Navigating Narratives: Uncovering the impact of news articles on stock movements

Ian Shin

May, 13 2020

Abstract. This report will unravel the influence and impact of narratives upon Altria's stock. By discovering the true nature of narratives, the applications could be endless, and many fields of study would need to be reviewed. By utilizing web-scraping, sentimental analysis, and machine learning, this report will be able to find the underlying relationship that exists between narratives and stocks.

1 Introduction

In 1637, the Dutch fell victim to one of humanities worst bubbles,which resulted in reputations to be destroyed, judicial reforms to occur, and thousands to incur enormous financial loss. This event would be later named "The Tulip Mania" and would eventually lead to a tulip's value to exceed the price of a house. Many believe that it was simply an imbalance amongst demand and supply that caused this bubble. However, considering the climate in Europe during this time and the fact that tulips are relatively easy to cultivate, this claim seems false. It was the accepted idea that tulips were an exotic luxury that over inflated their value. Narratives, or stories, have existed since the beginning of civilization. Spanning from Greek/Nordic mythology, to Y2K, and even today with the outbreak of COVID-19, narratives have influenced societies, economies, and even decision making. With the introduction of social media, high-speed internet, and online news stations, narratives have more power than ever. This project will discover the relationship between the polarity of news articles, and their impact on stock prices. I decided to use Altria (MO) as my test case for several reasons. First, it was based in the United States and is the

second largest tobacco company in the US. Also, compared to it's competitor Philip Morris, Altria has been public since 1970 while Philip Morris went public in 2008. Furthermore, Altria's 35 percent stake in JUUL Labs provided a very rare occurence. JUUl is the dominant electronic E-Cigarette manufacturer, and have been under fire by the FDA, politician and news outlets for targeting under-age teenagers. Tons of anti-JUUL advertisements were released, news reports of JUUL users hospitalized were published, health regulations and state laws were filed which led to the banning of many JUUL products in seven states. Health regulations on consumer goods does not occur very often; and using Altria which has a large share in such a controversial company seemed like a perfect fit. Lastly, it is often said that the tobacco industry is resistant to recession and economic downfalls. This can be displayed during the 2008 housing bubble, where the tobacco industry outperformed major economic staples such as food, cleaning/beauty products. Possessing the characteristic of withstanding recessions, allowed my analysis of Altria's stock movement more streamlined.

2 Method

My approach comprised of three stages: Data collection, sentiment analysis, and modelling.

3 Data Collection

The news articles I have obtained were all from The Motley Fool (a reputable financial and investing advice company). I opted for Motley Fool for several reasons.Due to their reputation amongst the financial reporting sector, I believed that the majority of their audience are investors and have the ability to interact with the market. Furthermore Motley Fool's articles, are written by their employees, and not outsourced from other websites. For example, Yahoo Finance has articles from several other news outlets, which would increase the bias (some news outlets are very conservative, etc), broadens the audience, and can contain headlines that are more sarcastic in nature(this poses a problem for sentiment analysis) The Motley Fool contains many dynamic web elements, and in order to load older news articles a "Load More" button must be clicked. This is why I turned to Selenium. Due to its ability for web automation and dynamic scraping, Selenium was the perfect choice. Paired with BeautifulSoup I was able to load 40 pages, and extract 217 articles ranging from the year 2017 to 2020. During this stage, I ran into a hurdle. Due to hardware limitations, I was not able to obtain the content of each article title (especially at the volume of articles I wanted). I tried NewsAPI, to streamline the data collection process, however their free account only gave me three months of prior news, and

truncated the article contents to 200 characters. On top of this, I would only get about 100 articles, and this is processing the data. Therefore, I decided that a larger data set, with slightly less content was the better option.

4 Sentiment Analysis

Natural Language Processing has been the topic of discussion recently. Sentiment analysis, in particular has gained tremendous attention due to the astronomical growth of social media platforms. For this project, I decided to use VADER (Valence Aware Dictionary and sEntimental Reasoner). It is a "lexicon and rule-based sentiment analysis tool" which has been designed to handle sentiments from social media. It outputs four sentiments. 1.) Positive 2.) Neutral 3.) Negatives 4.) Compound. The first three values are represented with a value between 0-1, which symbolizes the ratio of text that sentiment contains.The compound output is computed by summing the scores of each word in the lexicon, and normalized between a range of -1 to 1. This column is the single most important value, and the only column I used for my regressions. Also, an assumption that I made,was to simply dismiss purely neutral sentiments. I simply then fed the news articles into this sentiment analyzer, and created a Panda's dataframe with columns "Dates", and "Aggr. Title Polarity". I then downloaded the Altria historical data from Yahoo Finance, and used Pandas to open and delete all columns except the closing price. I grouped the values by date, and deleted weekend dates (the market closes on weekends). After this was all done, I could finally move on to the modelling stage.

5 Modelling

For this stage, I had to first consider the nature of my variables. The polarity of each words is considered a limited latent dependent variable (due to the fact that they are restricted by values between -1 and 1) and is also categorical (they can either possess a negative or positive sentiment). This also implies that it is nonlinear, and a simple linear regression would not work. The next step I took was to determine if the polarity of the sentences possessed negative or positive sentiments. Due to the lack of "labels", an unsupervised learning algorithm was need; which is why I turned to K-means. This would allow me to cluster my data, and label my data into two classes. 0: Negative Sentiment and 1: Positive Sentiment This allowed me to move forward with supervised learning. Logistical Regression seemed like the obvious choice for these parameters. Initially, when viewing the Logistical Regression it was promising. As the value of the stock decreased, we can see a increase that a new article with negative sentiment was published. For example, if the stock price

was hovering around the 40 dollar mark, there would be a 60 percent likelihood, a negative article was published.

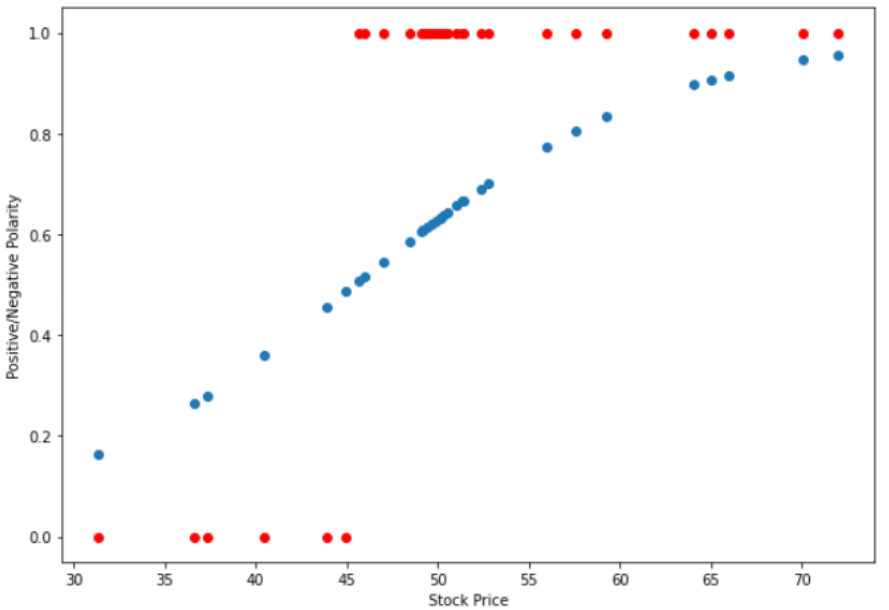

Figure 1: Logistical Regression

I also explored the marginal effect or partial dependence between the features and predicted value. This model would help explain the effects between the two.Shown in Figure 2, it is clear that as the price of the stock increase, the polarity of the news article increases as well (in a sigmoidal fashion)

However, when analyzing the logistical regression I was not satisfied by the results. Using sklearns "Classification Report" I was able to access the precision, recall, f1-score and support of the logistical regression. My model's overall performance was very poor. The overall correctly classification for both the 1 0 classes were under .5. Although,it seems like the "1 class" performed better when it comes to identifying "true positives" and therefore implies a larger F1-Score.This could be possibly due to the fact that the "1 class" has more instances.
However, I realized that as I changed the support column for each class (by manipulating the ""random state" of the train test split function) all the values of the report increases. This leads me to believe that it is actually the volume of data I am feeding

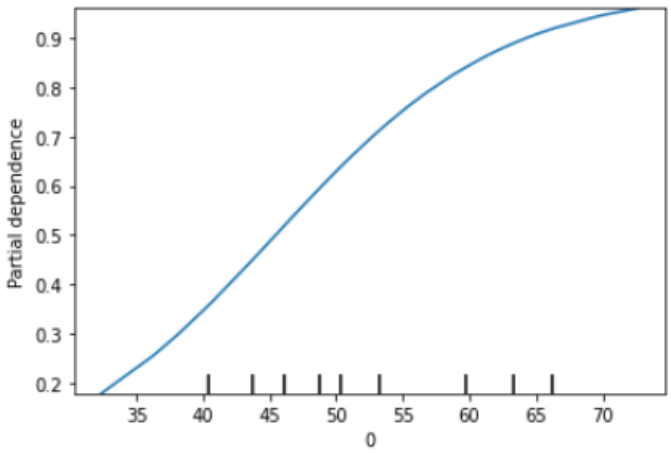

Figure 2: Partial Dependence

the model, that is creating such poor results. The reason I stuck with this particular "random state" shown in Figure 2. was due to the fact that the support amongst the classes were not imbalanced.

	precision	recall	f1-score	support
0	0.33	0.12	0.18	16
1	0.48	0.76	0.59	17
accuracy			0.45	33
macro avg	0.41	0.44	0.39	33
weighted avg	0.41	0.45	0.39	33

Figure 3: Classification Report

Figure 4. displays the confusion matrix. With "True Negative" = 2 "False Negative" = 14 "False Positive" = 4 and "True Positive" = 13. It is clear that this model performed poorly when classifying negative values.

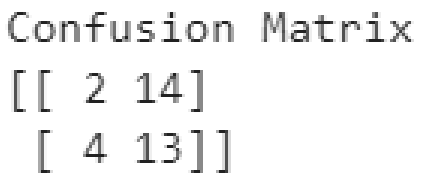

Figure 4: Confusion Matrix

6 Gaussian Process Regression

The logistical regression, although insightful, was not fulfilling enough. After hours of searching the web I discovered Gaussian Process Regression (GPR) and saw potential in it.It is a probabilistic model, that is non parametric due to the fact that for each observation, there is a latent variable that is considered. This works well with my project, due to the fact that the polarity scores are latent variables. (They are inferred not observed). Furthermore, GPR works well with smaller data sets. GPR follows a Bayesian approach, by utilizing prior distribution and Bayes' rule, and uses this information to make predictions. There are many GPR packages available for python, however I chose to stick with Sklearn.
The next step, was to decided what covariance kernel function to use. The Matern kernel seemed like the best fitting function. It is similar to Radial Basis Function, however generalized and has an additional parameter ν, which can allowed me to manipulate the "smoothness" of the function. After finding the proper kernel, I simply fed this information into the model, as well as fit my training/testing data. Figure 5 displays the prediction that was outputted from the GPR model.It is extremely volatile, particularly around the -.02 region. I have used other kernels, and was able to smooth out the model (i.e. WhiteKernel), however doing so would compromise the coefficient of determination. With the current model, we achieve a score of .70 which represents that most of the variation in the response variable hovers near the mean, and that there is a correlation between the two values.

7 Conclusion

In the beginning I stated that I would uncover the correlation between the movements of Altria's stocks and the sentiment of news articles. Though the models above, do

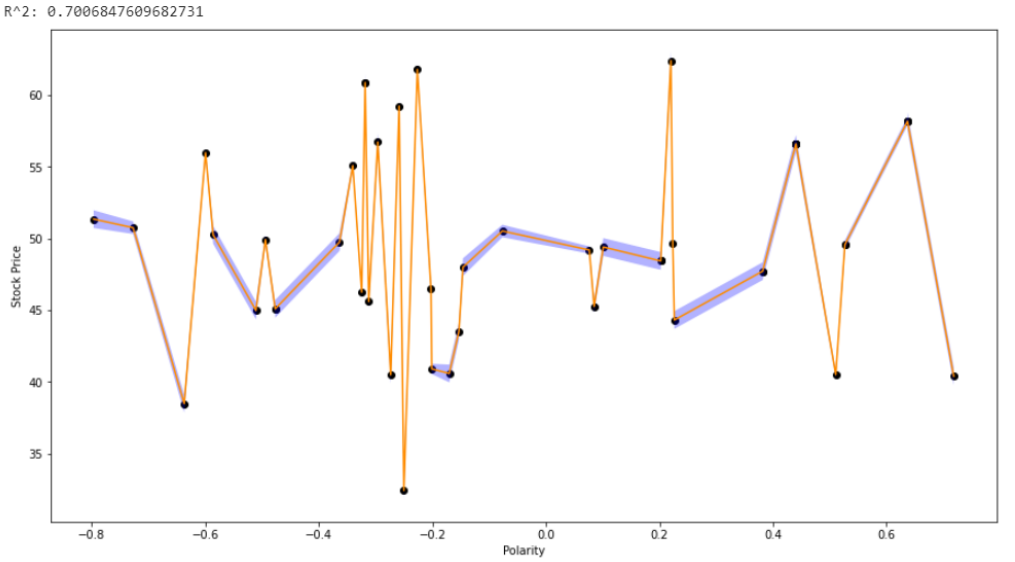

Figure 5: Gaussian Process Regression

indicated some positive relationships between the two, it is difficult to truly conclude this. In economics, in order for a majority of theories to hold we must accept the axiom that all people are rational. However, this is not true. Even when news articles are extremely negative, we can see an increase in stock prices. This can be due to contrarian investors who are "characterized by purchasing and selling in contrast to the prevailing sentiment of the time"; or even investors who view the tobacco industry as "fail proof" and buy when prices are low could have altered the results. I believe that if I were to analyze narratives, and financial benchmarks (such as cash flow, P/E ratios) that are used to value a stock, as a supplement to this project; I would yield fruitful results.