Predicting Stock Market Movements with Social Media and Machine Learning

Paraskevas Koukaras[a], Vasiliki Tsichli and Christos Tjortjis[b]
School of Science and Technology, International Hellenic University,
14th km Thessaloniki–N. Moudania, Thermi, 57001, Thessaloniki, Greece

Keywords: Social Media, Prediction, Machine Learning, Data Science, Stocks.

Abstract: Microblogging data analysis and sentiment extraction has become a popular approach for market prediction. However, this kind of data contain noise and it is difficult to distinguish truly valid information. In this work we collected 782.459 tweets starting from 2018/11/01 until 2019/31/07. For each day, we create a graph (271 graphs in total) describing users and their followers. We utilize each graph to obtain a PageRank score which is multiplied with sentiment data. Findings indicate that using an importance-based measure, such as PageRank, can improve the scoring ability of the applied prediction models. This approach is validated utilizing three datasets (PageRank, economic and sentiment). On average, the PageRank dataset achieved a lower mean squared error than the economic dataset and the sentiment dataset. Finally, we tested multiple machine learning models, showing that XGBoost is the best model, with the random forest being the second best and LSTM being the worst.

1 INTRODUCTION

Stock market forecasting is an important academic topic, which has attracted academic interest since the early 1960's (Fama, 1965). Although a lot of effort and time has been spent on predicting financial time series, the results of the research are not robust. In recent years a lot of researchers have shifted their focus from classical econometric approaches to machine learning approaches. With the rise of microblogging platforms, such as Twitter, StockTwits and others, information is more available than ever. Given that emotions can have a significant effect on economic decisions (Bollen et al., 2011), alongside with herding phenomena (Devenow and Welch, 1996), one can assume that mining information through microblogging platforms might be the key to achieve better results in predicting stock market movements.

Stock market forecasting has drawn a lot of academic attention since the 1960's. The first model that revolutionized how the stock was evaluated is the Capital Asset Pricing Model (or CAPM for short). CAPM was developed[1] by William Sharpe (Sharpe, 1964) who built on top of Markowitz's diversification theory. The model is fairly simple and is based on stock return sensitivity exhibited over the systemic risk (or market risk). It is quantitatively expressed with a beta (β) factor.

CAPM measures the return of a stock in accordance with the market risk. Every other risk that stems from the stock itself can be diversified as Markowitz proved in the portfolio theory. Thus, there is no point in measuring it. Although CAPM has been a fundamental decision making tool for asset managers, it has been criticized by academics due to its nature. It has been proven that the model is not robust and that it fails to give accurate results consistently. Fama and French (Fama and French, 1993) stated that the model is not robust and that a model that takes into account the size and the ratio of accounting over stock market value is more accurate. Their research prompted others to start looking for factors that may be affecting the returns of a stock. This gave birth to a whole new way of evaluating a stock, which is called technical analysis. Technical analysis is based on ratios and indicators that capture the momentum of the stock market. Although technical analysis is not based purely on academic research, it is extensively used and it is a common practice.

[a] https://orcid.org/0000-0002-1183-9878
[b] https://orcid.org/0000-0001-8263-9024

[1] There is a dispute on who deserves credit about CAPM, for more information check (Treynor, 1962)

Koukaras, P., Tsichli, V. and Tjortjis, C.
Predicting Stock Market Movements with Social Media and Machine Learning.
DOI: 10.5220/0010712600003058
In *Proceedings of the 17th International Conference on Web Information Systems and Technologies (WEBIST 2021)*, pages 436-443
ISBN: 978-989-758-536-4; ISSN: 2184-3252
Copyright © 2021 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

In recent years there has been a lot of effort to construct indicators or ratios based on the information of the microblogging community. Essentially, those indicators provide an overall sentiment over the market or a particular stock. Thus, the trader can have a more objective metric about the ”feelings”. Moreover, this data might contain useful information that otherwise would be unavailable. On the other hand, this approach contradicts with one of the most fundamental economic theories, the Efficient Market Hypothesis. As Fama (Fama, 1965) suggested, the price of a given stock embodies all the prior available information and it is impossible to forecast future values since the current ones reflect everything. Moreover, in efficient-market hypothesis (EMH), it is believed that the market adjusts the prices instantly as the news spread. Fama (Fama, 1965) also noted that the most probable future price is the current price. Nevertheless, recent empirical research provided evidence that sentiment plays an important role and can act as a determining factor of the stock market returns.

One of the biggest problems encountered by the researchers that used data from Twitter of other relevant sources is that they are noisy (See-To and Yang, 2017; Alshahrani Hasan and Fong, 2018), thus yielding spurious results. To deal with that problem, the authors either choose a specific news source, such as MarketWatch (Hájek, 2018) or Thomson Reuters (Mittermayer and Knolmayer, 2006) but this approach might lead to overlooking important information. Another issue is that they use a lot of data which might hinder their research in terms of efficiency and statistical robustness (Antweiler and Frank, 2004).

Our objective is to provide a more efficient way of handling those massive data, by looking for and distinguishing those data that matter the most. To achieve that, we use graphs that are constructed based on users and their data accordingly. We believe that our approach solves the problem of noisy microblogging data, without disregarding any useful information that might exist. Given our hypothesis we expect that the dataset which accounts for the noise in the data have a better score than the simple sentiment dataset.

2 METHODOLOGY

2.1 Data

In this section we present how we gathered Twitter data. Then, we provide an overview of the utilized economic variables and the reasoning behind these choices.

2.1.1 Twitter Data

We are interested in two categories of data, the tweets and the users that wrote those tweets. The main problem reported in the literature is the noisy nature of Twitter data (Rousidis et al., 2019; Koukaras et al., 2019; Koukaras and Tjortjis, 2019; Beleveslis et al., 2019; Oikonomou and Tjortjis, 2018). To overcome this problem, we used the ”cashtag” or ”$” in the tweets, which as (Chakraborty et al., 2017) notes, is more suited for gathering stock related data.

In total 782.459 tweets were downloaded starting from December, 1st of 2018 until July, 31st of 2019. Form these, we take all the tweets authors’ usernames and gather metrics for them. These metrics are used when we are checking the validity of our data. We also gather all users’ followers, a metric that is going to be used in the graph module. The module for gathering Twitter data is built upon a library called Twint. This library can provide tweets, users’ statistics (followers, following, likes, etc.) and also, it can gather users’ followers. Moreover, it also has a built-in function for storing those data directly to a database.

2.1.2 Economic Variables

Economic variables can act as predictors. These variables may vary from a fundamental analysis of a company’s balance sheet to technical indicators specially designed to capture specific events. In this work, we chose to use technical indicators for multiple reasons. First, technical analysis is based on examining a stock’s trend and constitutes a more robust tool for prediction. Moreover, one of the core principles of technical analysis is that a stock’s price reflects all the available information. Thus, it is focused more on past behavior of the market. Although technical analysis has been dismissed by academics (Malkiel and Fama, 1970), many of the leading trading companies use technical indicators to identify signals and trends on time. On the same line, we concluded that technical indicators are more suited for our research. Since they do not focus on news events, our final dataset will be more balanced with features that capture different aspects of trading. From all the available technical indicators, we opted for five of the most common ones:

1. The Aroon Oscillator is a trend indicator that measures the power of an ongoing trend and the probability to proceed by using elements of the Aroon Indicator (Aroon Up and Aroon Down). Readings above zero show an upward trend, while readings below zero show a downward trend. To signal prospective trend changes, traders watch for zero line crossovers (Mitchell, 2019).

2. The CCI was created to determine the rates of over-bought and over-sold stocks. This is done by evaluating the price-to-moving average (MA) relationship or by evaluating ordinary deviations from that median (Kuepper, 2019).
3. On-balance volume (OBV) is a momentum indicator that measures positive and negative volume flows (Staff, 2019).
4. The RSI is a momentum index measuring the magnitude of the latest price modifications that is used to assess which stocks are over-bought or over-sold. The RSI is an oscillator. Traditionally, traders interpret a score of 70 or higher as a sign that a stock is overbought or overestimated, leading to a trend reversal. An RSI of 30 or lower signals that a stock is undervalued (Blystone, 2019).
5. The Stochastic Oscillator attempts to predict price turning points by comparing the last closing price of a security to its price range. It takes values from 0 up to 100. A value of 70 or higher signals an overbought security.

These indicators were chosen for two main reasons. i) They are very robust and are extensively used in the industry and ii) they belong to the special category of ”Oscillators”. These are indicators that fluctuate within a range, commonly used to capture short term trends. Our sample period ranges from December, 1st of 2018 to July, 31st of 2019. This period is characterized by high fluctuations and small but powerful shocks (Trade War, No Deal Brexit, etc.). Thus, we believe that by using such variables will provide more accurate results instead of using fundamental analysis. Finally, to collect the economic variables, we used the API of Alpha Vantage.

2.2 Research Design

This section summarizes the main processes for conducting this research. At first, we designed a users Graph to obtain their importance incorporating the PageRank algorithm (Page et al., 1999). Afterwards, we analyzed the obtained tweets using two different lexicons. Lastly, we estimated five different machine learning models.

2.2.1 Identifying Influential Users

To identify influential users we generated a graph, we computed the PageRank score for each edge as well as the hub and authority scores. The Graph class is fairly simple and is based on the NetworkX library (Hagberg et al., 2008). Moreover, the PageRank and HITS algorithms are implemented in the NetworkX library (Hagberg et al., 2008).

PageRank and HITS are two algorithms that are often used to measure the importance of nodes on directed graphs. Both of the algorithms were designed to rank websites. The PageRank algorithm is a recursive algorithm. An internet page is important if and only if other important pages are linked with it. As it is usually described, a website's score is the probability of any random person browsing the web ending up on this website. This is by definition a Markov Process. Markov Processes model recursive phenomena, such as the weather. The PageRank algorithm starts with a set of websites (denoting the number of those websites with N). On each website, we assign a score of $1/N$. Afterwards, we sequentially update the score of each website by adding up the weight of every other website that links to it divided by the number of links emanating from the referring website. But if the website does not reference any other website, we distribute its score to the remaining websites. This process is executed until the scores are stable.

The Hypertext-Induced Topic Search (HITS) algorithm provides two scores, the ”Authority” and the ”Hub”. We tried to compute the HITS algorithm, but the algorithm never achieved convergence. Since we wanted to compute the hubs and the authorities for each day in our sample, the recursiveness of the algorithm poses a significant barrier. On the computing part, for each date, we needed to create a graph that references the follower relationships of the users. We are also interested in tweets posted between December, 1st of 2018 and July, 31st of 2019, thus creating 242 graphs.

2.2.2 Sentiment Analysis

Lexicon analysis outperforms other methodologies (Sohangir et al., 2018). In our approach, we used VADER (Valence Aware Dictionary and sEntiment Reasoner) (Hutto and Gilbert, 2014) and TextBlob. Both of these tools are part of the nltk library. VADER analyzer returns four scores, the negative, the positive, the neutral and the compound score. TextBlob returns two scores, the polarity (which should be very close to the compound score) and the subjectivity. We decided to use all of these variables as features in our models allowing us to compare those two analyzers. Furthermore, to achieve better accuracy on the scores, the tweets must be stripped from any special characters. More specifically, tweets often contain Unicode characters such as the non-breaking space. These characters should be normalized so as not to negatively affect the scoring of the analyzers.

2.2.3 Machine Learning Models

Decision Tree. The Decision Tree (DT) builds regression or classification models in the form of a tree structure. This means that the model breaks the dataset into smaller subsets by asking different questions each time. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches (Decisions), each one representing values for the attribute that was tested. Leaf nodes (Terminal Nodes) represent decisions on the numerical targets. The questions and their order is determined by the model itself using Information Gain (for classification) or ID3 (for regression) (Tzirakis and Tjortjis, 2017; Tjortjis and Keane, 2002). For each question, the model must make a strategic split using a criterion. Decision trees are not affected by missing values or outliers. They can handle both numerical and categorical values and they are very easy to understand. Also, trees can capture non-linear relationships. There are some disadvantages though. The most important one is that they tend to overfit to the training sample. A small difference in data might produce a completely different tree. Lastly, there is no guarantee that the tree will be the global optimal.

Random Forest. Random Forest (RF) is another method that uses a tree structure to solve a regression or a classification problem. A random forest is a collection of decision trees, with each tree voting on the final decision. In the training phase, each tree on the forest considers only a random sample of the data. In the prediction phase, each tree makes a prediction and the average of all of the trees will be considered as the final value.

XGBoost. Boosting and bagging are two methods commonly used in weak prediction trees, such as decision trees, to improve their performance. Those two methods work sequentially, meaning that a new model is added to correct the error of the existing models until no further improvements can be made. XGBoost (eXtreme Gradient Boosting) is a method where new models are created, predicting the residuals or errors of existing models and then, they are added together to make the final prediction. Its name comes from the algorithm used to minimize the loss function, which is called gradient descent.

K-Nearest Neighbors. k-Nearest Neighbors (k-NN) is one of the most basic and essential machine learning algorithms. Like the trees, it belongs to the supervised machine learning algorithms. k-NN is a non-parametric method, meaning that it does not make any assumptions about the distribution of the data. k-NN is a fairly simple model that calculates similarities based on the distances between the data points. When a new entry needs to be classified, the algorithm measures the distances between the new data and the already classified data. Then, the new entry is assigned to the class that has the minimum distance to the new data point. There are multiple methods to measure the distance, such as the Euclidean or the Manhattan distance.

LSTM. Simple neural networks cannot understand the context and the order of data. For that, we need some sort of memory. Recurrent neural networks are a special form of neural networks where their units are inter-connected creating various output value dependencies (Hochreiter, 1991). RNNs are extremely important and have been successfully used in a lot of applications, such as speech recognition. But, RNNs suffer from the vanishing gradients problem. This problem refers to the hidden neuron activation functions. If those functions are saturating non-linearities, like the tanh function, then the derivatives can be very small, even close to zero. Multiplying many such derivatives leads to zero meaning that the neural network cannot propagate back for too many instances.

Hochreiter & Schmidhuber (Hochreiter and Schmidhuber, 1997) introduced another kind of recurrent neural networks, the long short term memory (LSTM). Those models have the same ”chain-like” structure, but the module responsible for the ”repetition” part has a different structure. In a classic RNN, the repetition module is a neural network with a hidden layer, usually with tanh as the activation function. On an LSTM, instead of having a single hidden layer, there are four. On the first stage or gate, the neural network decides which information to discard from the cell state. On the second stage, the model incorporates the new information and decides what to keep and what to discard. The model updates the old cell state into the new cell state. In the third stage, the model discards the old information and adds new information. In this stage, the candidate values are estimated. Lastly, the output values depend on the state of the first and the third layer.

3 RESULTS & EVALUATION

This section presents the results of this research. We present the feature selection and the summary of the results per dataset (Sentiment, Economic and PageRank) and per model (DT, k-NN, LSTM, RF and XGBoost).

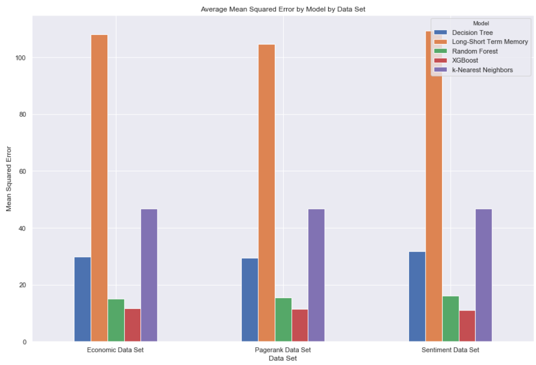

Figure 1: Average Mean Squared Error per Model per Dataset.

All of the scores refer to the mean squared error, thus the best score is the lowest (Figure 1). We evaluate our results using a naive trading strategy and comparing it across all datasets regarding our stocks portfolio (Table 1).

Table 1: Initial Portfolio.

Ticker	Quantity	Price	Amount
AAPL	1	204,5	204,5
CAT	1	139,09	139,09
HD	1	217,26	217,26
UNH	1	264,66	264,66
XOM	1	75,93	75,93
IBM	1	143,53	143,53
TRV	1	154,59	154,59
V	1	179,31	179,31
BA	1	362,75	362,75
INTC	1	49,17	49,17
GS	1	215,52	215,52
JNJ	1	132,5	132,5
WBA	1	55,81	55,81
DOW	1	52,32	52,32
VZ	1	57,41	57,41
JPM	1	115,12	115,12
PG	1	115,89	115,89
KO	1	52,14	52,14
MSFT	1	137,08	137,08
CVX	1	124,76	124,76
MRK	1	81,59	81,59
CSCO	1	57,62	57,62
UTX	1	133,19	133,19
MMM	1	176,49	176,49
WMT	1	114,76	114,76
MCD	1	213,72	213,72
PFE	1	42,85	42,85
AXP	1	128,06	128,06
DIS	1	144,3	144,3

3.1 The Trading Strategy

For evaluating results we utilize a naive trading strategy comprising the following points:

1. At the end of each day we sell the stocks that are predicted to have a loss in the next day.
2. We buy the stocks that are predicted to have a positive return.
3. We choose to buy the one that maximizes the return and we do not take into account variance, estimated error or diversification effects.
4. In the next day we first update the prices and then we calculate gains or losses.

3.2 Feature Selection

This section describes the created features, as well as the descriptive statistics of those features per ticker. It is noted that all of the variables are not available for the day we want to predict, thus all the created features are values of previous days. Since there is no consensus on the literature on which time lag is the most important, for every variable we created the lags from 1 to 3 days prior (Bollen et al., 2011).

One major aspect of this paper is to determine if the sentiment data are noisy and how this can be redeemed. Therefore, we decided to create three different datasets. The first dataset contains the lagged economic variables and the lags of closing prices from previous days. The second dataset contains all the features of the economic dataset as well as the sentiment data. Lastly, the PageRank dataset contains all of the features from the sentiment dataset, but the sentiment variables are multiplied by the PageRank value for each user.

One major drawback of calculating daily PageRank values for each user is that the algorithm does not always estimate the importance for all of the users. Thus, we decided to fill all those dates with the mean value for each user. After that process, we fill all the residual non-estimated PageRank values with 0. This is done since the aim is to have a timely importance measure for the user. In cases where this was not feasible, we theorized that the number of the user followers does not significantly alter from day to day. Therefore, we considered logical to proceed with filling any missing values with their respective mean. Lastly, if there was no mean, the PageRank algorithm did not find any importance in the user for any day. Thus we filled the residual empty values with 0 marking them as noisy and not important.

3.3 Economic Dataset Evaluation

We began with the evaluation of results for the economic dataset and the XGBoost model. Our predictions suggested that we should sell MRK, MCD,

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

MSFT, V, PFE, DOW, JNJ, WMT, DIS, BA, HD, AXP, CAT, IBM, TRV, MMM, JPM, AAPL, NKE, KO, CSCO, GS, and PG and buy four shares of Intel's stock. Our predictions proved correct and Intel's stock recorded a gain, so our portfolio had a total evaluation of $4.041,61$\$. Our decisions for $2019/7/18$ also proved correct and, again, we recorded a gain of $0,30\%$. On the contrary, for $2019/7/19$ our decisions lead to a negative return of $-0,47\%$. The biggest gain was observed on $2019/7/30$ with a daily return of $1,87\%$. Our worst day was the next day, where we lost most of our gains (-$75,99$\$). Finally, our cumulative return for the whole period was positive, $0,75\%$.

3.4 Sentiment Dataset Evaluation

In the sentiment's dataset we began by selling most of our portfolios' stocks and buying only one. More specifically, we sold 23 stocks and bought WBA's stock. This decision was wrong, as we sold Intel's stock, which as we have seen in the previous dataset leads to a significant gain. These decisions naturally lead to a significant loss of $-1,91\%$. Although the next day ($2019/7/18$) our predictions resulted in a daily positive return of $0,26\%$, although it was not enough to overturn the cumulative negative return. Our best return was on $2019/7/29$ with $1,39\%$. Even that return could not reverse our overall losses for this dataset resulting in a cumulative loss of $-3,05\%$.

3.5 PageRank Dataset Evaluation

For the PageRank dataset in the first day, we sold the following stocks, V, MRK, PFE, JNJ, HD, AXP, WMT, MCD, NKE, CAT, TRV, CVX, JPM, MMM, CSCO, INTC, IBM, KO, PG, DIS, and GS. This decreased the value of bought stocks to $1.343,65$\$ and increased the available funds to $2.686,87$\$. At this point, 10 units of ticker UNH were bought at $264,66$ per unit. This updated the value of bought stocks to $3.990,25$\$ and the available funds to $40,27$\$. Since we were still on the same day, the evaluation of the portfolio had not changed, because we had not updated the prices yet. On the next day, after updating the prices, we saw that our portfolio had a value of $4.051,48$\$ meaning that our approach resulted in a positive return of $1,5\%$.

On the second day, we decided to sell the stocks of VZ, AAPL, and UTX and buy three units of Nike's stock. This decision resulted in a loss of $75,88$\$ and a total return of $-1,3\%$. The decision was based on the prediction that Nike's stock would have a positive return. On the contrary, the actual result was a loss of $-1,07\%$. We followed the same strategy for every day. We ended up having two stocks, that of XOM's and Intel's on $25/7/19$. From this point and afterwards, the predictions showed that Intel's stock would have a positive return, so we held on to our stocks. This never happened, and our overall return was negative, resulting in a loss of -122\$ or $-3,03\%$.

Table 2 aggregates daily transactions to top daily losses and gains for the investigated period (2018/11/01 until 2019/31/07) as well as the cumulative returns per dataset. Positive values stand for gains and negative values for losses.

Table 2: Top daily Losses & Gains and Cumulative Returns per dataset.

Dataset	Loss (%)	Gain (%)	Return (%)
Economic	-1,83	1,87	0,75
Sentiment	-1,91	1,39	-3,05
PageRank	-1,87	0,86	-3,01

4 CONCLUSIONS

4.1 Summary

This work addresses the problem of predicting stock market movements. The main contribution resides to the fact that it considers social media as a data source for improving predictions. More specifically it utilizes Twitter data to extract sentiment and investigates whether online sentiment can have a significant positive impact on the forecasting ability of various prediction models. However, these data may introduce biases to the process of result validation due to their noisy nature. To address that, we proposed a new methodology incorporating graphs and obtaining a daily importance measure for all of the users as well as weighting their tweets.

Table 3 summarizes the results for the computed errors of all of the stocks. The PageRank dataset performed better than both the economic and the simple sentiment dataset. Moreover, we were able to confirm that the most important feature, on the sentiment data, is the negative score of the tweet. However, we were not able to confirm which time lag is the most important, since results are highly dependant on the feature.

Five different models were tested. For each stock and for each dataset, we estimated a Decision Tree, a Random Forest, an XGBoost, an LSTM, and a k-Nearest Neighbors. For 15 out of 30 stocks the PageRank dataset performed better than the other datasets. The most important feature of the sentiment data was the negative score. For 13 out of 30 stocks

Table 3: Best Dataset Per Ticker.

Ticker	PageRank	Sentiment	Economic
AAPL			✓
AXP		✓	
BA		✓	
CAT	✓		
CSCO		✓	
CVX	✓		
DIS	✓		
DOW	✓		
GS			✓
HD	✓		
IBM		✓	
INTC			✓
JNJ			✓
JPM		✓	
KO	✓		
MCD			✓
MMM	✓		
MRK	✓		
MSFT			✓
NKE	✓		
PFE		✓	
PG	✓		
TRV	✓		
UNH			✓
UTX			✓
V	✓		
VZ			✓
WBA	✓		
WMT	✓		
XOM	✓		

Table 4: Best Dataset Per Model on PageRank Dataset.

Ticker	DT	k-NN	LSTM	RF	XGBoost
AAPL	✓				
AXP				✓	
BA					✓
CAT	✓	✓			
CSCO				✓	
CVX					✓
DIS		✓			
DOW	✓				
GS				✓	
HD			✓		
IBM			✓		
INTC		✓			
JNJ					✓
JPM				✓	
KO	✓				
MCD					✓
MMM					✓
MRK					✓
MSFT					✓
NKE			✓		
PFE					✓
PG			✓		
TRV					✓
UNH					✓
UTX					✓
V					✓
VZ				✓	
WBA				✓	
WMT					✓
XOM			✓		

the XGBoost performed better than the other models. We could not confirm which time lag is the most important, as this feature was highly depend and on the stock.

Table 4 presents a summarized version of the results in the PageRank dataset. The best model was XGBoost achieving the lowest scores at 13 stocks. Furthermore, it was the most robust model, having the lowest average error and the lowest standard deviation.

Although PageRank's dataset provided the best scores for most of the stocks, during evaluation the only profitable dataset was proved to be the economic $(0,75\%)$. The other two datasets, Sentiment and PageRank recorded losses of $-3,05\%$ and $-3,01\%$, respectively.

4.2 Limitations

This study acts like a proof of concept that microblogging data can be a powerful feature in predicting stock market data, if we can determine and distinguish the important ones. This is feasible but the required data pose an obstacle.

Since all of our data come from the Twint library, and not from the official Twitter API, we could collect a specific amount of tweets. Moreover, this library is significantly slower than the official, thus it was very difficult to obtain data for a longer period. We believe that if we had two years worth of data and all the tweets per day, then our results would be significantly better.

Lastly, on the evaluation part, we choose a greedy strategy and not an optimal one. The optimal solution would require an extra module that would implement diversification according to Markowitz's Portfolio Theorem (Markowitz, 1991) and the extraction of optimal weights per stock. Moreover, every transaction should incrementally position us to more efficient decisions.

4.3 Further Research

There are a lot of aspects in our research that we want to explore in the future. First, we could utilize more models, such as SVM which is commonly used in the literature. Also, we would like to explore other economic variables. There are other such variables that we could embed in our research. Moreover,

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

we could expand our methodology to other financial instruments to explore the possibility that sentiment data can act as features on government and corporate bonds, or even on derivatives. Lastly, as we observed in some models, there were cases where the mean squared error was low, but the fit between the actual and the predicted price was not good. Thus, it would be very helpful if we could define a new measure that can improve the fit capturing.

REFERENCES

Alshahrani Hasan, A. and Fong, A. C. (2018). Sentiment Analysis Based Fuzzy Decision Platform for the Saudi Stock Market. In *2018 IEEE International Conference on Electro/Information Technology (EIT)*, pages 0023–0029, Rochester, MI. IEEE.

Antweiler, W. and Frank, M. Z. (2004). Is all that talk just noise? the information content of internet stock message boards. *The Journal of finance*, 59(3):1259–1294.

Beleveslis, D., Tjortjis, C., Psaradelis, D., and Nikoglou, D. (2019). A hybrid method for sentiment analysis of election related tweets. In *2019 4th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, pages 1–6. IEEE.

Blystone, D. (2019). Overbought or oversold? use the relative strength index to find out.

Bollen, J., Mao, H., and Zeng, X.-J. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8. arXiv: 1010.3003.

Chakraborty, P., Pria, U. S., Rony, M. R. A. H., and Majumdar, M. A. (2017). Predicting stock movement using sentiment analysis of twitter feed. In *2017 6th International Conference on Informatics, Electronics and Vision & 2017 7th International Symposium in Computational Medical and Health Technology (ICIEV-ISCMHT)*, pages 1–6. IEEE.

Devenow, A. and Welch, I. (1996). Rational herding in financial economics. *European Economic Review*, 40(3-5):603–615.

Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business*, 38(1):34–105.

Fama, E. F. and French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of financial economics*, 33(1):3–56.

Hagberg, A., Swart, P., and S Chult, D. (2008). Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Hájek, P. (2018). Combining bag-of-words and sentiment features of annual reports to predict abnormal stock returns. *Neural Computing and Applications*, 29(7):343–358.

Hochreiter, S. (1991). Investigations on dynamic neural networks. *Diploma, Technical University*, 91(1).

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international AAAI conference on weblogs and social media*.

Koukaras, P. and Tjortjis, C. (2019). Social media analytics, types and methodology. In *Machine Learning Paradigms*, pages 401–427. Springer.

Koukaras, P., Tjortjis, C., and Rousidis, D. (2019). Social media types: introducing a data driven taxonomy. *Computing*, pages 1–46.

Kuepper, J. (2019). Timing trades with the commodity channel index.

Malkiel, B. G. and Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, 25(2):383–417.

Markowitz, H. M. (1991). Foundations of portfolio theory. *The journal of finance*, 46(2):469–477.

Mitchell, C. (2019). Aroon oscillator definition and tactics.

Mittermayer, M.-a. and Knolmayer, G. (2006). NewsCATS: A News Categorization and Trading System. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 1002–1007, Hong Kong, China. IEEE.

Oikonomou, L. and Tjortjis, C. (2018). A method for predicting the winner of the usa presidential elections using data extracted from twitter. In *2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM)*, pages 1–8. IEEE.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Rousidis, D., Koukaras, P., and Tjortjis, C. (2019). Social media prediction: A literature review. *Multimedia Tools and Applications*.

See-To, E. W. K. and Yang, Y. (2017). Market sentiment dispersion and its effects on stock return and volatility. *Electronic Markets*, 27(3):283–296.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442.

Sohangir, S., Petty, N., and Wang, D. (2018). Financial Sentiment Lexicon Analysis. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, pages 286–289, Laguna Hills, CA, USA. IEEE.

Staff, I. (2019). On-balance volume: The way to smart money.

Tjortjis, C. and Keane, J. (2002). T3: a classification algorithm for data mining. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 50–55. Springer.

Treynor, J. L. (1962). Jack treynor's' toward a theory of market value of risky assets'. *Available at SSRN 628187*.

Tzirakis, P. and Tjortjis, C. (2017). T3c: improving a decision tree classification algorithm's interval splits on continuous attributes. *Advances in Data Analysis and Classification*, 11(2):353–370.