Indicator aided text-based classification for stock movement

Adithya Murali
Department of Computer Science
University of Wisconsin Madison
Madison, WI 53706
murali5@wisc.edu

Nikhil Kannan
Department of Computer Science
University of Wisconsin Madison
Madison, WI 53706
nkannan2@wisc.edu

Abbinaya Kalyanaraman
Department of Computer Science
University of Wisconsin Madison
Madison, WI 53706
akalyanaram2@wisc.edu

Akshaya Kalyanaraman
Department of Computer Science
University of Wisconsin Madison
Madison, WI 53706
kalyanarama3@wisc.edu

Abstract

Machine learning has been applied in many areas of finance such as credit risk modelling and fraud detection [1][2]. With increasing computation power, it has been applied quite extensively to predicting stock trends based on various financial factors [8][9]. Furthermore, news data has become a popular factor which drives investor sentiment [12]. Here, we demonstrate a Random Forest classification model which makes use of news data and financial indicators to forecast trends over various time periods. We show that news data, when augmented with financial indicators, produces the ability to forecast long-term trends.

1 Introduction

Application of machine learning models to finance has increased with increased computational capabilities. Machine learning has been incorporated in various use cases to forecast movement of financial securities. For example, machine learning models have been incorporated in analysis of credit risk models and securities fraud prevention [1] [2]. As a result, the power of machine learning models has yielded its application in understanding movement of stock prices inevitable.

Previously, stock trends were analyzed using technical indicators such as ADX and RSI [3]. These indicators helped establish a formal understanding of trends in stock movements across a range of time periods, and have employed themselves as macroeconomic indicators and used in intra-day FX trading [4] [5]. Consequently, these indicators found themselves as data points in learning algorithms [5]. Newer models with hybrid methods have come to place, indicating a "better than chance" accuracy of predicting market movement [6]. In essence, these models reinforce the idea that the market is predictable [7].

As of late, text-based prediction of stock movement has gained popularity in a bid to gauge sentiment. Previous research has leveraged data from Twitter to understand the movement of stock using moods, reporting accuracy as high as 86.7% indicating that there is possible correlation between sentiment of investors and stock price movement [8] [9]. Other techniques have used financial news data to forecast returns [10]. However, even though news articles are structured, they suffer from domain-dependence which can potentially result in incorrect sentiment classification [11].

In our work, we demonstrate a hybrid approach where we use text data along with a combination of widely used financial indicators to obtain classification for movement of 10 stocks in the Oil and Gas industry listed on the New York Stock Exchange. Instead of using sentiment analysis tools to classify text, we use a more nuanced approach to assigning sentiment based on stock movement across a given time-period. We preprocess the text data and use Random Forests for building our predictive model. As input, our model takes a matrix of word weights and indicator values and assigns the appropriate label (either as positive or negative). In section 2, we describe the methods used for generating data and models used. Section 3 and 4 delve into the results of our analysis and discussions on the use of this technique as part of a viable trading strategy.

2 Methods

2.1 Obtaining Data

We aimed at constructing a dataset that accurately reflects investor sentiment. In order to achieve this, we scraped news articles from two popular websites: Investopedia.com and seekingalpha.com. Investopedia caters general news about stock price movements, whereas seekingalpha.com publishes articles that more written by relatively seasoned traders and investors. This mixture in data provided for a more diverse dataset that gives us better representation of investor sentiment.

Table 1: Stocks

Company Ticker
COP
EMR
ENB
EOG
GE
HAL
OXY
PSX
VLO
XOM

For each of the 10 stocks noted above, we scraped articles from both websites ranging from Nov 2012 to Nov 2018. Next, the date of publication was extracted and stored in YYYY-MM-DD format. The text was then removed of all special characters, numbers and stop words. However, we did not perform stemming or lemmatization to preserve the contextual use of certain words in financial vocabulary. For instance, "higher" can be stemmed to "high", but in the context of financial news articles, these words may carry different meanings. The phrase "NASDAQ jumps to new highs!" may harbor positive sentiment among investors but the phrase "VIX climbs higher by 50 points" may be perceived as the opposite by some investors. Here, VIX indicates the Volatility Index, which provides a snapshot of the market volatility at any given time.

To complement the news data, we computed 13 trend (Table 2.) forecasting technical indicators that allow investors to forecast trends in the market. For the 10 stocks in Table 1, we obtained daily opening prices, closing prices, volume, day high and day low from Yahoo Finance (https://finance.yahoo.com/). These values ranged from 26th November 2006 to 26th November 2018. Next, we calculated the indicators given in Table 2 for each date for each stock.
We chose to construct an independent dataset for each stock since the information contained for a given stock may not be generalizable to other stocks given the need to preserve contextual information.
Existing sentiment analysis tools don't consider the contextual references of certain words which may be domain specific [11]. In our study, we seek to incorporate financial sentiment data as a result of stock price movement across some given time frame or time period t. Given some news n, we

Table 2: Financial Indicators

Indicator	Calculation	Notes
Volatility (Standard Deviation)	$\sqrt{\frac{1}{N}\sum_1^n (x_i - \mu)^2}$	$\mu = 20$ Day SMA $n = 20$
Momentum	$V_i - V_{i-n}$	$i =$ Closing price on Day i $n =$ Lookback period
Average True Range (ATR)	$ATR_i = \frac{ATR_{i-1} * 13 + TR_i}{14}$	$TR_i = max \begin{cases} \text{High}_i - \text{Low}_i \\ \text{High}_i - \text{Close}_{i-1} \\ \text{Low}_i - \text{Close}_{i-1} \end{cases}$
Relative Strength Index (RSI)	$RSI_i = 100 - \frac{100}{1 + RS_i}$	$RS_{14} = \frac{(\sum_{k=1}^{14} Gain(k))/14}{(\sum_{k=1}^{14} Loss(k))/14}$ $RS_i = \frac{\frac{\left(\sum_{k=i-14}^{i-1} Gain(k)\right)}{14} * 13 + \frac{Gain(i)}{14}}{\frac{\left(\sum_{k=i-14}^{i-1} Gain(k)\right)}{14} * 13 + \frac{Gain(i)}{14}}$
Moving Average Convergence Divergence Oscillator (MACD)	$MACD_i = EMA_{12}(Close_i) - EMA_{24}(Close_i)$ $Signal = EMA_9(Close_i)$	
True Strength Index (TSI)	$TSI_i = 100 * \frac{Double\ Smoothed\ PC}{Double\ Smoothed\ \lvert PC \rvert}$	$PC_i = \text{Close}_i - \text{Close}_{i-1}$ $Double\ Smoothed\ PC = EMA_{25}(EMA_{12}(PC))$ $Double\ Smoothed\ \lvert PC \rvert = EMA_{25}(EMA_{12}(\lvert PC \rvert))$
Fast Stochastic Oscillator	$\%K = \frac{Close_i - min(Low_{i-n:i})}{max(High_{i-n:i}) - min(Low_{i-n:i})} * 100$ $\%D = SMA_3(\%K_i)$	$n =$ Lookback period
Ultimate Oscillator (UO)	$UO_i = 100 * \frac{\left[\left(4 * \frac{\sum_{k=i-7}^{i} BP_k}{\sum_{k=i-7}^{i} TR_k}\right) + \left(2 * \frac{\sum_{k=i-14}^{i} BP_k}{\sum_{k=i-14}^{i} TR_k}\right) + \left(\frac{\sum_{k=i-28}^{i} BP_k}{\sum_{k=i-28}^{i} TR_k}\right)\right]}{(4+2+1)}$	$BP_i = \text{Close}_i - \text{min(Low, Close}_{i-1})$ $TR_i = \text{max(High}_i, \text{Close}_{i-1}) - \text{min(Low}_i, \text{Close}_{i-1})$
Bollinger %B	$\%B = \frac{Close_i - LowerBand_i}{UpperBand_i - LowerBand_i}$	$UpperBand_i = SMA_{20}(\text{Close}_i) + 2 * \sigma(i)$ $LowerBand_i = SMA_{20}(\text{Close}_i) + 2 * \sigma(i)$
Aroon Indicator	$AroonUp_i = \frac{\left(25 - argmax_i\left(high_{i-25:i}\right)\right)}{25} * 100$ $AroonDown_i = \frac{\left(25 - argmin_i\left(low_{i-25:i}\right)\right)}{25} * 100$	

assign either a label "positive" or "negative" to the article n depending on the movement of the stock price within a future time frame. For instance, if t = 10 Days, we will compute the difference between stock closing prices on day i and day $i+10$. For a news article n released on day i, we will assign a label "positive" to this news article if the difference in stock prices on day i and day $i+10$ is positive. Similarly, we will assign a label "negative" if the difference in stock prices is negative. This binds the sentiment of each word to the actual market movement, thus tying together the contextual sentiment of domain-specific words to market reactions. A sample is shown in Table 3.

Table 3: Financial sentiment data across time periods

Close	5 Days	10 Days	2 Weeks	1 Month	2 Months
71.967178	Negative	Negative	Negative	Negative	Negative
71.867592	Positive	Negative	Negative	Negative	Negative
71.509102	Negative	Negative	Negative	Positive	Negative
71.260147	Positive	Negative	Positive	Positive	Negative
70.991287	Positive	Negative	Positive	Positive	Negative

2.2 Model Design

News has shown to influence future stock trends [12]. Furthermore, investors typically pay attention to stocks that make the news to determine future purchasing strategies [13]. Thus, a predictive model can be applied to forecast stock price trends.

We convert our text data for each stock into a sparse matrix of word counts. We then apply the TF-IDF algorithm to obtain weights for each word in the text corpus to significance their relevance to the stock [14]. For each corresponding article, we then append indicator data corresponding to the release date of that article.

We employ an ensemble learning technique for classification purposes as they have proven to perform better than standalone learning techniques such as Support Vector Machines in domains such as finance and genomics [15] [16] [17]. Here, we use the Random Forests classification technique to classify a given vector of words and corresponding indicator values as either "positive" or "negative" [18]. We construct our model using the Random Forests module available in the scikit-learn package for Python. We set the number of estimators to 100 and Gini impurity as criterion for split. We obtain the accuracy for our models using 10-fold cross-validation. The number of features in the dataset often becomes excessive due to the number of words being considered for classification. The use of Random Forests allows us to reduce the number of rule-based splits, allowing us to attain superior accuracies as demonstrated in Section 3.

3 Results

3.1 Comparison against other models

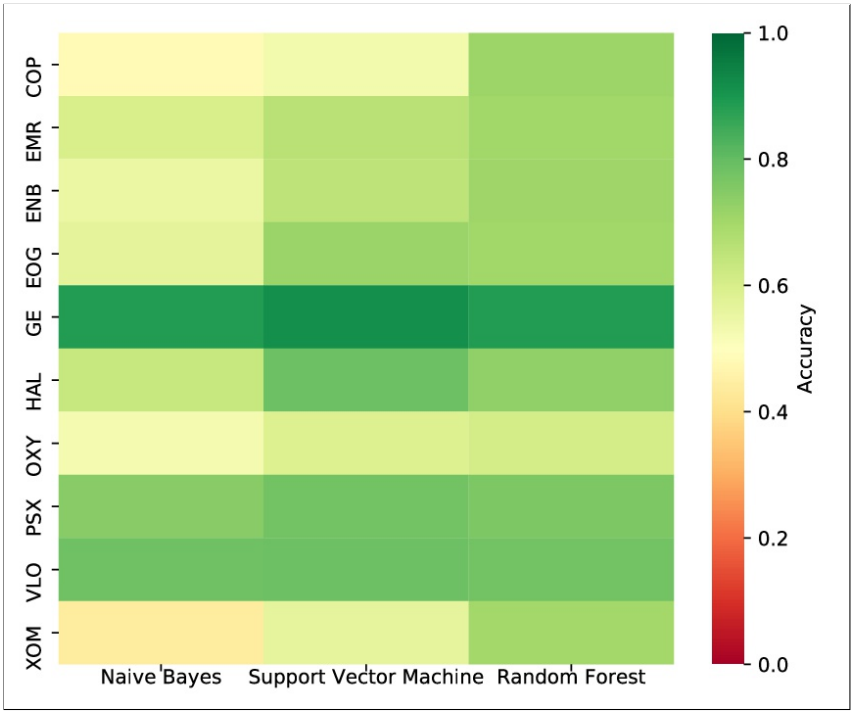

Figure 1: Heatmap indicating classification accuracy of Naïve Bayes, Support Vector Machines and Random Forest classifiers on 10 stocks. Random Forests show better accuracy in comparison to Naïve Bayes and Support Vector Machines for trends forecasted 2 months after the release of an article.

Support Vector Machines and Naive Bayes classifiers are two other classification algorithms that have been widely used in text-classification [19]. However, our method considers each word in the corpus to be a feature. Since the number of times a word occurs in an article is highly variable, this results in a highly sparse data set, hindering the ability of Support Vector Machines or Naïve Bayes classifiers to perform optimally. Figure 1. indicates this phenomenon, where we can notice for most stocks, Naïve Bayes classifiers perform at an average of 61%, whereas Support Vector Machines perform slightly better, with 68% average accuracy. Random Forests seem to perform best, with an average accuracy of 71% due to rule-based decisions. The presence of both highly sparse text data coupled with dense indicator data assists in creating clearly defined splits during the training process.

3.2 Performance across time periods

In order to understand the ability of the classifier to gauge trends across time periods, we computed classification accuracy using difference in closing prices over 5 days, 10 days, 2 weeks, 1 month and 2 Months.

In Figure 2., within the early days of release of an article we can see that using news data in combination with technical indicators performs comparably to using news data alone. This is expected as trends indicated by news articles or technical indicators may not take shape immediately. However, as the time period from the release of the article increases, we notice that a combination of news data and technical indicators performs better than using text alone. At the 2 Month time period, we notice that use of news and indicator data performs categorically better to using text.

We notice this behavior because news data is not immediately digested by the readership

and this does not result in an immediate effect on the stock price. As a news article spreads through platforms such as twitter (often as URL's) the information is disseminated to a wider audience, often quickly [20]. Once this information has been spread, the effect on the market begins to take shape. Although, the foreshadowing of a new trend as a result of news is not fully evident through the use of news alone. Here, technical indicators provide augment the news-based trend forecast, providing higher accuracy as the time period increases.

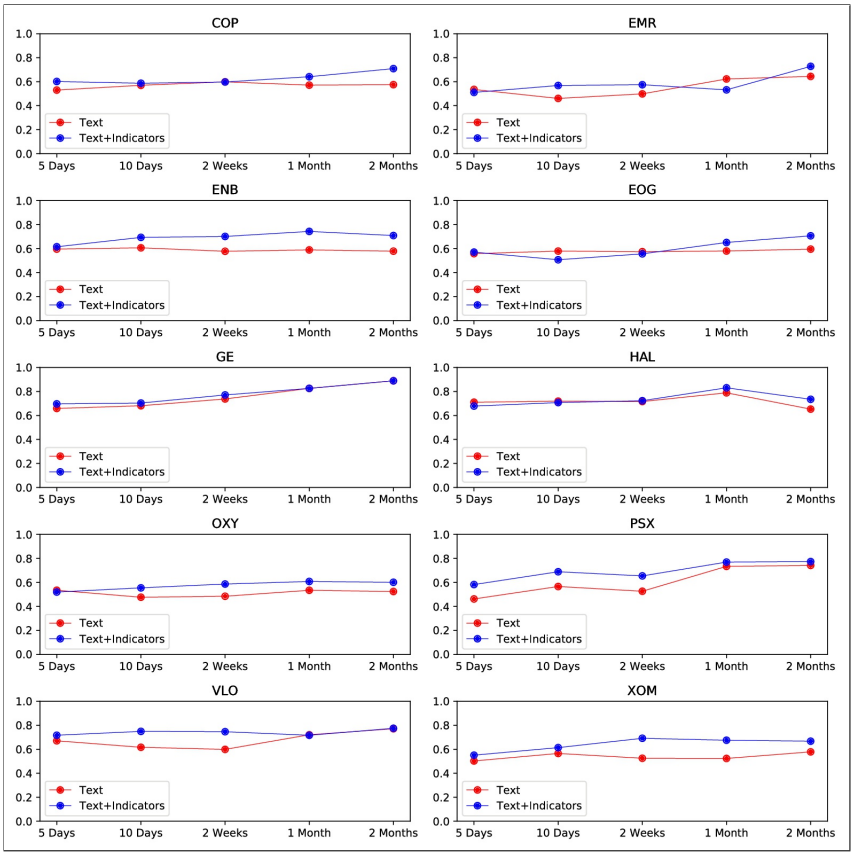

Figure 2: Accuracy across time periods for each of the 10 stocks. Y-axis represents accuracy. X-axis represents time-periods.

3.3 Increasing Context: performance across n-grams

While the use of individual words as features for classification has allowed for increased classification accuracy, it does not represent context. In order to provide context to news articles, we ensure that all features in our dataset is represented by combinations of uni-grams, bi-grams and tri-grams of all possible words in the news corpus.

In general, we notice that the use of tri-grams alone provides us with superior performance in comparison to other combinations of n-grams. In most cases, this is especially pronounced as the time-period increases; for GE, the 2 Month period hits as high as 95%.

This can be attributed to the fact that as the number of n-grams taken into consideration increases, the context of words taken into consideration increases. In Figure 3. we notice the worst performance in the case of n-gram combinations. In general, we notice that the combination of unigrams, bigrams and trigrams has the worst performance, followed by the unigram-bigram and bigram-trigram combinations. Since our model considers each n-gram as a feature, these combinations would construct an extremely sparse dataset, causing the classifier to rely entirely on the financial technical indicators. Hence, causing comparably poorer performance.

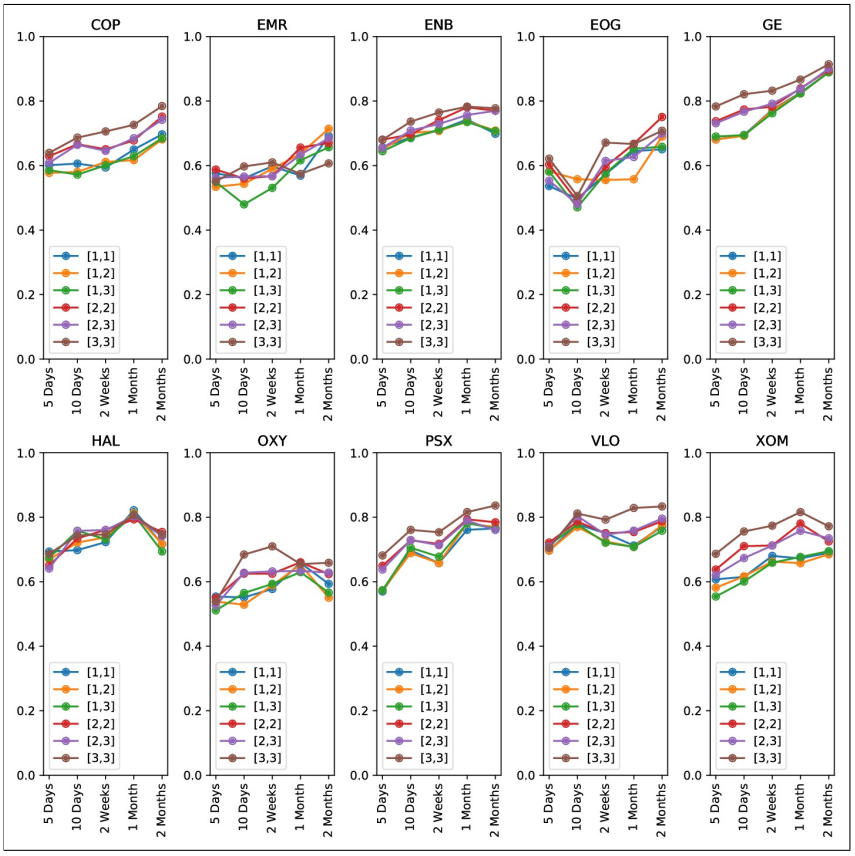

Figure 3: [1,1] uni-grams only. [1,2] bi-grams and uni-grams, [1,3] uni-grams, bi-grams and tri-grams. [2,2] bi-grams only, [2,3] big-grams and tri-grams. [3,3] tri-grams only. The Y-axis in each plot represents accuracy. X-axis represents time-period

The performance increases when using unigrams, bigrams or trigrams in isolation. Of these, unigrams perform worst in comparison. This can be attributed to the fact that words as features are not considered in context. For bigrams and trigrams, we notice an increase in the accuracy as context is considered. Since trigrams incorporate the highest amount context, their performance is seen to provide optimal accuracy.

4 Conclusions

Previous works in understanding sentiment resort to using sentiment analysis tools to acquire the mood of a news article used to forecast market trends. Here, we propose an alternate method to gauge sentiment of a news article using market movement from date of article release. We label an article as being "positive" if price of the corresponding stock mentioned in the news article has gone up and "negative" otherwise. We then combine this news data with 13 technical indicators used to forecast trends in stock movement to generate a new dataset with better forecasting properties. We then use a Random Forest machine learning model to predict "positive" and "negative" for a new news article.

We then showed that Random Forests provide better accuracy in comparison to other commonly used text-classification methods such as Support Vector Machines and Naïve Bayes classifiers since the high sparsity of the feature set makes it challenging for algorithms such as Support Vector Machines to provide meaningful results.

We then looked at how the classifier performs when using only news versus using a combination of both news and technical indicators. In general, we have shown that the combined use of technical indicators and news articles perform better than using text in isolation. Furthermore, we have shown that the accuracy increases as we extend the time period further from date of publication of the article. This shows that news articles may not be immediately effective in the movement of stock price, but their effects can be seen gradually in stock movement over an extended period of time.

Finally, we looked at how word contexts affect classification accuracy. For all 10 stocks, we saw that trigrams were most effective at classification in all cases because they retained highest context among words. In contrast, we saw that combinations of n-grams performed worst among all 10 stocks due to the high sparsity and often rely heavily on technical indicators provide higher classification accuracy.

The purely academic nature of this study may not necessarily make it amenable to use in real trading strategies. Large number of secondary and tertiary factors are at play when understanding relationships between trends and news. Our classifier may not work optimally in case of unsuspected, non-cyclic movements in markets. Furthermore, our selection of oil stocks is used to demonstrate the capability of this technique in cyclical markets where trends are based on movements in external factors.

Future work can attempt to incorporate "secondary" industries, or industries that maintain a symbiotic relationship with a primary industry. An example of this is the relationship between the Oil and Gas industry and the Cosmetics industry. Many cosmetics are dependent on petroleum, thus increased prices of petroleum by Oil and Gas companies can indirectly affect the movement of stock prices of companies in the Cosmetics industry. Thus, we can potentially predict movement of stocks in Cosmetics industry by using news articles based on Oil and Gas.

Acknowledgments

We would like to thank our Professor, Dr. Yingyu Liang, for providing us with an opportunity to research on this topic, and better understand the Machine Learning techniques in stock analysis.

Supplementary Material

The code for the implementation of stock analysis, which calculates the accuracy of the classifiers can be found in the following github repository:

`https://github.com/TurbulentCupcake/CS760-FinalProject.git`

The below figure shows the ROC curve indicating the accuracy for each of the 10 stocks.

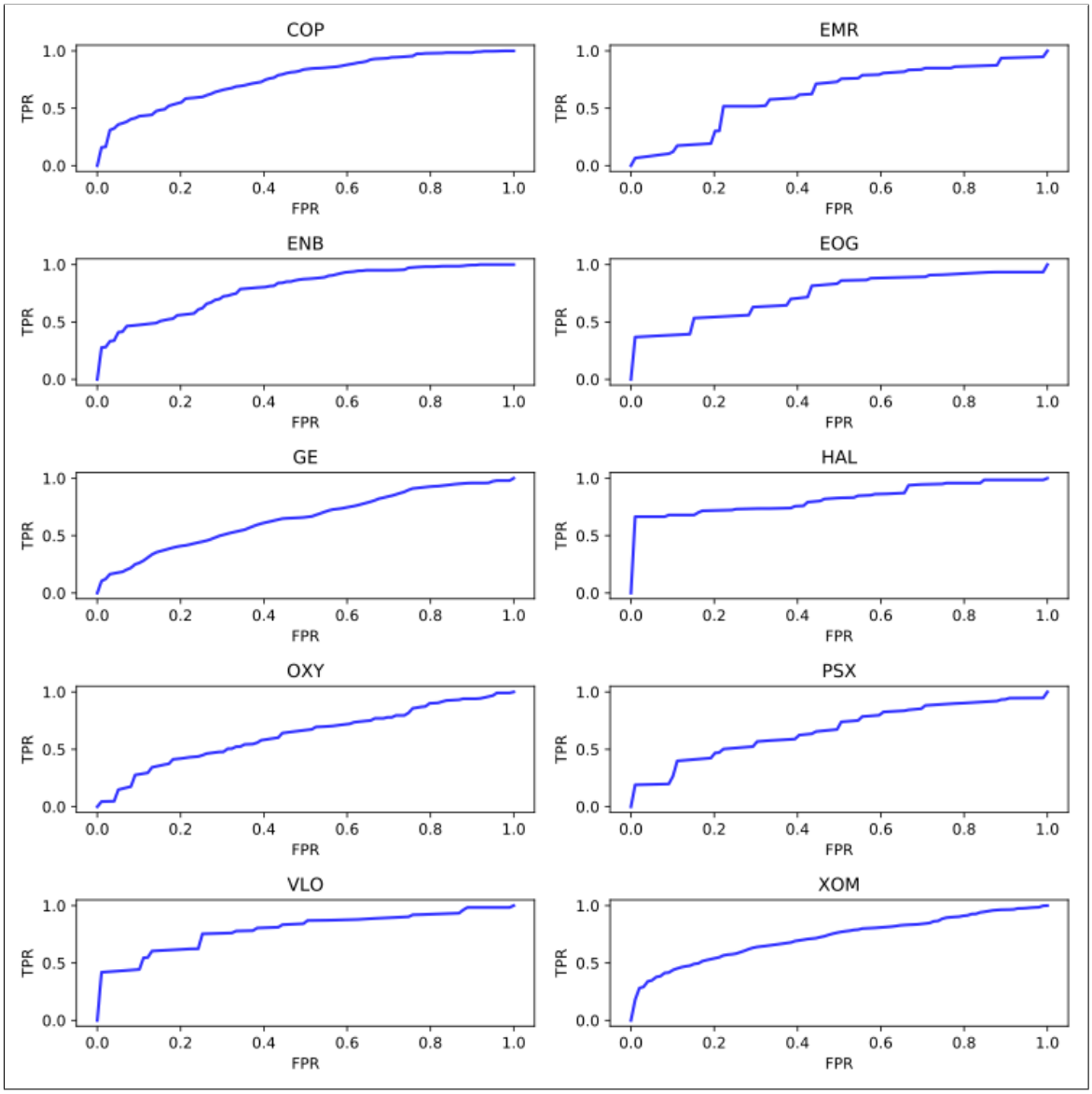

Figure 4: ROC curve indicating accuracy for each of the 10 stocks in consideration.

References

[1] J. Galindo and P. Tamayo, "Credit Risk Assessment Using Statistical and Machine Learning: Basic Methodology and Risk Modeling Applications," Springer Computational Economics, vol. 15, no. 1-2, pp. 107-143, 2000.

[2] E. Ngai, Y. Hu, Y. Wong, Y. Chen and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature,," Decision Support Systems, vol. 50, no. 3, pp. 559-569, 2011.

[3] J. W. Wilder, New Concepts in Technical Trading Systems, Trend Research, 1978.

[4] C. J. Neely, D. E. Rapach, J. Tu and G. Zhou, "Forecasting the Equity Risk Premium: The Role of Technical Indicators," Management Science, vol. 60, no. 7, 2014.

[5] M. A. H. Dempster, T. W. Payne, Y. Romahi and G. W. P. Thompson, "Computational learning techniques for intraday FX trading using popular technical indicators," EEE Transactions on Neural Networks, vol. 12, no. 4, pp. 744-754, 2001.

[6] R. Choudhry and K. Garg, "A Hybrid Machine Learning System for Stock Market Forecasting," International Journal of Computer and Information Engineering, vol. 2, no. 3, 2008.

[7] M. H. L. B. Abdullah and V. Ganapathy, "Neural network ensemble for financial trend prediction," Intelligent Systems and Technologies for the New Millennium, vol. 3, pp. 157-161, 2000.

[8] J. Bollen, H. Mao and X. Zeng, "Twitter mood predicts the stock market," Journal of Computational Science, vol. 2, no. 1, pp. 1-8, 2011.

[9] X. Zhang, H. Fuehres and Peter A. Gloor, "Predicting Stock Market Indicators Through Twitter “I hope it is not as bad as I fear”,," Procedia - Social and Behavioral Sciences, vol. 26, pp. 55-62, 2011.

[10] B. Wuthrich, V. Cho, S. Leung, D. Permunetilleke, K. Sankaran and J. Zhang, "Daily stock market forecast from textual web data,," 1998 IEEE International Conference on Systems, Man, and Cybernetics, vol. 3, pp. 2720-2725, 1998.

[11] D. M. E.-D. M. Hussein, "A survey on sentiment analysis challenges," Journal of King Saud University - Engineering Sciences, vol. 30, no. 4, pp. 330-338, 2018.

[12] A. Groß-Klußmann and N. Hautsch, "When machines read the news: Using automated text analytics to quantify high frequency news-implied market reactions," Journal of Empirical Finance, vol. 18, p. 321–340, 2010.

[13] B. M. Barber and T. Odean, " All That Glitters: The Effect of Attention and News on the Buying Behavior of Individual and Institutional Investors," The Review of Financial Studies, vol. 21, no. 2, pp. 785-818, 2008

[14] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," Journal of Documentation, vol. 28, no. 1, 1972.

[15] C. CORTES and V. VAPNIK, "Support-Vector Networks," Machine Leaming, vol. 20, pp. 273-297 , 1995.

[16] G. Wang, J. Hao, J. Ma and H. Jiang, "A comparative assessment of ensemble learning for credit scoring," Expert Systems with Applications, vol. 38, no. 1, pp. 223-230, 2011.

[17] A. Statnikov, L. Wang and C. F. Aliferis, "A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification," BMC Bioinformatics, vol. 9, no. 319, 2008.

[18] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan and B. P. Feuston, "Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling," Journal of Chemical Information and Computer Sciences , vol. 43, no. 6, pp. 1947-1958, 2003.

[19] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features".

[20] B. Suh, L. Hong, P. Pirolli and E. H. Chi, "Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network," Palo Alto Research Center, Inc.

[21] T. C. Doron Avramov, "Predicting stock returns," Journal of Financial Economics, pp. 387-415, 2006.