Predicting Stock Price Using Reddit World News Sentiment Data (Team 10)

Executive Summary

This project aims to forecast stock price trend in a long term fashion through stock features extracted from Quandl and sentiments extracted from Reddit world news. The model was trained on historical stock price data of 60 companies from 6 different industries combined with the top 25 news headlines on Reddit world news (on a daily basis for the last 8 years). These features were used to predict stock price trends through C5.0 Decision Tree with accuracy slightly greater than 70%.

Project Introduction

The project aims to predict stock market trends by using sentiment measures gathered from Reddit news headlines. The sentiment from the daily top 25 news headlines of r/worldnews will be analyzed and used to predict stock trends. We also break down stock price and create derived data describing the day of the week, the industry of the companies, and lagged price and sentiment data.

Previous Work Done by Others

While working on our project, we came across a research by Indiana University which was carried out in 2011.

In the research, they utilized Twitter data to conduct affect analysis and polarity classification through open source tool, OpinionFinder, and Google-Profile of Mood States (GPOMS) and reach an accuracy of 86% (Bollen et al., 2011).

The major difference between this project and our project is that they divided the Twitter feeds into 6 mood dimensions (Calm, Alert, Sure, Vital, Kind, and Happy) while in this project we divided Reddit news headlines into sentiments (including Very Positive, Positive, Neutral, Negative, Very Negative, and Sarcasm).

Data Description

Our feature set contains the following, from a final dataset of 120K rows. The 41 Columns describe 60 companies from 6 industries, and the main features are as follows:

1. Sentiments:
   1 day before, 2 days before, 3 days before sentiment score with very positive, positive, neutral, negative, very negative, and sarcasm on 25 different headline for each day. We also created derived attributes to summarize the sentiment and lagged them.
2. Stock Prices:
   1 day before, 2 days before, 3 days before (all values are “U” for “up” and “D” for “Down”. A very small number of values have “N” or neutral).
3. Industry and Company Code (only industry was used to predict)
4. Day of week and month.

Current State of Affairs

Our project was designed with the three following motivating ideas:

1. Stock Price Trend Prediction can be essential for investors when they are timing important decisions. We want to test if we can accomplish this using sentiment analysis.
2. The tone of general news can affect stock prices. We utilize the sentiment of news article headlines to capture this effect. We want to use the available sentiment analysis tools to create counts of each type of sentiment, for every day.
3. Reddit is very large, crowd-sourced, and also acts as a news source for the US public. We want to use this as a source for news, rather than traditional news sites, because the top stories are chosen by users.

Description of the Algorithm

The Decision Tree (C5.0 variant) algorithm that our team selected has several benefits. Out of all of the algorithms that we tried (FFBP-ANN, CHAID, QUEST, SVM, Logistic Regression and CART), it had the best accuracy. In addition, the algorithm accepts both categorical and continuous input variables.

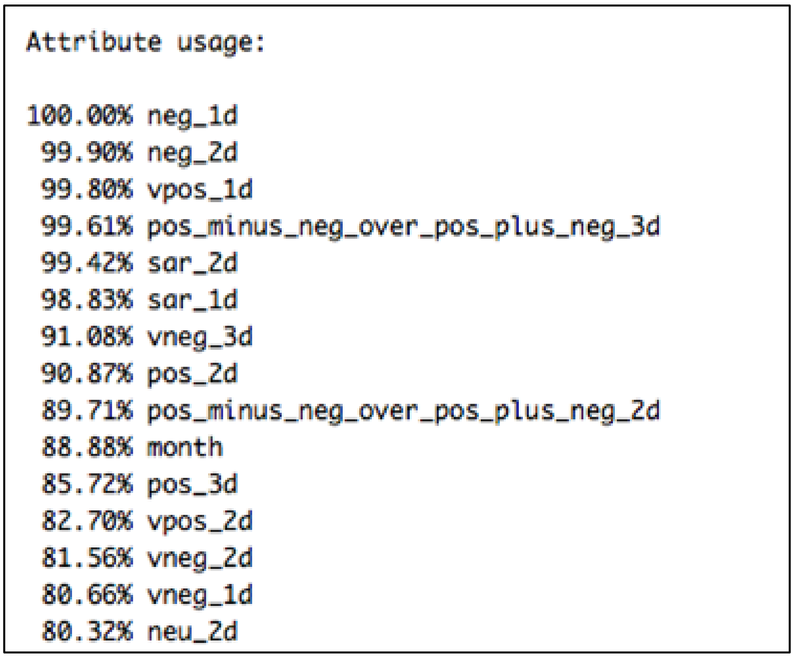
To train the Decision Tree algorithm, we randomly split the entire database up into a testing dataset and a training dataset. Approximately 80 percent of the data points were assigned to the training dataset. The remaining 20 percent of the data points were assigned to the testing dataset. The Decision Tree algorithm was run against the training dataset to train it, which consisted of approximately 5,000 data points. To test the accuracy of the newly trained Decision Tree algorithm, the "predict" function in R statistical package was used to make predictions for all of the data points within the testing dataset. The difference between the predicted results provided by the Decision Tree algorithm and the observed results were used to calculate accuracy.

The final accuracy of the Decision Tree algorithm we trained is 70.5 percent. The confusion matrix of the prediction is provided in Table 1.

	Classified Down	Classified No Change	Classified Up
Class Down	33,439	0	12,734
Class No Change	433	17	440
Classified Up	10,515	5	37,681

Table 1. Confusion Matrix (Decision Tree Summary Output): This confusion matrix is used to show the number of correctly classified data points for each of the 3 classification classes (Down, No Change, Up). Numbers listed on the diagonal are observations that have been correctly classified, while off diagonal observations were ones that have been incorrectly classified.

Figure 1. provides a summary of the attribute usage (more than 80%) of our model. The most important predictors tended were lagged sentiment values, rather than industry or day of the week. Also, it is obvious that sentiment-related attributes dominate the model.



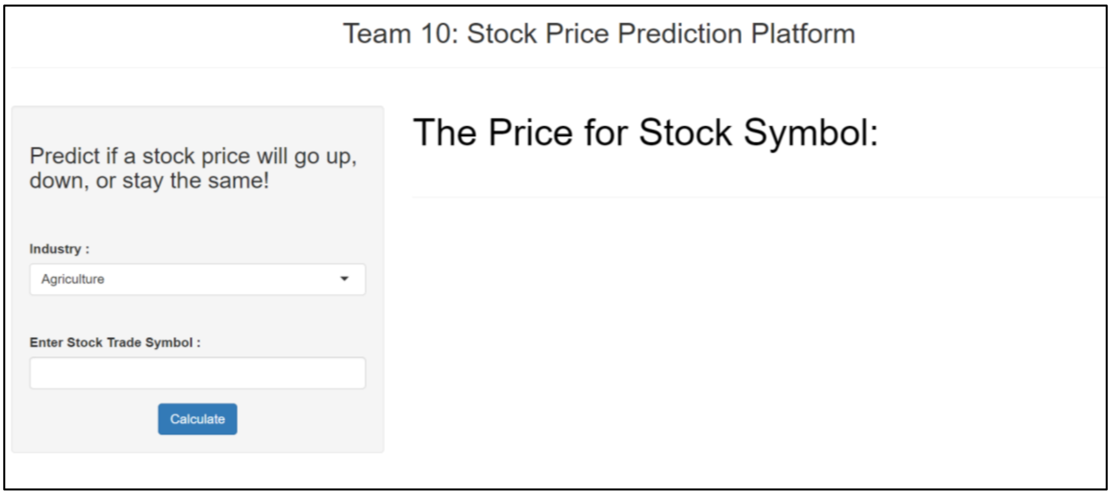
```
Attribute usage:

100.00% neg_1d
 99.90% neg_2d
 99.80% vpos_1d
 99.61% pos_minus_neg_over_pos_plus_neg_3d
 99.42% sar_2d
 98.83% sar_1d
 91.08% vneg_3d
 90.87% pos_2d
 89.71% pos_minus_neg_over_pos_plus_neg_2d
 88.88% month
 85.72% pos_3d
 82.70% vpos_2d
 81.56% vneg_2d
 80.66% vneg_1d
 80.32% neu_2d
```

Figure 1. Attribute Usage: Attributes with more than 80% utilization.

Implementation and Results

The C5.0 implementation was done in R's "C50" package, and then trained with the data previously described. This model was then saved and used as part of the final deployable project on ShinyApps.io. Figure 2. shows the landing page of the running application.



Team 10: Stock Price Prediction Platform

Predict if a stock price will go up, down, or stay the same!

Industry : Agriculture

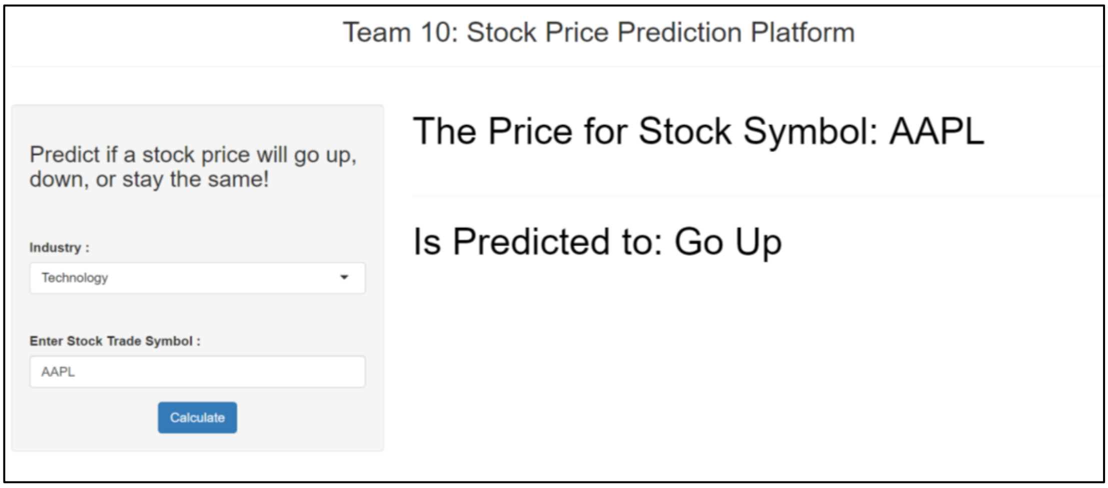
Enter Stock Trade Symbol :

The Price for Stock Symbol:

Figure 2. Landing Page of the Application

The user must specify a company symbol and industry that they want to check, so that on the backend, our system can pull in the appropriate stock data (to provide lagged values) and also sentiment values from reddit. The sentiment values are collected by a separate script that must be run daily at 11pm, to get that day's top 25 headlines.

Figure 3. shows an example of a prediction that had been run for a company with code "AAPL" and industry "Technology". Our system predicted that the stock would go "down" that day.



Team 10: Stock Price Prediction Platform

Predict if a stock price will go up, down, or stay the same!

Industry : Technology

Enter Stock Trade Symbol : AAPL

The Price for Stock Symbol: AAPL

Is Predicted to: Go Up

Figure 3. An Example of the Prediction on "Technology" and "AAPL"

References

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, *2*(1), 1-8.