Capstone #2 -- Examining and Predicting Fake News

Introduction

The problem of fake online news is a persistent problem problem in contemporary society, impacting politics and society. Many NLP and machine learning scholars have researched this increasingly important problem and have developed highly-technical, insightful analyses. In this project, I examine the problem of fake news classification, by analyzing a large dataset of scraped news articles using various Python libraries. Using the large fake news dataset scraped by Maciej Szpakowski available at https://github.com/several27/FakeNewsCorpus , I sample a subset of news articles and perform text analysis on them. This project is relevant to people who are interested in how machine learning can interact with our understanding of our contemporary online journalistic environment of the 2010s.

Data Cleaning and Pre-processing

Initial sampling and EDA

The corpus that I used includes over 20 million articles. Because of considerations regarding time and processing power, I first decided to sample 10,000 articles for my analysis. While performing initial EDA on the corpus, it became apparent that all articles were categorized into 12 different news types (Fig. 1).

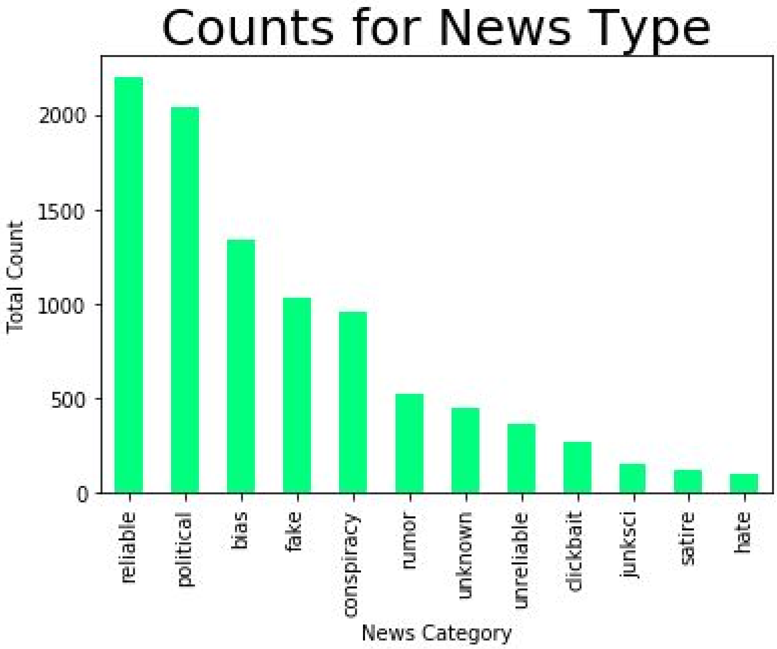
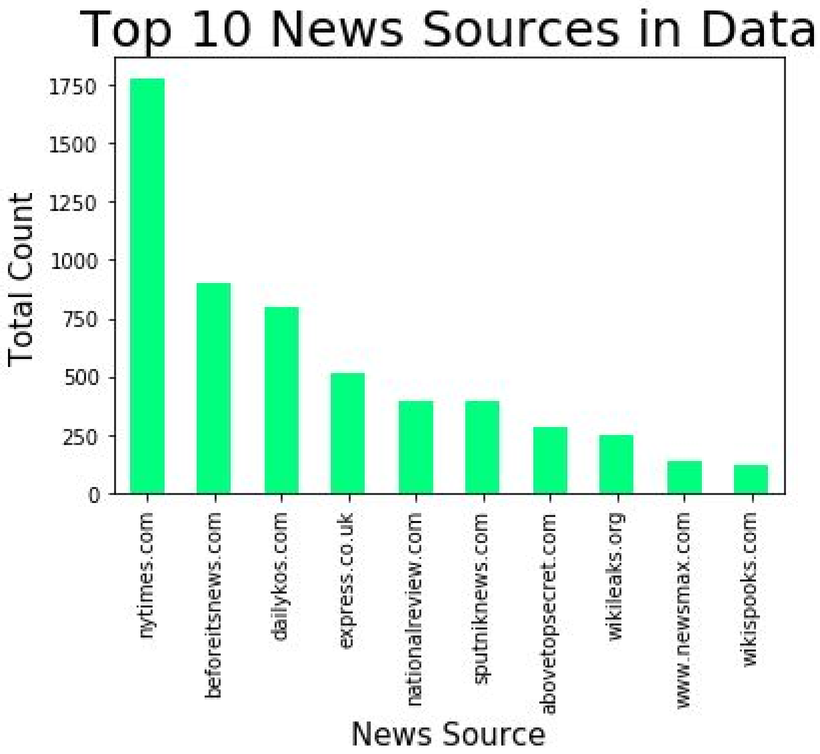


Fig 1.

Upon inspecting each of these news types, some issues began to emerge. Namely, all articles from any given source were given a particular label without consideration of individual articles. For example, all articles from nytimes.com were labelled as “reliable,” all articles from “beforeitsnews.com” were labelled as ‘fake’, and all articles from “sputniknews.com” were labelled as “bias”. Figure 2 shows the top ten news sources from particular web domains represented in the dataset.

Fig. 2



While the vast majority of articles from The New York Times can likely be considered reliable, some of the labelling of other various news sources seems to present some issues in the data. For example, all articles from ‘dailykos.com’ were labelled as ‘political’ (Fig. 5), all articles from ‘express.co.uk’ were labelled as ‘rumor (Fig. 6)’, and all articles from ‘sputniknews.com’ (Fig. 7) were labelled as ‘bias.’ Is every single article from each of these sources inherently more political or biased than articles from sources labelled as ‘reliable’? Because of the issues presented with the labelling, I decided to resample from the original dataset.

Fig. 3

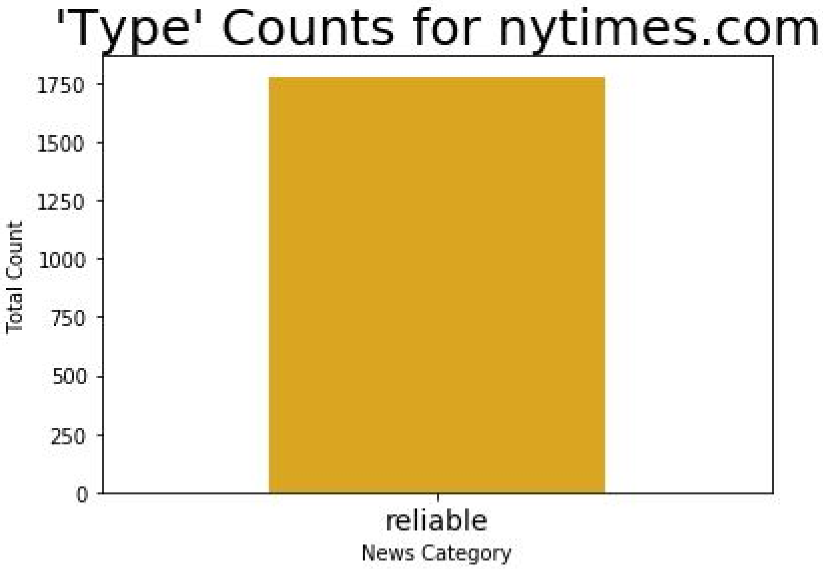

'Type' Counts for nytimes.com
1750
1500
1250
1000
750
500
250
0
Total Count
reliable
News Category

Fig. 4

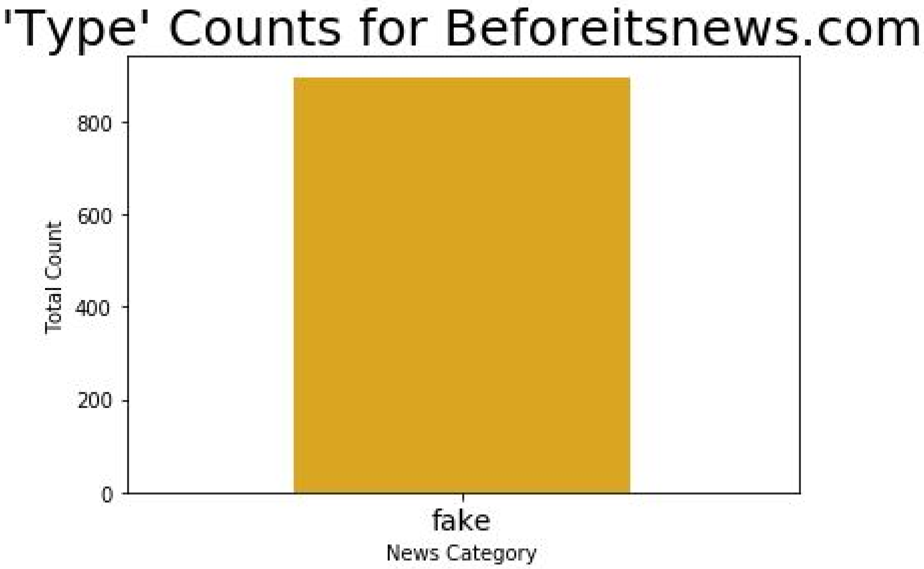

'Type' Counts for Beforeitsnews.com
800
600
400
200
0
Total Count
fake
News Category

Fig. 5

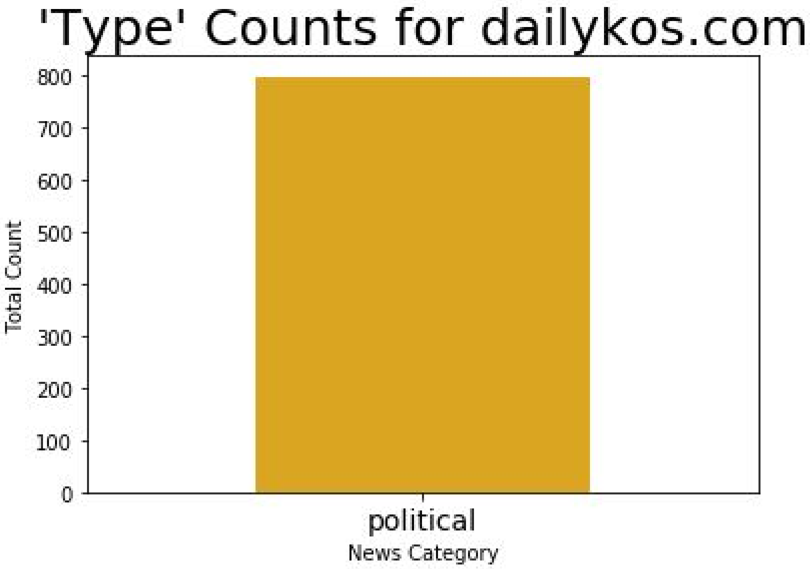

'Type' Counts for dailykos.com
Total Count
0
100
200
300
400
500
600
700
800
political
News Category

Fig. 6

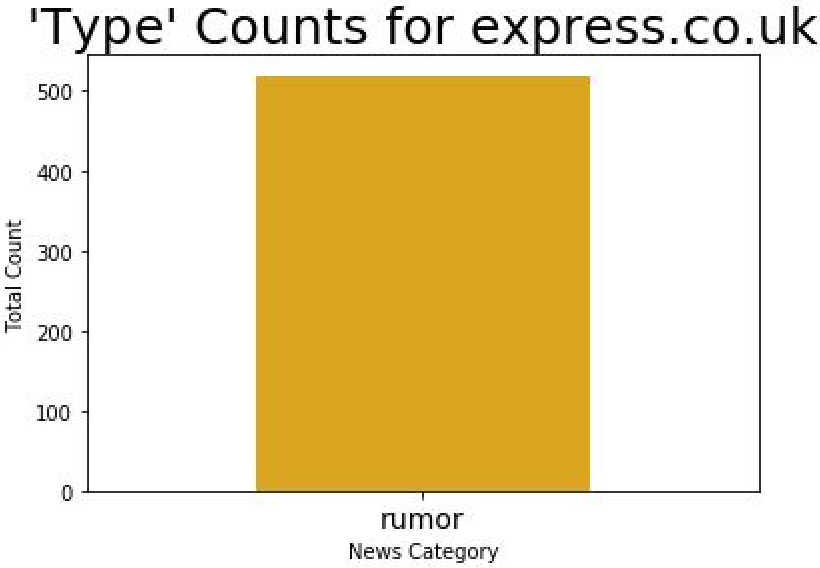
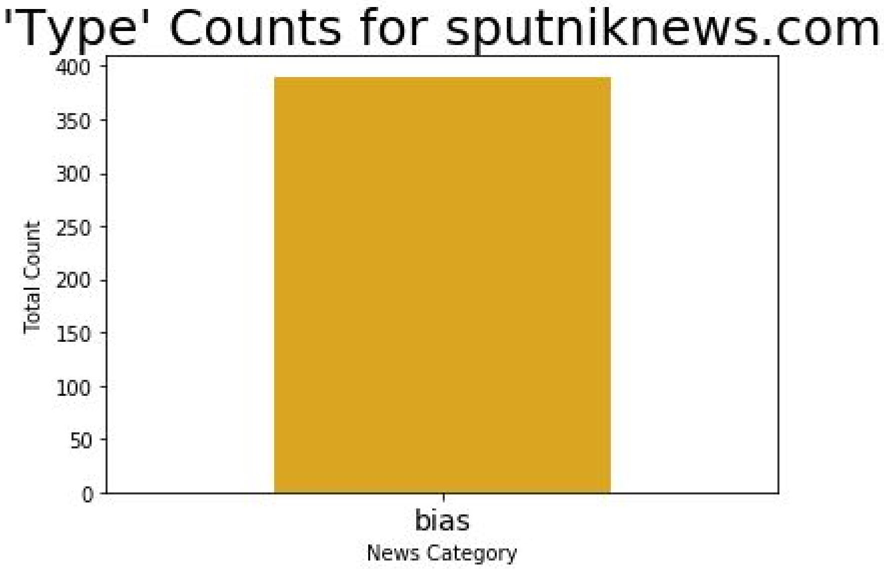

'Type' Counts for express.co.uk
Total Count
0
100
200
300
400
500
rumor
News Category

Fig. 7



Final sampling

Due to the inherent biases in the dataset for the labelled categories ‘junksci’, ‘bias’, etc., it was decided to resample the dataset by including data from the two categories ‘fake’ and ‘reliable’. Although there are issues with designating all articles from certain sources with the labels ‘fake’ or ‘reliable’, the sources used with these labels seem to be more consistent than with the other labels -- nytimes.com is, for the most part, reliable, whereas, a major website represented in the data, beforeitsnews.com, is not.

EDA of Sampled Data

The sampled data contains 10,000 articles labelled ‘reliable’ and 10,000 labelled ‘fake’. The majority of these 20,000 articles were from nytimes.com (labelled as ‘reliable’) and beforeitsnews.com (labelled as ‘fake’). There were, however, 141 difference online news sources represented in the data (Fig. 9).

Fig 8.

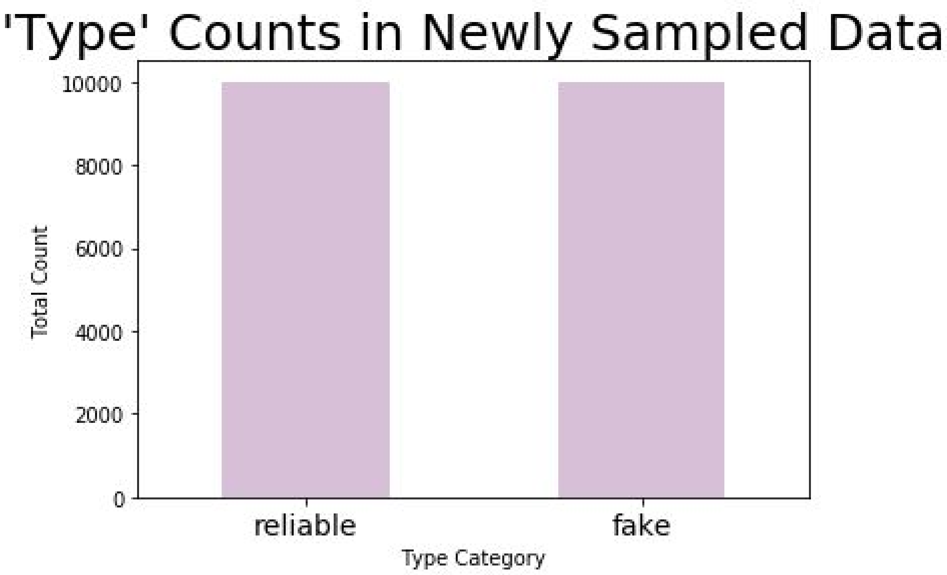
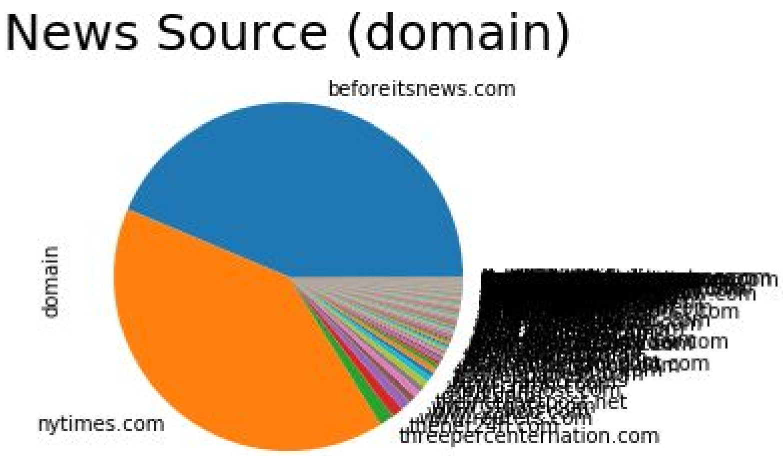


Fig. 9



Predictive Modeling

The data was divided into a training and testing set. A multinomial naive Bayes classifier was trained using sckit-learn, and then used to predict the labels for the testing data. Accuracy varied depending on vectorization approach:

Bag-of-words

Predictions were 88.1% accurate

Tf-idf

Predictions were 88.9% accurate

Tf-idf with two bigrams

Predictions were 92.2% accurate.

There may be some issues with this model, linked with the data itself. Inspecting the most predictive features revealed, for example, that some of the most predictive bigrams for classification were ‘york time’ and ‘york citi’. This is due to the over-representation of The New York Times as news labelled as ‘reliable’ in the dataset.