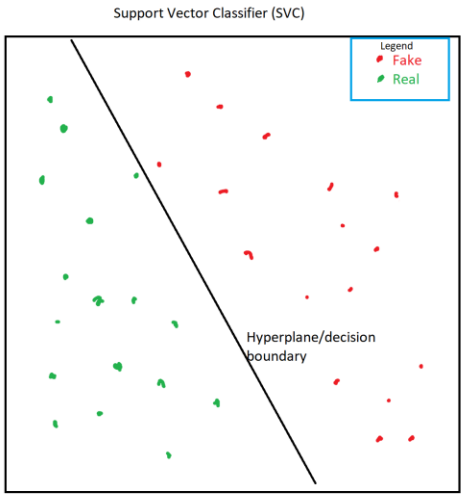

Figure 1: Overview of SVC model

The second model will be a neural network, specifically a recurrent neural network (RNN) combined with convolutional layers. The utilization of a recurrent architecture will allow for learning long-term dependencies and patterns in text data, and the addition of convolutional layers will allow for learning local features. Following the implementation and training of the models, we will test and compare their performance on different news corpus, evaluate their generalization ability and performance with various metrics. We will also have some visualizations of the dataset and the results produced by the models.

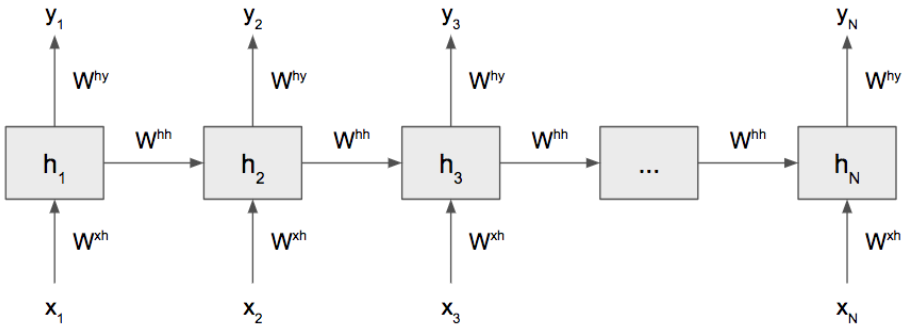

Figure 2: Overview of RNN architecture

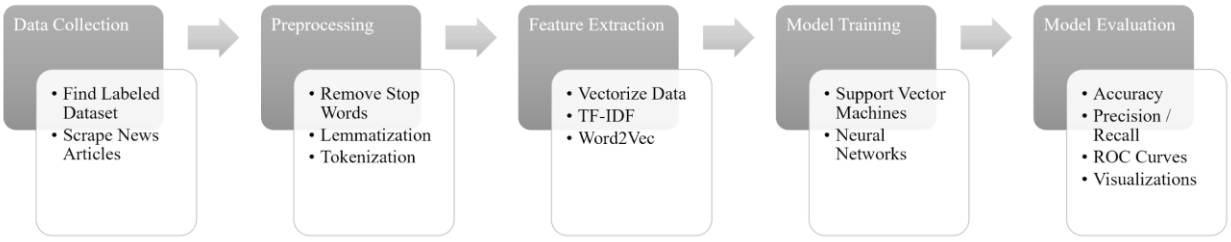

Figure 3: Overview of Classification Pipeline

Software Tools

We will utilize the Python programming language and associated libraries. For getting the data, scraping may be needed, and a tool like Beautiful Soup would be used for this. Kaggle may also be used to get a useful dataset. Numpy and Pandas will be used to handle and clean up the data efficiently. In order to preprocess, tokenize, and vectorize the data, NLTK and Scikit-learn will be utilized. In addition to these libraries, various other libraries may be needed for feature extraction of the data. Scikit-learn and Tensorflow/Keras will be used to create and train the machine learning models. Data visualization libraries such as Matplotlib and Seaborn will also be used to create graphs of the data and results.

Measurements and Comparisons

Since the following is a binary classification problem, we plan to evaluate our models with the standard metrics associated with classification, including accuracy, precision, recall, and a ROC curve. The accuracy metric will be used to get a general idea of how the model performs and how many samples it correctly classifies over the whole dataset. The precision metric will be used to understand how well our models classify positive data, which refers to a fake news article, and how often they confuse legitimate news articles with fake ones. The recall metric will be used to understand how often the models miss out on correctly classifying actual fake news articles. Lastly, a ROC curve will visualize how well the models perform with different classification thresholds. The corresponding AUC value will allow us to understand the measure of separability and the models' overall performance.