Sentiment Analysis for Financial News

Harrison DiStefano, Jaehee Jeong, Lisa Kaunitz, Junghwan Park, David Sun
STAT 418 | Winter 22 | Professor Mendler

Overview

Executive Summary

The purpose of this project was to create an end-to-end solution of the observed associations between headline sentiments, and abnormal market returns and volatility. While our data did not contain specific market metrics, the literature states there is a statistically significant association between volatility seen in the markets, tied with the sentiments of financial news headlines[1]. With this relationship in mind, we used our dataset of 4,845 financial news headlines with predetermined sentiment labels to formulate our project goals:

1. Build a model that predicts news sentiments which may affect changes in the market.
2. Implement an app that can take in a potential news headline and output the predicted sentiment.

Our dataset was obtained from Kaggle and imported via .csv file. For the data preprocessing, we used both R and Python to remove stopwords, non-alphanumeric characters, punctuations, and stemming words. For the modeling, we utilized LSTM, GRU, BERT Sentimental Analysis and BERT Fine-Tuning. The desired model was BERT Fine-Tuning with eight epochs, 0.43 loss, and 84.6% accuracy.

Descriptive Analytics - EDA

Exploratory Data Analysis

Before the modeling, we explored data using R as it is a perfect tool to tokenize and "tidy" the text. We first had to clean the data using the R package called "tm" so we could remove all non-alphanumeric characters, punctuation, English stopwords, and stemming words. We also converted all words to lowercase and cleaned all strip whitespaces. After the process, we could explore the data at a glance by plotting word clouds for each of the three categories. As the second most important currency in the world, "EUR" was the most prominent word over the three categories and the word "profit" appeared a lot in negative headlines. After we make tibble in R to tokenize each headline using the "tidytext" package, we were prepared to conduct sentiment analysis.

[1] Wan, Xingchen, et al. "Sentiment Correlation in Financial News Networks and Associated Market Movements." *Nature News*, Nature Publishing Group, 4 Feb. 2021, https://www.nature.com/articles/s41598-021-82338-6.

Figure 1: word cloud for positive (left), neutral (middle), and negative (right)

We examined how “NRC emotions” fall into each category using the “Syuzhet” package. As can be seen in the figure, the whole tokens for headlines are composed of about 80 percent of positive words and 20 percent of negative words, while trust and anticipation words were taking most part of nrc sentiments.

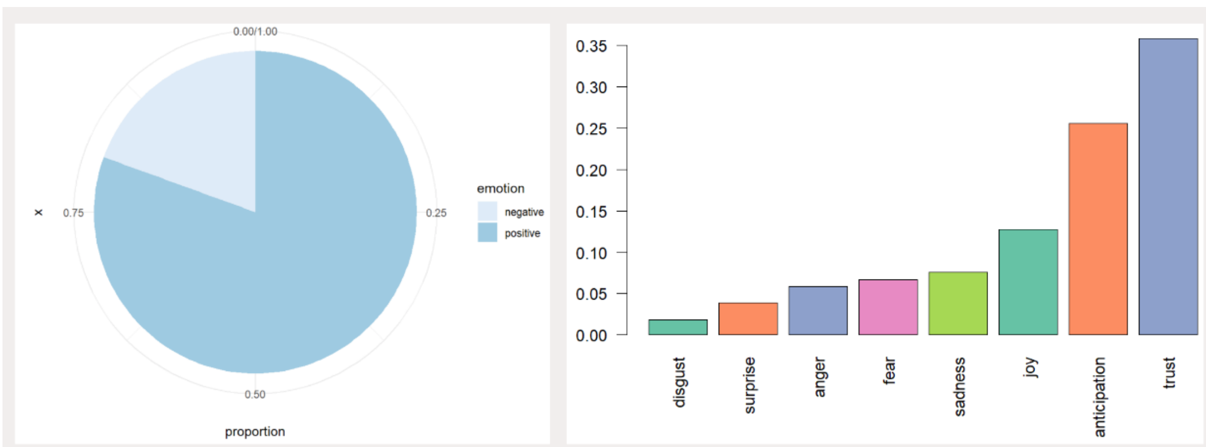

Figure 2: NRC sentiments for the whole headline data

We also conducted sentiment analysis for each category (positive, negative, and neutral). It was found that the composition of emotions were quite similar within positive and neutral categories, while the tokenized headlines in a negative category consisted of more anger, fear, sadness. It was interesting to find out that the positive sentiments such as joy and anticipation were taking more parts than those in positive or neutral, and we attribute this to negations.

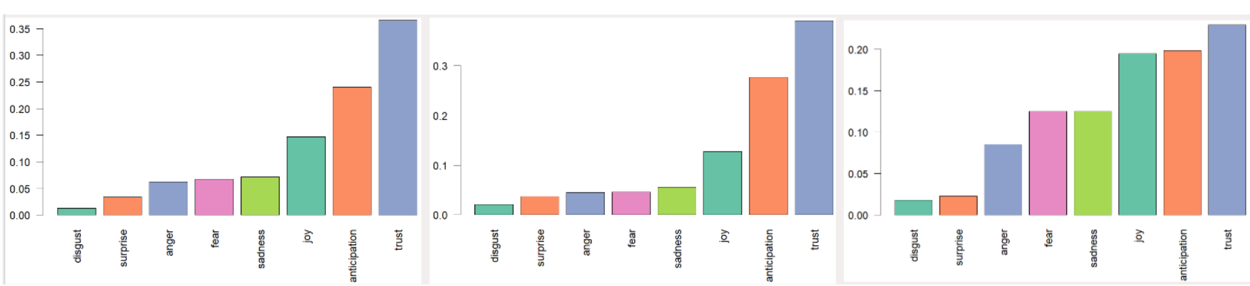

Figure 3: nrc sentiments for positive (left), neutral (middle), and negative (right)

Analysis

Models

We decided to use four models: LSTM (Long Short Term Memory), GRU (Gated Recurrent Unit), BERT(Bidirectional Encoder Representations from Transformers) Sentimental Analysis, and BERT Fine-Tuning. Before we started, we cleaned all the punctuations and eliminated non-English terms to achieve better accuracy for each model. We then tokenized and padded the texts, encoded the classification results, and splited the data as training and testing datasets with the proportion of 8 to 2. We used Adam as the optimizer. For LSTM and GRU, we set the learning rate as 1e-4 and that of BERT Fine-Tuning was set at 1e-5. We put all the models' training processes with the EarlyStopping feature with three epochs for patience to save time. The results are shown in Table 1. BERT Fine-Tuning had the best result; thus, it would be the selected model. The architecture is shown in Figure 4.

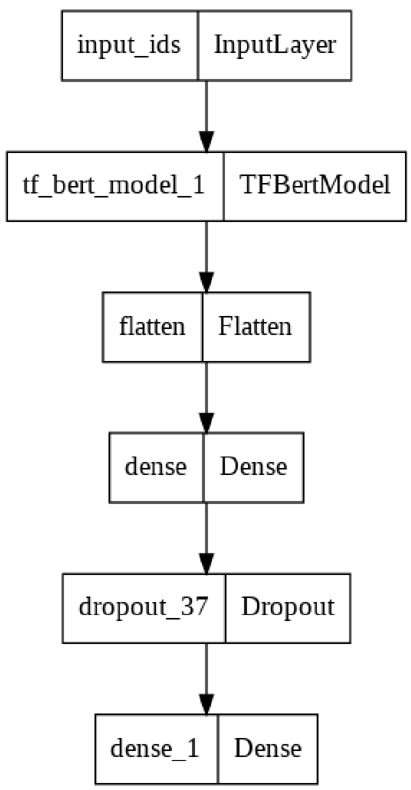

Figure 4: Architecture of BERT Fine-Tuning Model

Flask Application

Using Python and Flask, we created a web application that will give a sentiment score to any financial news headline based on the BERT Fine-Tuning model we trained on the data. The homepage is shown in Figure 5.1. It is used by simply pasting a headline into the text bar and hitting 'Submit' where the user will be redirected to a page displaying the headline as well as a sentiment score as shown in Figure 5.2.

The front-end uses Bootstrap templates for HTML and CSS and the back-end loads the trained Tensorflow model which generates a prediction on the given headline.

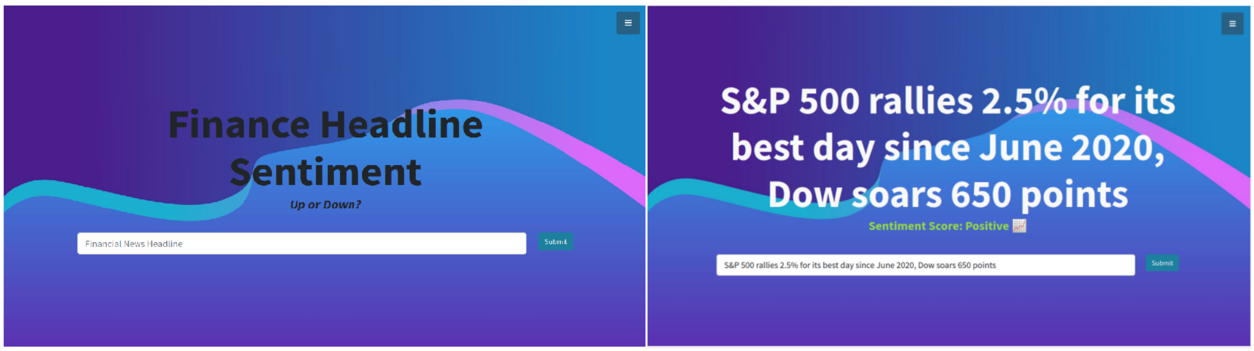

Figure 5: Flask App

Conclusion

Results

As our exploratory data analysis suggested, our headlines were composed of mostly positive sentiments with some negative sentiments as well. We utilized multiple RNN models and BERT features. Our best model was BERT Fine-Tuning with eight epochs, 0.43 loss, and 84.6% accuracy.

Our model is crucial since the short and middle term of the stock markets are impacted by the news. Quickly capturing the sentiment of the news titles could help investors, especially day traders, gain profits and avoid losses. Not only in finance,but our model could also be utilized in the review industry. Instead of reading each review or headline, our model returns the sentimental results, and this helps companies to improve their products and services.

Models	Epoch	Loss	Test Accuracy
LSTM	18	0.845	63.9%
GRU	23	0.838	63.4%
BERT Sentimental Analysis	-	-	23.1%
BERT Fine-Tuning	8	0.43	84.6%

Table 1: Results Of Each Model

Shortcomings and Future Directions

As stated in the introduction one of our biggest shortcomings was the lack of detail in the data. Our dataset only contained two columns, one being the news headline, and the other being the sentiment label. This project could be further extended by scraping data from the web to include the corresponding date, market trend, and stock prices in relation to each observation. While we did not have that data at our disposal, we did a few literature reviews to find scientific articles written about the correlations between the sentiment to news headlines and stock price. Our project is useful in our functionality of predicting the sentiment of a potential headline and using it as a tool for journalists to give confidence in the reaction to the headline, whether it be positive, negative, or neutral. One further extension is to create more labels that express sentiments in more detail, rather than three vague categories. Is there a difference between a “sad” headline versus an “angry” headline? Does the use of inflammatory language have a more polarizing effect on one aspect of the market than another? These are all questions that could be further explored within the context of sentiment analysis.