STEM Fellowship Big Data Challenge 2021

Using Machine Learning Methods To Predict Fake News & Exploring Vaccine Manufacturer Sentiment

Raymond Romaniuk[1]

[1]Brock University

May 31, 2021

Abstract

With 2.5 quintillion bytes of data created each day [1], the spread of misinformation is more relevant than ever. Big data requires big responsibility and we must do our best to slow the spread of false claims that may turn to public health risks.

Our paper aims to use machine learning methods to accurately classify fake news headlines to stop the spread of potentially dangerous health narratives. In addition to this we explore the sentiment shown towards COVID-19 vaccine manufacturers Pfizer, Moderna and AstraZeneca.

We use two datasets for our analysis. The first containing news headlines shared on social media and labelled as real and fake news. The second containing vaccine related tweets.

We use machine learning methods, like logistic regression and support vector machines, to attempt to classify news headlines as real or fake news. Additionally, we explore the uses of clustering algorithms and neural networks. To determine the sentiment of our vaccine related tweets we use Python's Natural Language Toolkit. With our estimated sentiments we fit time series models to forecast future public sentiment.

We found that the Kernel Support Vector Machine was most accurately able to predict whether the news headline provided was real or fake news. It's accuracy was substantially higher than any of our other models at 73.46%.

We also observed that our sentiment analysis of vaccine related tweets successfully identified the impact of negative news stories on a manufacturers public sentiment. For example, the reports that the AstraZeneca vaccine may cause blood clots [2] visibly increased AstraZeneca's negative sentiment.

The fake news classifier we explored may be a helpful addition to large social media platforms like Facebook, Twitter and Instagram. The development of an accurate model could limit the spread of potentially dangerous false health claims and create a safer social media environment for all.

Keywords
machine learning, sentiment analysis, time series analysis, classification, vaccine

1 Introduction

Social media is a staple of many individuals daily lives. However, with the many advantages that come from a globally connected society, there are also many negative impacts from it. One of them being the instantaneous spread of fake news. Studies show that it "takes true stories about six times as long to reach 1,500 people as it does for false stories to reach the same number of people" [3]. With approximately 456000 tweets sent every minute [1] the spread of fake news seems inevitable.

What if we could create a model that effectively classifies fake news headlines to limit their spread? That is exactly what we will attempt to do. We will use machine learning and explore which model provides the most accurate classifications. As we will learn this can be quite difficult, but we will find that one of our models significantly outperforms our others.

With the COVID-19 vaccine currently being rolled out across Canada we will also explore how people feel about receiving it. With vaccine manufacturers Pfizer, Moderna and AstraZeneca becoming household names we will perform sentiment analysis on tweets related to them to gauge their public perceptions. Doing so will allow us to gather insight on how the fake news stories, that we are attempting to mitigate, may impact the sentiment towards these manufacturers. We will analyze the manufacturer sentiment at a monthly and daily level, and attempt to forecast future public sentiment.

Our research has the potential to change the social media landscape for good and protect those who are susceptible to believing false news reports.

2 Materials & Methods

The data used for the statistical analysis performed in this paper comes from two sources:

1. The GitHub for the paper *A Heuristic-driven Ensemble Framework for COVID-19 Fake News Detection* [4]

2. *All COVID-19 Vaccines Tweets* dataset by Gabreil Preda, from Kaggle [5]

The dataset corresponding to the paper *A Heuristic-driven Ensemble Framework for COVID-19 Fake News Detection* contains article headlines that have been classified as "real" or "fake". There are both training and testing datasets, with the training dataset containing 6420 labelled observations and the testing dataset containing 2140 labelled observations.

The second dataset we will use contains real tweets, obtained through Twitter's API, about the prominent COVID-19 vaccine manufacturers (Pfizer, Moderna, AstraZeneca, etc.). Overall there are 78319 tweets, with each tweet also having information about the user, the number of retweets and the date of the tweet. These tweets range from December 12, 2020 to May 21, 2021.

Python was used for all of the statistical analysis of this paper, while Microsoft's Power BI was used to create the visualizations. The notable Python packages that were used include: scikit-learn (sklearn) to create the models for our article headline classification, TensorFlow to create a neural network, Natural Language Toolkit (NLTK) to analyze the sentiment of the vaccine tweets, and statsmodels to perform our time series analysis.

We used a variety of classification models to try to obtain the highest prediction accuracy. These models include: Naive Bayes, Logistic Regression, K-Nearest Neighbours (KNN), Kernel Support Vector Machines (SVM) and XGBoost. Additionally we also explored the use of clustering methods, like K-Means and Hierarchical, and fit an Artificial Neural Network.

Subsequently, we split the vaccine tweet data by vaccine manufacturer and obtained datasets for Pfizer, Moderna and AstraZeneca. Next, the tweets for each manufacturer were organized by date.

Using Python's NLTK package, sentiment analysis was performed on each tweet and they were classified as having a negative sentiment, positive sentiment, or neutral sentiment. The proportion of each sentiment was then calculated for each day.

Once sentiment proportions were gathered for each day we used the statsmodels package to perform time series analysis and try to accurately forecast future sentiment. Time Series methods we used include: Autoregression, Moving Average, Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA), Simple Exponential Smoothing and Holt-Winters Exponential Smoothing.

Finally, Power BI was used to visualize the data and allow readers to visualize our predictions.

3 Results

Table 1 shows the accuracy of each of our fitted and tuned machine learning models when classifying a news headline as real or fake news.

Table 1: Prediction Accuracy By Model

Model	Accuracy
Kernel Support Vector Machine	73.46%
Artificial Neural Network	56.03%
Logistic Regression	52.76%
Linear Support Vector Machine	52.52%
Bernoulli Naive Bayes	51.59%
XGBoost	50.70%
K-Means Clustering	47.71%
Hierarchical Clustering	47.71%
K-Nearest Neighbours	47.66%
Complement Naive Bayes	44.95%
Gaussian Naive Bayes	43.36%

Next, Table 2 shows the estimated proportion of positive, negative and neutral sentiment for the Pfizer, Moderna and AstraZeneca vaccines by month.

Additionally Figures 1 and 2 show our forecasted positive and negative proportions, for each of the vaccine manufacturers, plotted with the true proportion on each day.

Table 2: Monthly Vaccine Sentiment

Manufacturer	Month	Pos	Neg	Neu
Pfizer	Dec	38.1%	13.2%	48.7%
	Jan	38.8%	21.0%	40.2%
	Feb	35.8%	17.8%	46.4%
	Mar	37.4%	15.4%	47.2%
	Apr	32.3%	17.5%	50.2%
	May	37.2%	14.9%	47.9%
Moderna	Dec	48.0%	8.8%	43.1%
	Jan	34.2%	15.8%	50.0%
	Feb	35.9%	17.4%	46.7%
	Mar	37.5%	15.2%	47.3%
	Apr	36.1%	16.7%	47.2%
	May	36.8%	17.2%	46.0%
AstraZeneca	Dec	40.4%	5.8%	53.9%
	Jan	29.0%	18.7%	52.3%
	Feb	33.7%	21.9%	44.5%
	Mar	36.5%	23.5%	39.9%
	Apr	28.1%	28.2%	43.7%
	May	31.2%	17.9%	50.9%

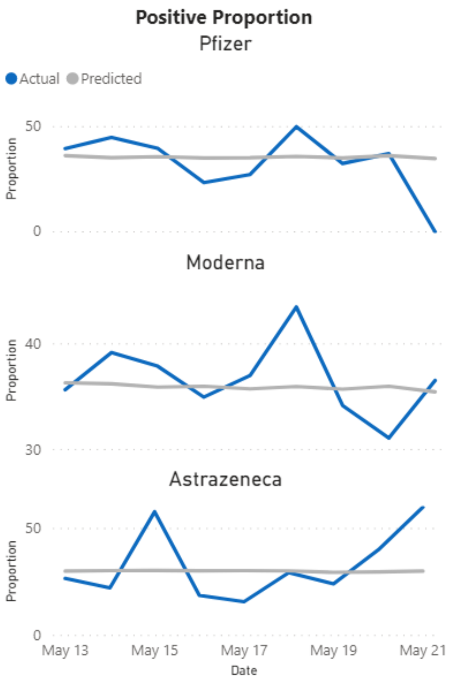

Figure 1: Forecasted positive proportion for each manufacturer plotted with the true proportion.

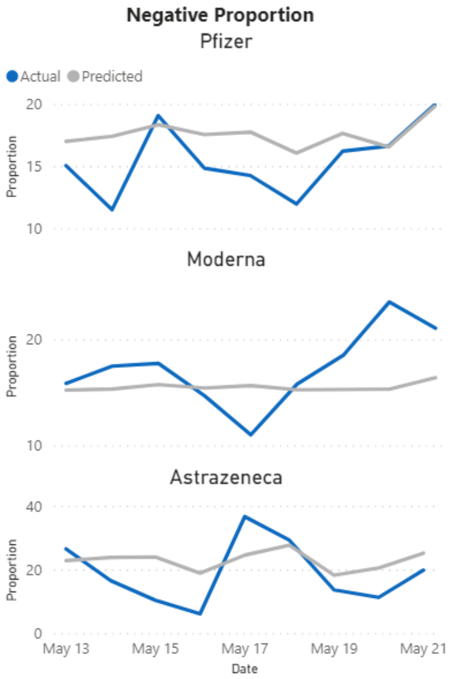

Figure 2: Forecasted negative proportion for each manufacturer plotted with the true proportion.

4 Discussion

With more information available on social media than any one person could consume, it is inevitable that plenty of it will be misleading and potentially detrimental to the individuals consuming it. From the dangerous suggestions made by US President Donald Trump to inject disinfectant as a COVID-19 treatment [6], to the potential for prominent influencers to spread misinformation [7]. There is potentially harmful "fake news" on all social media platforms.

Since fake news stories are 70% likelier to be retweeted on Twitter than real news stories [3], the development of a model to screen articles, that could accurately predict whether an article was factual or not, would be extremely beneficial.

As one would assume, using natural language processing to classify the headline of a news article, as real or fake news, can be quite challenging. In Table 1 we see that almost half (approximately 45.5%) of the techniques we used returned a prediction accuracy, on our test data, of less than 50%. The classification models K-Nearest Neighbours, Complement Naive Bayes and Gaussian Naive Bayes were all ineffective and we would receive more accurate predictions by predicting the opposite of what these models do.

For our natural language processing we use the bag-of-words model, where each unique word in our headline dataset is its own independent variable with its value being the number of times it appears in a given headline. Table 3 displays an example of how the bag-of-words model works.

Table 3: Bag-of-Words Model Example

Headline	COVID	Is	Very	Bad
COVID Is Very Bad	1	1	1	1
Very, Very Bad	0	0	2	1

With this bag-of-words model our K-Means and Hierarchical Clustering algorithms classified each observation in our testing data, except for one, as fake news, thus also providing sub 50% prediction accuracies. For our purposes clustering algorithms seem to be ineffective at differentiating between real and fake news.

The six machine learning methods that led to prediction accuracies greater than 50% are: the Kernel Support Vector Machine, Logistic Regression, Linear Support Vector Machine, Artificial Neural Network, Bernoulli Naive Bayes and XGBoost. Each of these methods returned an accuracy between 50 and 53 percent, except for the Kernel Support Vector Machine and Artificial Neural. Of these four methods, with mid-level accuracy, the Logistic Regression returned the highest proportion of correct predictions at 52.76%.

Logistic Regression is a machine learning model that, using the logistic function (Figure 3), bounds predictions between zero and one. These predictions can be used as the probability that the article headline in question is real news. We will classify those with a predicted probability greater than 50% as real news.

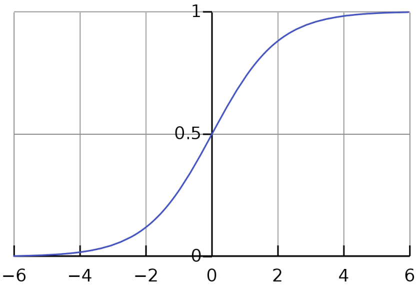

Figure 3: Plot of the logistic function.

Our second most accurate machine learning method is the Artificial Neural Network, which falls into a subset of machine learning called deep learning. Deep learning is "a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input" [8].

The Artificial Neural Network that we found provided the greatest prediction accuracy, of 56.03%, contained two hidden layers with 50 nodes per layer and using the Softmax activation function. Figure 4 shows an example of an Artificial Neural Network with two hidden layers, similar to ours, but with only four nodes per layer.

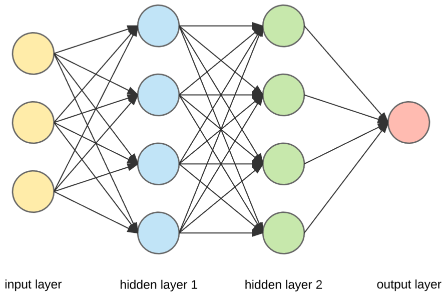

Figure 4: Example of an Artificial Neural Network with two hidden layers and four nodes per layer.

The Softmax activation function is a generalization of the logistic function, that we previously introduced, in multiple dimensions [9]. Since the Logistic Regression model is already one of our most accurate classifiers, it is not surprising that using a variation of it in our Artificial Neural Network provides increased prediction accuracy.

Worth noting, our tuned Artificial Neural Network used a batch size of 30 and 100 epochs for training on the training data. The batch size is the "total number of training examples present in a single batch" and an epoch is the number of times we wish the entire training dataset to be "passed forward and backward though the neural network" where the weights connecting each node are adjusted each time [10].

Saving the most accurate model for last, we have a prediction accuracy of 73.46% for our Kernel Support Vector Machine model. This is a 31% increase in accuracy from our Artificial Neural Network.

The Kernel Support Vector Machine can be used for regression and, in our case, classification aiming to create a hyperplane that optimally splits the data while maximizing, what is called, the margin (Figure 5 [11]).

The introduction of a kernel function maps the data points into higher dimensional feature spaces [12] and allows for computationally cheaper calculations. Figure 6 [13] shows an example of how the selected kernel works with the hyperplane to classify observations.

We tested multiple kernel functions and found

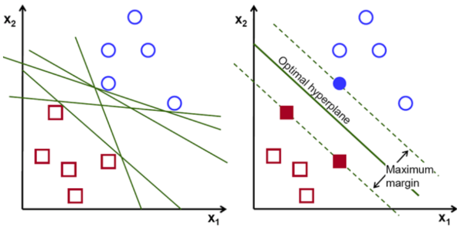

Figure 5: Example of how the optimal hyperplane is selected, in two-dimensions, by maximizing the margin.

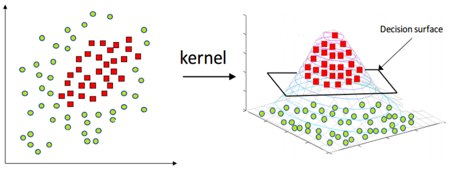

Figure 6: Example of how a specified kernel interacts with a hyperplane to classify observations.

that, overall, the Radial Basis Function (RBF) provided the most accurate predictions. In addition to option of selecting the kernel function, the Support Vector Classification (SVC) class, in Python, also included a regularization parameter that we tuned to maximize the prediction accuracy. Figure 7 shows our Kernel SVM's prediction accuracy for regularization parameters between zero and one, while Figure 8 focuses on where our prediction accuracy is maximized.

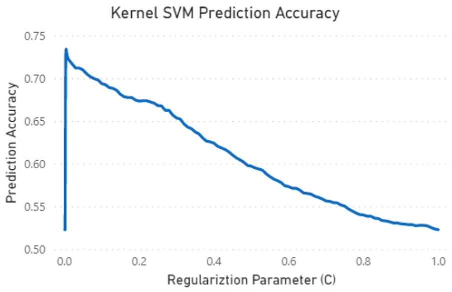

Figure 7: Plot of the Kernel Support Vector Machine's prediction accuracy with increasing regularization parameter, in the range zero to one.

From Figures 7 and 8 we see that our prediction accuracy is maximized when the regularization parameter is 0.005.

Since we are using previously built natural language processing libraries there is the possibility that our data pre-processing may not be the most effective for our purposes. In future implementations more testing of the libraries and potentially some manually created functions may benefit our predictions.

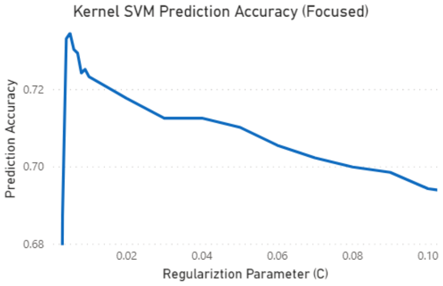

Figure 8: Plot of the Kernel Support Vector Machine's prediction accuracy, with focus on the maximum value.

Now that we've explored the creation of a machine learning model to predict whether a news article headline is real or fake news, we felt it would be interesting to also investigate the sentiment towards the COVID-19 vaccine. Instead of using our COVID-19 social media news headlines for this, we will use our vaccine related tweets dataset.

First, we split our tweets dataset, containing 78319 vaccine related tweets, by vaccine manufacturer. Doing so we obtain 9703 tweets for Pfizer, 19474 tweets for Moderna and 4878 tweets for AstraZeneca. We then additionally split these three datasets by month and by day.

With all the tweets sorted by vaccine manufacturer and date we then performed sentiment analysis on each tweet. This was accomplished using Python's Natural Language Toolkit to score the probability that a given tweet had a positive, negative or neutral sentiment.

Figure 9 shows the change in the proportion of positive and negative sentiment, over the period December 12 to May 21, by month.

We see that, in the case of Pfizer and Moderna, the proportion of tweets with a positive sentiment are almost double those with negative sentiment, for the entire duration. However, this is not the case for AstraZeneca. The number of positive and negative tweets, per month, has a much smaller difference, with the proportion of negative tweets, even, slightly surpassing the proportion of positive tweets in the month of April.

The increase in negative tweets towards AstraZeneca is likely due to the reports of blood clots developing in individuals that had received the vaccine [2]. This situation being clearly visible in our sentiment classifications is a positive indication that our natural language processing methods are performing as expected.

Next, we moved on to performing time series analysis on the daily sentiment for the three manufacturers. Table 4 shows the mean squared error of the time series models that we tested,

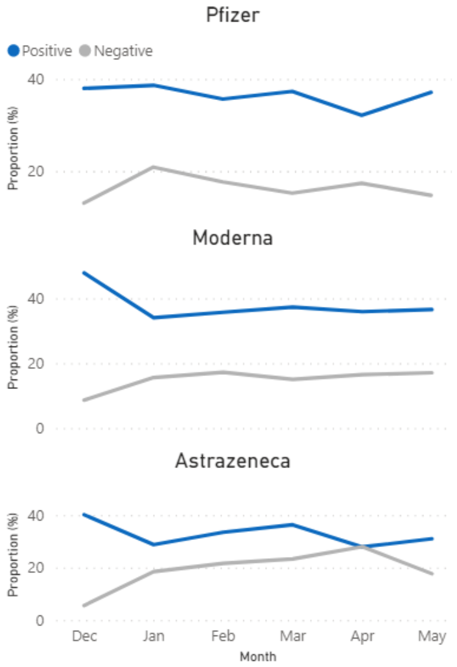

Figure 9: Proportion of positive and negative sentiment for each vaccine manufacturer, by month.

when predicting the last nine days of our testing data.

Table 4: Time Series Model Mean Squared Error's

Model	MSE
Autoregression	451.77
Autoregressive Moving Average	461.06
Moving Average	479.91
Holt-Winters Exponential Smoothing	500.72
Simple Exponential Smoothing	500.75
Autoregressive Integrated Moving Average	6535.22

We find that the Autoregression, Autoregressive Moving Average and Moving Average time series models are our three most accurate, in terms of mean squared error. Mean squared error is calculated as the difference in each prediction from the true value, squared and divided by the total number of observations. The mean squared error is written as:

$$MSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y_i})^2}{n}$$

Autoregression, our most accurate time series model, uses "observations from previous time steps as input to a regression equation to predict the value at the next time step" [14]. Unlike our previous machine learning models, the only input for our model is the sentiment of the prior days. This model does not take into account variables like the day of the week or seasonality. We found that the optimal number of lags in our model was one, meaning that our model uses the sentiment proportion of the previous day to predict today's sentiment proportion.

Our third most accurate time series model, the Moving Average, observes the error of our previous predictions and, each day, adjusts the prediction to continue to minimize our future errors.

The Autoregressive Moving Average model combines the Autoregression and Moving Average models. It takes into account the previously observed value in the series, like in Autoregression, and the previously calculated prediction errors, like in the Moving Average model.

A notable issue that is likely causing error in our predictions is that our dataset does not contain Pfizer, Moderna and AstraZeneca tweets from April 23 to May 12. Thus, when we make our predictions for the last nine days of our available data (May 13 to May 21), we may not be receiving the most accurate predictions due to having almost three weeks of unavailable data.

Ideally we would have preferred to have a complete dataset without any missing data, however we were unable to obtain a Twitter Developer account in time to create our own curated dataset of vaccine tweets.

Additionally, the mean squared error obtained by the Autoregressive Integrated Moving Average (ARIMA) model is significantly higher than any of the other models. The exact reason for this is unknown and we were unable to find any issues within the Python code that was used to develop it.

Conclusions

The significant accuracy obtained using the Kernel Support Vector Machine to predict whether a news headline is real or fake news may be an impactful feature in future social media updates. With more posts being uploaded daily to social media than any group of individuals could monitor, the development of a machine learning model that could accurately classify news articles as real or fake news would be a massive achievement. This model would stop the incredibly quick spread of fake news and protect individuals from potentially dangerous health and safety related claims.

A fake news classifier model, like the ones we tested, could be a great idea for social media giants like Facebook, Twitter and Instagram to implement. The one caveat would be whether it would be possible to create an accurate enough model.

As for our analysis of the vaccine manufacturer sentiment, it would be extremely interesting to explore how news reports related to each of them effect their sentiment scores. An example would be how AstraZeneca sentiment shifted with the reports of vaccinated individuals developing blood clots. It would be intriguing to explore how impactful a positive news story is on a companies public sentiment relative to the impact of a negative story.

More testing in the future may be required to determine how accurately each tweets sentiment is being scored.

Finally our time series analysis may benefit from a more well rounded dataset set without missing values and a greater number of tweets per manufacturer. Obtaining this more well rounded dataset is something that we plan to explore in future iterations of this research topic.

Acknowledgements

I'd just like to thank the STEM Fellowship for putting together this competition allowing me to sink my teeth into some real world big data, and try some new machine learning methods in the process.

References

[1] Bernard Marr. How much data do we create every day? the mind-blowing stats everyone should read, Sep 2019.

[2] Adam Miller. Future of astrazeneca covid-19 vaccine in question in canada over blood clots, supply issues | cbc news, May 2021.

[3] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online, 2018.

[4] Sourya Dipta Das, Ayan Basak, and Saikat Dutta. A heuristic-driven ensemble framework for covid-19 fake news detection. *arXiv preprint arXiv:2101.03545*, 2021.

[5] Gabriel Preda. All covid-19 vaccines tweets, 2021. https://www.kaggle.com/gpreda/all-covid19-vaccines-tweets.

[6] Coronavirus: Outcry after trump suggests injecting disinfectant as treatment, Apr 2020.

[7] John Leicester. Influencers say they got offered thousands to spread fake news on pfizer covid-19 vaccine, May 2021.

[8] Wikipedia. Deep learning, May 2021.

[9] Wikipedia. Softmax function, May 2021.

[10] Sagar Sharma. Epoch vs batch size vs iterations, Mar 2019.

[11] Rohith Gandhi. Support vector machine - introduction to machine learning algorithms, Jul 2018.

[12] Wikipedia. Support-vector machine, May 2021.

[13] Grace Zhang. What is the kernel trick? why is it important?, Nov 2018.

[14] Jason Brownlee. Autoregression models for time series forecasting with python, Aug 2020.