Predicting Stock Market Price Movement using NLP on Reddit Data

CSC715M Project

Sidney Guaro
De La Salle University
sidney_guaro@dlsu.edu.ph

***Abstract*—There have been a lot of discussions about the financial market in Reddit and WallStreetBets. Conversations on these possibly have the ability to influence the stock market. The goal of this project is to leverage the Reddit text data to predict a stock market movement. To utilize the text data, I employed sentence embedding, document embedding, and Neural Network (NN) models. Accordingly, this project examines several NN architecture and text data. Results show that text-based model can outperform the naive forecasting approach.**

***Index Terms*—Stock market prediction, natural language processing, sentence embedding, document embedding, neural networks**

I. Introduction

Some stock market price movements today are influenced by social media like Reddit. These stocks have a high gain or loss, owing primarily to the individual online investors. In the subreddit WallStreetBets, it is thought to have started a short squeeze on GameStop (GME) stock recently, making it one of the most shorted stocks on WallStreetBets, with 102% of its shares shorted [2]. Following these events, Reddit has received a lot of attention making other stocks worth looking into whether they've been impacted by the posts.

With the use of Natural Language Processing (NLP), the goal of this project, as with [1], is to answer the question: Does the text data on Reddit provide enough information to predict a stock market price movement? To that aim, I created NN models to predict the stock price movement. I used the embedding modules as inputs. These modules first convert the text data into the following embeddings: a) Sentence Embedding based on BERT [3] [4] using title data b) body document embedding based on Doc2Vec [5] c) number of stock mentions. These embeddings are then fed to Convolutional Neural Networks (CNN)s to predict a stock price movement.

II. Related Work

In representing the text data, it is important to have an accurate vector representation for models to train on. There have been a lot of works in this context for sentence embeddings. [6] proposed Sent2Vec, an efficient model for sentence embedding that learns for the n-grams. [7] is using a supervised learning method to generate embeddings on global vectors. [8] uses a transformation to create an embedding, and the variation Sentence-BERT [3] creates semantically relevant sentence embeddings using a siamese and triplet network topology. [5] introduced Doc2Vec that represents a document in an embedding.

There are also many models that predict stock market movement using social media feeds. On Twitter, [9] uses SVM and [10] uses Hybrid Naive Bayes Classifiers to classify sentiments on stocks. [11] uses SBERT to extract the tweet sentiment and along with it, predict the stock price movement. [12] uses Boosted Regression Tree on a similar approach. On Reddit, [14] explores 24 related Reddit communities to extract 112 time series features to predict cryptocurrency movement. [1] uses different embedding techniques that afterward are trained on a CNN model using Reddit data. [15] [16] shows that there is a relationship between the Reddit posts about GME and its stock price.

TABLE I
Model Results of the Reference Paper

#	Method	Training Period	Testing Period
0	Baseline	47.9%	46%
1	Sentence Embedding Averaging Model	61%	51.4%
2	Sentence Embedding CNN Model	100%	57%
3	Document Embedding CNN Model	91%	54%
4	Sentence Embedding CNN + Sentiment Model	93%	30%
5	Document Embedding CNN + Sentiment Model	83%	46%
5	Document & Sentence Embedding CNN + Sentiment Model	98%	51%

The main reference of this project is [1]. In the paper, the author has explored sentence embedding using SBERT [3], document embedding using Doc2Vec [5], sentiment analysis using VADAR and TextBlob, averaging sentence embedding, and CNN model that is trained on embeddings. The author used several stocks that are mentioned per day. These approaches are used to predict the stocks' movement. The author also combined different approaches to further increase the accuracy in stock movement prediction. Table I shows the results of different models in [1]. Results show that there are cases of overfitting specifically for the CNN models. The author also noted that it is worth exploring hourly frequency on an individual stock that is highly impacted by the Reddit posts.

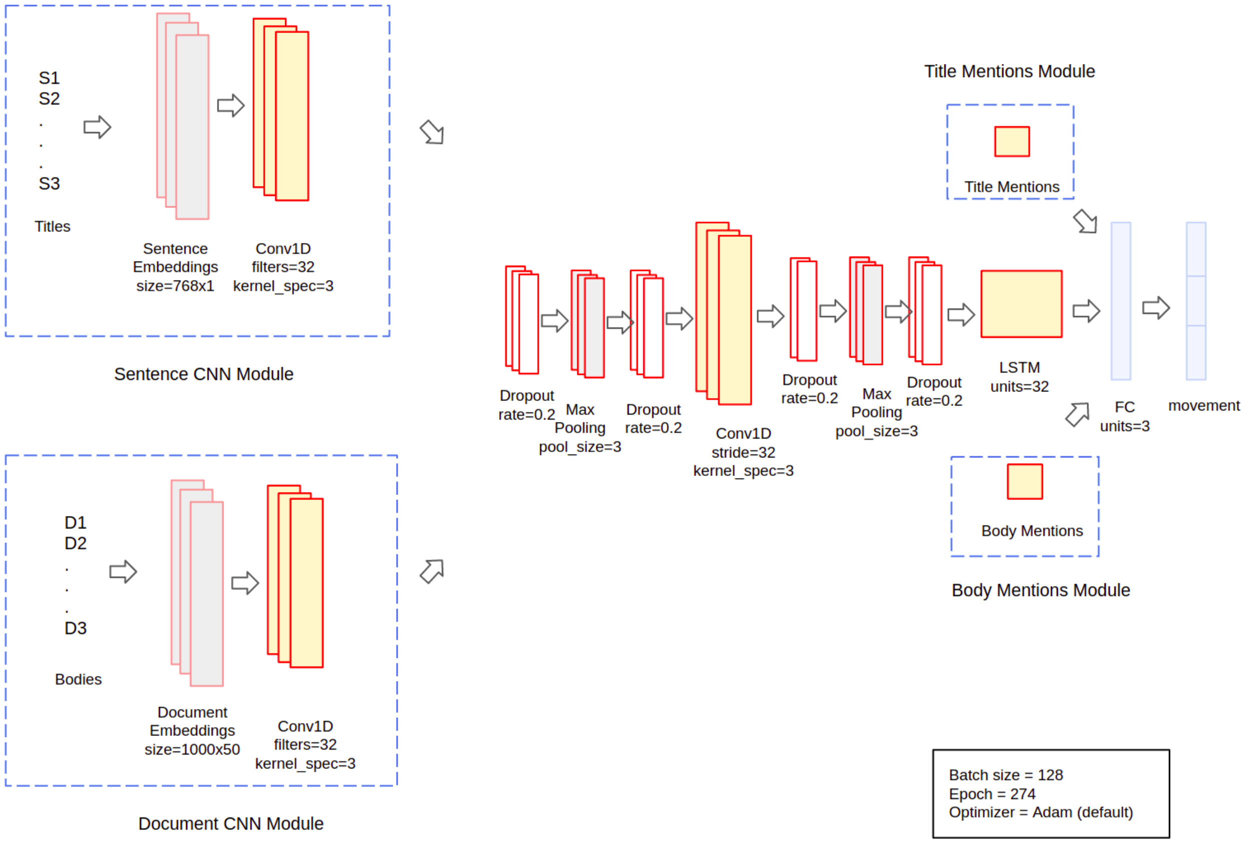

Fig. 1. Model structure of the Stock Price Movement

Given the articles pertaining to the relationship of WallStreetBets and GME stock price movement along with the work of [1], the aim of my project is to create a predictive model using the hourly posts from WallStreetBets on a specific stock. I will be using SBERT and Doc2Vec to train the CNN model. As mentioned in [1], the highest accuracy score recorded is SBERT. On the other hand, with a lot of text data in an hour, Doc2Vec may give a compressed representation.

III. Approach

A. Exploratory Data Analysis

I completed the following data processing and exploration procedures to aid model development:

- Data cleaning: data that are collected are cleaned by converting it to lower case, and removing web links, symbols, and stop words.
- Data exploration: I listed the US stocks and extracted the counts on title and body. I look at then the stocks that are mostly mentioned that are aided with financial news about Reddit stocks.
- Model target: In terms of the model target, I choose the market movement, which is calculated as

$$Movement_t = \begin{cases} 1 & x > r \\ 0 & r > x > -r \\ -1 & x < -r \end{cases}$$

where $x = Open_t - Close_t$ and r is the threshold, in this case 0.003. The $Movement_t$ can be designated with the up, neutral, down movement. This is different from [1] as I am interested in the movement, not the actual price. It is also reasonable to look at the movement because a small change in the price can be disregarded as charges might be higher.

B. Embeddings

Like [1], I used the SBERT to transform texts into sentence embeddings [3]. I fine-tuned [4] to generated embeddings with dimensions of 768.

I also generated a 50 vector sizes document embeddings using the body trained in Doc2Vec [4].

C. Neural Networks

On top of the embedding, I used NN models according to these modules as shown in figure 1. The following are the modules used:

- Sentence CNN Module: This module uses title as input to the SBERT. The generated sentence embedding vectors are fed into a CNN model. Information is then extracted using the CNN model.
- Document CNN Module: Rather than sentences, body text is converted to document embeddings then used as inputs to the CNN model. A CNN model is then utilized like the sentence embedding.
- Title Mentions Module: Apart from the title text data, I am interested in how does the number of title mentions affect the accuracy of the model.
- Body Mentions Module: Like Title Mentions CNN Module, I am interested in the relationship between body mentions and accuracy.

Using the modules, I have built the NN models:

- Sentence NN Model: To forecast market movement, this model employs a Sentence CNN module and a neural network architecture to output the market movement.
- Document NN Model: This is similar to Sentence CNN Model however Document Module is used instead.
- Sentence + Mention NN Model: This is expanded Sentence NN Model, combined with Title Mentions module.
- Document + Mention NN Model: Like Sentence NN Model, this is combined with Body Mentions module.
- Sentence + Document + Mention NN Model: I also combined the Sentence + Mention NN Model and Document + Mention NN Model to see if there is an improvement on the performance.

D. Baseline

Similar to [1], I used the naive forecasting approach. This is to anticipate the future using current market price or condition. The equation for the forecasting in this project is $Movement_t = Movement_t - 1$.

IV. Results and Discussions

A. Data

For this project, I collected Reddit data using Pushshift.io [18]. The frequency of the posts from WallStreetBets is in hours containing title, body, and time. Afterward, I picked the Wipro Limited (WIT) to work on. According to [20], WIT can be associated with WallStreetBets posts. Additionally, WIT is also at the top 100 mentioned stocks with in the time range. I also added the market data to the collected text data. The hourly price data for the SP500 index is obtained from Yahoo Finance [19].

The summary of the data is as follows:

- Time Range: January 01, 2020 to September 13, 2021
- Number of posts: 1,395,676
- WIT mentions: 111,427
- Number of historical prices: 2,983
- WIT text and price data: 240,138

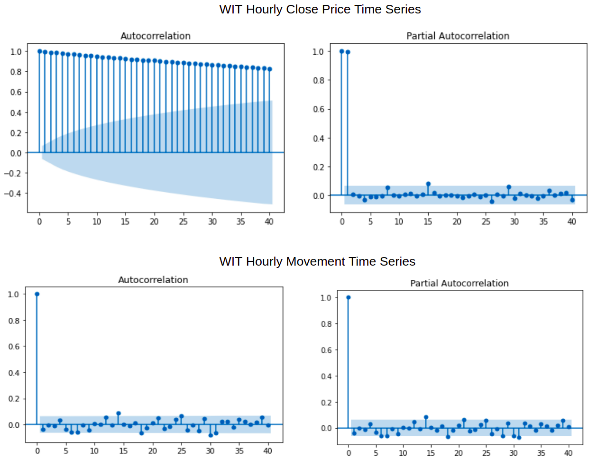

Fig. 2. WIT Time Series Analysis

For the data split, since the model is based on time series, it is possible that a random split of the train and test dataset is not applicable. Thus, I divided my data into two categories: in and out of time. I use all the data in 2020 as training set and 2021 as testing set. Despite the amount of data, I am limited to the data of training set and computation. As a result, I down-sampled the dataset. The result of the split is defined as follows:

- Training row count: 36,865
- Testing row count: 203,273
- Training/Testing up movement row count: 15,000
- Training/Testing neutral movement row count: 5,000
- Training/Testing down movement row count: 15,000

B. Baseline Analysis

For the baseline to work, WIT price and movement should have a strong correlation with historical performance. As illustrated by the autocorrelation and partial autocorrelation in figure 2, the WIT price and movement have a significant association with past performance.

C. Experiment Results

Table II shows the accuracy of the train and test set. Here are some of my observations based on the results of the experiment:

- There is an improvement in the accuracy with the inclusion of number of mentions.
- Sentence embedding performs better than document embedding given that it is only based on title. The reason for this is that document embeddings are mixed with different day comments thus affecting the accuracy of the model. This is a constraint in the data collection as pushshift.io api converts the comments into a single text blob making the different day comments undifferentiated.
- All the models are underfitted. The cause of this is the architecture of the models. Since the models only has 32 filters, it does not capture the whole vector embeddings.
- The models of this project achieved lower accuracy as compared to the results of [1] however, this is for a single stock. The reported accuracy of the main reference paper

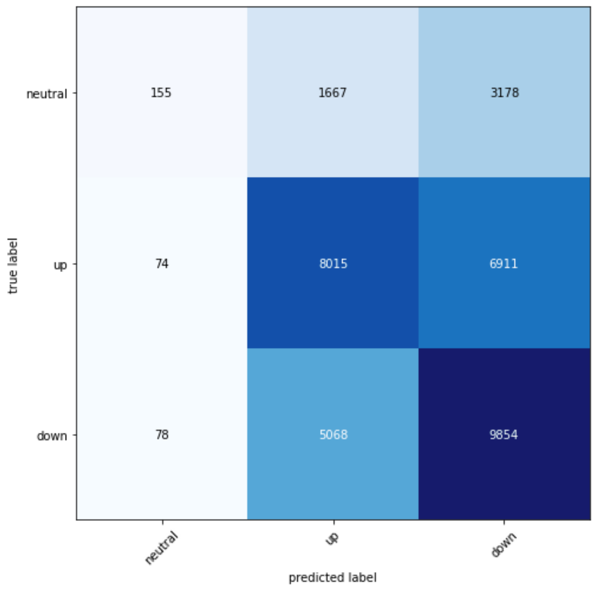

Fig. 3. Confusion Matrix of Sentence + Mention NN Model

is the stocks mentioned per day. Also, this is for the case of down-sampled dataset.

TABLE II
MODEL RESULTS

#	Method	Training Period	Testing Period
0	Baseline	39%	37.4%
1	Sentence NN Model	49%	47%
2	Document NN Model	47%	44.6%
3	Sentence + Mention NN Model	49%	51%
4	Document + Mention NN Model	50.5%	46.7%
5	Sentence + Document + Mention NN Model	40.6%	38%

In addition to the results, I investigated the model with highest accuracy, Sentence + Mention NN model. The model has difficulties in differentiating up and down movement as shown in figure 3. Furthermore, figure 4 shows that up and down sentences have close distances. This suggests an improvement in the distinction between up and down.

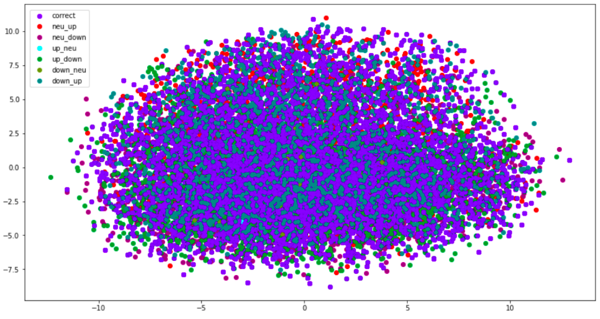

Fig. 4. Sentence Mention Embedding Space

V. CONCLUSION

The goal of this project is to forecast a stock price movement through Reddit text data. Consequently, I propose a NN architecture that takes outputs the stock price movement of a stock. I define its components as modules that take hourly text inputs as title and body, and numerical inputs that take title mentions and body mentions. Before feeding into the NN model, the text inputs are converted to learned representations namely sentence embeddings and document embeddings. With the modules, I generated different combinations of NN models. Results show that the model-based NLP technique outperforms the naive forecasting method. The model's performance can also be improved by adjusting the model's complexity and modifying the threshold value defined for the price movement. This is also true in adding features such as post score or other time series price features. In terms of data, increasing the time range and filtering the data to contain only sentences with mentioned stocks can possibly improve the model's performance as well.

REFERENCES

[1] M. Xu, "NLP for Stock Market Prediction with Reddit Data", February 2021.

[2] J. Hadfield, "Exchanges ban Trades, Biden ADMINISTRATION 'MONITORING SITUATION' after Reddit Drives Wall Street Hedgefunds to brink of bankruptcy in GAMESTOP SHORT," National File, 28-Jan-2021. [Online]. Available: https://nationalfile.com/exchanges-ban-trades-biden-administration-monitoring-situation-after-reddit-drives-wall-street-hedgefunds-to-brink-of-bankruptcy-in-gamestop-short/. [Accessed: 17-Sep-2021].

[3] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence EMBEDDINGS using siamese bert-networks," Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019.

[4] N. Reimers and I. Gurevych, "Making monolingual sentence embeddings multilingual using knowledge distillation," Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020.

[5] E. M. Ponti, I. Vulić, and A. Korhonen, "Decoding sentiment from distributed representations of sentences," Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017), 2017.

[6] M. N. Moghadasi and Y. Zhuang, "Sent2vec: A new sentence embedding representation with sentimental semantic," 2020 IEEE International Conference on Big Data (Big Data), 2020.

[7] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep BIDIRECTIONAL transformers for language understanding," arXiv.org, 24-May-2019.

[9] S. V. Kolasani and R. Assaf, "Predicting stock movement using sentiment analysis of twitter feed with neural networks," Journal of Data Analysis and Information Processing, 29-Sep-2020.

[10] G. A. A.Jabbar Alkubaisi, S. S. Kamaruddin, and H. Husni, "Stock market classification model using sentiment analysis on twitter based on hybrid naive bayes classifiers," Computer and Information Science, vol. 11, no. 1, p. 52, 2018.

[11] M. Jenny, "Using Twitter Sentiment for Stock Movement Prediction and Portfolio Optimization," 2021.

[12] P. Chakraborty, U. S. Pria, M. R. Rony, and M. A. Majumdar, "Predicting stock movement using sentiment analysis of twitter feed," 2017 6th International Conference on Informatics, Electronics and Vision, 2017 7th International Symposium in Computational Medical and Health Technology (ICIEV-ISCMHT), 2017.

[13] S. Wooley, A. Edmonds, A. Bagavathi, and S. Krishnan, “Extracting cryptocurrency price movements from the reddit network sentiment,” 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 2019.

[14] S. Wooley, A. Edmonds, A. Bagavathi, and S. Krishnan, “Extracting cryptocurrency price movements from the reddit network sentiment,” 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 2019.

[15] S. Lindskog and J. A. Serur, “Reddit sentiment analysis,” SSRN Electronic Journal, 2020.

[16] A. Anand and J. Pathak, “Wallstreetbets against wall street: The role of reddit in the gamestop short squeeze,” SSRN Electronic Journal, 2021.

[17] C. Hutto and E. Gilbert, “VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text”, ICWSM, vol. 8, no. 1, pp. 216-225, May 2014.

[18] Dmarx, “Dmarx/Psaw: Python pushshift.io api wrapper (for comment/submission search),” GitHub. [Online]. Available: https://github.com/dmarx/psaw. [Accessed: 18-Sep-2021].

[19] Ranaroussi, “Ranaroussi/Yfinance: Yahoo! finance market data DOWNLOADER (+faster PANDAS DATAREADER),” GitHub. [Online]. Available: https://github.com/ranaroussi/yfinance. [Accessed: 18-Sep-2021].

[20] C. Lau InvestorPlace, “7 best Reddit penny stocks to buy if you have $500 to spend,” Nasdaq. [Online]. Available: https://www.nasdaq.com/articles/7-best-reddit-penny-stocks-to-buy-if-you-have-%24500-to-spend-2021-06-02. [Accessed: 18-Sep-2021].

[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic MINORITY OVER-SAMPLING TECHNIQUE,” Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.