Evaluating the predictive performance of Reddit posts on American stock prices

Erik Lundin

January 15, 2022

E-mail: erilu186@student.liu.se

GitHub repository: https://github.com/ErikLundin98/text_mining_Reddit

Abstract

The problem of predicting future stock prices is a popular research topic. Traditional approaches resort to time series analysis and quantitative modeling, but recent methods using text mining and natural language processing approaches have shown promising results as well, as they are able to leverage information stored in text that contains general sentiment and beliefs that the public holds towards stocks. This project approaches the issue of stock price prediction by formulating it as a binary text classification problem, where the stock either goes up or down, and uses sentiment analysis, document vectorization and metadata from Reddit posts to train a classifier to predict price movements. Furthermore, the project examines if a moving average sentiment score in the form of a polarity score on its own can predict stock prices. The experimental results indicate that simply using a moving average of the sentiment of Reddit posts related to stocks is not a strong predictor of future stock prices. However, a linear SVM classifier trained on features extracted from Reddit posts can achieve a test accuracy of 90% when trained and evaluated on data from the year of 2021, which is about 30% higher than the naïve majority classifier baseline.

Introduction

On the popular discussion website known as Reddit[1], people can engage in discussions and news in so called subreddits. A subreddit generally encapsulates a certain topic of conversation, such as cute animals, funny videos and investing. In recent times, one such subreddit that has gained traction in media is r/wallstreetbets, a subreddit centered around

[1] https://www.reddit.com/

investing mainly in U.S. listed stocks, that has over 11 million members as of January 15, 2022. One of the reasons for this traction is that the members collectively held a significant amount of shares in GameStop Corporation (GME) during its surge in early 2021. This resulted in the price reaching never before seen heights and subsequently giving owners of the stock a good return on their investment. It did however also result in a high volatility of the stock, which gained 1600% during January alone and subsequently lose more than half of its value in February[19], which naturally affected investors that used posts in the subreddit as investment advice negatively. It is generally accepted as a fact that the members of the subreddit jointly contributed to the surge of GME:s price [5][1]. Furthermore, Long et al. [13] show that there was a connection between short term developments of GME:s stock price and Reddit comments in r/wallstreetbets during the beginning of 2021. Due to this, it is interesting to explore if there is a causality between the opinions and information shared in these subreddits on different stocks and the stocks' future performance in terms of price growth; if this would be the case, the Reddit posts could be used as a basis point for a profitable trading strategy.

There are other subreddits related to investing as well, as r/gme, r/investing and r/stockmarket. These subreddits generally contain investors that use different styles when investing: r/wallstreetbets' members are known for seeking risky investments that have a chance to yield very high returns short term, while r/investing contain a mix of members of different investing styles, some are more risk averse and share their thoughts on investments that are safe and yield long term returns. There is an interesting rivalry between the subreddits as the investing styles differ greatly between them. For a person looking to decide which investing style suits them best, it could therefore be of interest to compare them and see if one subreddit seems to be better at predicting stock prices than another.

Aim

The aim of this project is to examine if posts made in investing subreddits on the Reddit platform can be used to predict if stock prices will go up or down in the future. Furthermore, the project explores if some subreddits are better than others at predicting the stock price movements.

Research Questions

Throughout the report, the following research questions will be answered:

1. Is the general sentiment of a stock in a certain subreddit a good indicator of its future performance?
2. Is there a difference in accuracy of the general sentiment and stock price movements for different subreddits?

3. Can a classifier that is trained on sentiment scores and document embeddings of subreddit posts be used to successfully predict if stock prices will go up or down?

Limitations & Assumptions

Due to the limited time frame of this project, some limitations and assumptions have been made. The predictive performance is considered a binary classification problem where the label is 1 if the price of a stock has increased from the current price. The label is -1 if the price has decreased. Furthermore, when comparing the subreddits' predictive performance (research question 1 and 2), it is assumed that the polarity score from a sentiment analysis model corresponds to the belief that a stock will go up, i.e. if the over all polarity score is negative, the stock is expected to go down, and if the score is positive, the stock is expected to go up.

The Reddit posts' predictive performance will only be evaluated on a daily basis, on stock prices (daily close prices) that are observed 1, 3, 7 or 14 days after the date that the prediction is made. In the method chapter of the report, further assumptions are made which will be discussed in the discussion chapter.

Theory & Related Work

This chapter covers the theory that will be used, as well as relevant previous work. It is assumed that the reader has a basic understanding of core concepts related to text mining.

Sentiment Analysis

Sentiment analysis is defined by Ronen Feldman[4] as "the task of finding the opinions of authors about specific entities". Sentiment analysis performed on documents is a common form of sentiment analysis, in which a long or short text can for instance be assigned a polarity score, or a label denoting if the text is "positive" or "negative". A lower level of sentiment analysis is sentence-level sentiment analysis in which a single sentence is analyzed at a time. Sentence-level sentiment analysis is often a part of document analysis, as polarity scores and other metrics for the sentences are aggregated to create a sentiment score for the whole document.

There are many ways to create sentiment analysis models; two successful approaches are rule-based models [11] and using deep neural networks such as BERT:s [3]. BERT is a Bidirectional Transformer encoder that is capable self-attention, which means that it by itself can weigh the importance of each input (token). One of the model's strengths lies in its ability to learn dependencies between words that are distant to each other, which separates it from more classical models which might be limited to handling words that are directly adjacent to each other (bigrams). The BERT model has led to significant advances

in natural language processing, including sentiment analysis [2][18]. BERT is a pre-trained model[3], but it can be fine-tuned by training it on domain-specific corpuses to perform even better in specific applications.

Financial Sentiment Analysis with FinBERT

One such specialization of the BERT model is FinBERT, which is designed for NLP tasks within the financial domain[2]. The model has been further pre-trained on a financial corpus called TRC2-financial. The corpus contains 46,143 news articles published by Reuters. To train the model on sentiment analysis, the Financial PhraseBank dataset was used. This dataset contains annotated sentences from financial news, where labels correspond to how the information in each sentence from a subjective standpoint can affect the mentioned stock's price. The labels consist of "Positive", "Negative", "Neutral". FinBERT showed a 86% accuracy when tested on 20% of the PhraseBank dataset and outperformed the baseline methods, which include an LSTM model[7], ULMFit[9] and ELMo[17]. The model handles a maximum token amount of 512, and outputs a logit for each label (positive, negative & neutral)[2]

Social Media Sentiment Analysis with VADER

Another sentiment analysis model is VADER introduced by Hutto et al[11]. The model is rule-based and specifically developed for sentiment analysis on text in social media. It uses a sentiment lexicon that contains commonly used words, emoticons and acronyms that have been manually rated from -4 to +4. As examples, "okay" has the rating of 0.9, ":(" is rated -2.2 and horrible is rated -2.5. Tokens that are not included in the dictionary (which consists of 7.500 tokens in total) are scored with a 0. VADER also incorporates grammatical rules that take into consideration that some (neutral) words can significantly affect the polarity of a whole sentence. Consider for example "it is good" versus "it is not good". When determining the aggregate sentiment score for a text, the individual tokens are scored based on the sentiment lexicon, then modified according to the grammatical rules.[11]

The output scores of the model are "positive", "negative", "neutral" and "compound". Positive, negative and neutral correspond to the share of words in the text that have been scored > 0, < 0 and $= 0$. Compound is an aggregate score that takes the grammatical rules into account to create an overall score, which is normalized to the range (-1, 1).[11]

Related Work

There is related work that explores the predictive power of reddit sentiment. In a Bachelor's thesis by Michael Lubitz [14], it is explored how a moving average of the sentiment depicted in Reddit comments posted in the subreddit r/economics can be used to predict the closing value of the S&P 500 index on the same day. The thesis explores calculating the moving average average sentiment by weighting comments by weighting the comments based on

upvotes and amount of comments made on the comments, and finds that by training a classifier on this moving average, an accuracy of around 55% can be achieved[14]. This study shows that it is worth exploring how Reddit sentiment correlates with stock price movements, but it makes a naïve assumption that all posts in r/economics are directly related to the S&P 500 index. If one could separate comments (or posts) related to the index from posts not related to the index, there is a possibility that the accuracy could be improved. Furthermore, it is worth investigating if the same predictive performance can be achieved on other financial assets, i.e. other indices and company stocks.

In another article by Hu et al. [10], it is explored through graph data collected from r/wallstreetbets if four measures; traffic volume, discussion tone, dispersion of opinions and connectedness of submitters, have a connection with stock price movements. it is found that a "higher traffic, more positive tone of posts and comments, and higher connectedness at Reddit lead to higher returns, high retail order flow, and lower shorting flows in the future"[10]. The study specifically investigated the predictive performance of these metrics on 50 select stocks and also shows that there seems to exist a causality between Reddit discussions and stock prices. The study does however limit itself to using these four metrics as descriptors.

The above articles show promising results and that there seems to exist a connection between Reddit discussions and stock prices. Existing articles do not compare different subreddits' performance in this regard to each other, and limit themselves to examining a single subreddit. The method that use machine learning models only aims at predicting prices for one instrument in one subreddit[14], In [10], the study is limited to using 4 metrics and only looks at if there is a connection between the metrics and prices, not how well a machine learning model can predict stock prices based on the metrics.

Data

The dataset of Reddit posts that is used in this project is curated by user leukipp on Kaggle[2]. The dataset contains of Reddit posts from 14 different subreddits. The data is structured in a table format. The features that were used in this project are listed below with the following features:

- created (date): Time of the post being posted
- title (string): Title of the post
- selftext (string): Text of the post
- upvote_ratio (float): Score between 0-1 indicating #upvotes/#total votes

[2] `https://www.kaggle.com/leukipp/reddit-finance-data`

- score (uint): Total amount of upvotes
- gilded (uint): The amount of "Reddit gold" the post has been awarded
- total_awards_received (uint): The amount of awards the post has received
- num_comments (uint): The total amount of comments made on the post

All posts are posted during 2021. Before cleaning the dataset, this is the distribution of posts between different subreddits:

subreddit	#posts
personalfinance	96556
gme	95918
wallstreetbets	33892
stocks	25520
options	15554
pennystocks	14880
stockmarket	7828
investing	6097
total	199689

Table 1: Distribution of posts for the different subreddits

When it comes to stock prices, a dataset containing a list of 7797 stocks listed listed on the US Nasdaq Stock Exchange is used[15]. The dataset consists of both the symbol (1-5 letters), and the name of the listed company. To retrieve daily close prices for relevant stocks, Yahoo finance was used.

Method

This section describes the process of processing and combining the datasets, as well as how to replicate the experimental results of the project.

Data Processing and Cleaning

Before the experiments can be made, the two datasets of listed stocks and Reddit posts need to be cleaned. The purpose of the listed stock dataset is to use it to recognize the organization associated with the stock in the Reddit posts' texts and titles. Therefore, the stock dataset should contain stock symbols and company names as they are written in social media. To solve this, manual processing is done to the data in the following manner:

Words and abbreviations matching common corporate suffixes are removed from company names. The suffixes that are removed are: 'Inc', 'Ltd', 'S.A.', 'SE', 'Corp', 'Corporation', 'Incorporated', 'Class A', 'Common Stock', '(', 'plc', 'PLC', 'Limited','II', 'REIT', 'ADS', 'Co.', 'BDC', 'Enterprise', 'B.H.N.', 'S A', 'INC.', '.com'. For instance "Best Company Incorporated" becomes "Best Company". Entries with symbols containing '^' are removed completely as they correspond to funds and not stocks. Lastly, entries missing a company name are removed. Similarly entries in the Reddit dataset that are missing either a title or text are removed. The dataset, which contains separate files for each subreddit, are concatenated to a single file.

Matching Reddit Posts with Stocks

To find Reddit posts that discuss a certain stock, SpaCy's Named Entity Recognizer[8] is used. For each entry in the dataset, all entities that are classified as an organization ('ORG') by SpaCy's "en_core_web_lg" model[3] are extracted. If any of these match with either the symbol or company name in the stock dataset, the post is considered to be related to that stock. In the case where a single Reddit post matches with multiple stocks, the whole title and selftext of the post is considered to be equally related to all of those stocks. The symbols corresponding to the matching stocks for each post are added as a feature to the dataset. After this, all Reddit posts that do not match with any stock are removed.

Creating Labels for Stocks' Closing Prices

As a large majority of the stocks are only mentioned in a small amount of Reddit posts, only the 100 most discussed stocks (i.e. stocks with the highest amount of occurrences) are considered. After this filtering has been done, daily closing prices for these 100 stocks are fetched for dates 2021-01-01 to 2021-12-25. To create labels a rolling calculation is performed for each observed date. Four labeled datasets are created corresponding to four time horizons: 1 day, 3 days, 7 days and 14 days. For each date t, stock and dataset time horizon $t + n$, the label is -1 if the price at $t + n$ is below the price at t, and 1 if the price at $t + n$ is above t. When the price at $t + n$ is the same as at t, the date is ignored.

Performing Sentiment Analysis on Reddit Posts

Both VADER and FinBERT are used to assign sentiment score to the Reddit posts. For each post in the dataset, VADER is used through the NLTK library[4] in Python. VADER's Sentiment Intensity Analyzer is used to calculate polarity score for the title and text separately. The polarity score yields a negative, neutral, positive and compound score, these 4 scores are appended to the dataset for both the title and text.

[3] `https://spacy.io/models/en`

[4] `https://www.nltk.org/_modules/nltk/sentiment/vader.html`

FinBERT is also used, in the following manner:

- FinBERT's pretrained model and tokenizer are downloaded from HuggingFace's library of transformer models
- The positive, negative and neutral logits are calculated and softmaxed for the title and text of each post separately, and also appended to the dataset

In total, 14 new columns are added to each row of the dataset.

Method for Answering RQ1 and RQ2

To determine if the sentiment towards stocks in the subreddits can be a good indicator for future stock growth, a general sentiment to a stock needs to be aggregated for a given day. In other words, for each subreddit, date and stock, a sentiment score is assigned. This is done by calculating moving average sentiments scores for each stock and sentiment analysis model and is described below:

$S_{subreddit} \leftarrow \{personalfinance, gme, wallstreetbets,$
$stocks, options, pennystocks, stockmarket, investing, all\}$
for $W_{sentiment} \in \{1d, 3d, 7d, 14d\}$ **do**
  for $S \in S_{subreddit}$ **do**
    for $M \in \{VADER, FinBERT\}$ **do**
      for each stock **do**
1. Calculate the mean of each sentiment score (positive, negative)
for each date for model M
2. Calculate the moving average means of the $W_{sentiment}$ last dates for each date
3. Set the prediction for that stock to 1 if the moving average mean
is higher for positive, -1 otherwise
      end for
    end for
  end for
end for

Additionally, another aggregation method is tried where each Reddit post in the moving average is weighted according to its amount of upvotes (this replaces step 1 in the pseudocode above). Due to the somewhat complex method, the reader is asked to refer to the linked repository with code that demonstrates how the method is implemented. The predictive accuracy is then calculated for each subreddit (and all subreddits combined) by for every combination of sentiment window size ($W_{sentiment}$, stock price time horizon (1,

3, 7 or 14 days) and each subreddit calculating the accuracy of the predictions for each of the models. A total accuracy is then calculated for each subreddit by calculating the weighted average of each accuracy for each stock. In total, this results in one accuracy for each subreddit subset (including all subreddits combined), stock price time horizon and moving average sentiment window. In other words, there are 9x4x4 = 144 accuracies reported.

A baseline comparison here is selected to be a classifier that always believe that the stock will go up, i.e. always predicts a 1.

Method for Answering RQ3

To answer Research Question 3, whether a classifier can be trained on the Reddit posts to accurately predict stock prices, the used data is limited in the following manner:

- The classifier is trained on Reddit posts from one of the following subsets of subreddits: r/wallstreetbets, r/gme, r/stocks and a combination of all three
- The classifiers are trained to predict a single stock price at a time, on three different stocks (the 3 most commonly discussed stocks), which are GME, AMC and AMZN.

The reason for making these limitations is that the resulting dataset for each stock and subreddit becomes too small if other stocks or subreddits are used. The dataset is further processed to make it suitable for training the classifier:

- Posts from other subreddits than the three mentioned above are removed
- Posts not mentioning one of the three stocks above are removed
- The "selftext" and "title" columns are combined into a single column by concatenating the title and text and separating them with ". ".
- The concatenated text column is vectorized using the vectorizer included in SpaCy's "en_core_web_lg" model which creates a document embedding of size (300, 1)
- Now that the dataset only contains numeric values, the dataset is grouped by the creation date by calculating the mean value of each column for each date and the moving average is calculated for each column based on window sizes 1d, 3d, 7d and 14d. Note that the average is not calculated by weighting the values based on votes, but simply by taking the mean of all features in posts made on the same date.

After the data is processed, it always contains at most 365 values as it contains one set of features per day in 2021. For each of the three stocks, the dataset contains the following columns:

To evaluate if a classifier can be successfully trained to predict stock prices based on the

'created'	'finbert_title_pos'	'nltk_text_neu'
'gilded'	'finbert_title_neu'	'nltk_text_pos'
'total_awards_received'	'finbert_text_neg'	'nltk_text_comb'
'num_comments'	'score'	'finbert_title_neg'
'nltk_title_neg'	'subreddit'	'finbert_text_pos'
'nltk_title_pos'	'nltk_title_neu'	'finbert_text_neu'
'nltk_text_neg'	'nltk_title_comb'	'finbert_text_neu',
'upvote_ratio'		+300 document embedding columns

Table 2: Dataset columns used for training and evaluating a classifier on the data.

features in this dataset, the dataset is split into a training set containing 60% of the data and a test set containing 40% of the data. 5 different classifiers are evaluated:

- A K-nearest neighbors classifier
- A linear support vector machine
- A support vector machine using a radial basis function kernel
- A decision tree classifier
- A dummy classifier that selects the most frequent class label in the training data

The classifiers are trained on each possible combination of stock and sentiment window size (1d, 3d, 7d, 14d), and then evaluated on every prize horizon window (1d, 3d, 7d, 14d).

Results

This chapter presents the results of implementing the method described in the method chapter. Due to the large amount of results for the combining models, only a selection of the results which are deemed most relevant for fulfilling the aim of the project are presented in tables in this chapter. A comprehensive list of results can be found in the GitHub repository[5].

Stock prediction using moving average sentiment scores

A summary of the results of using a moving average sentiment score to predict stock price movements is presented in table 3.

The results show that among the investigated subreddits, the subreddit which moving average sentiment score best predicted stock price movements was r/pennystocks, which

[5] `https://github.com/ErikLundin98/text_mining_reddit`

sentiment model	VADER	FinBERT	VADER	FinBERT
Metric	best accuracy		moving average method	
gme	51.46%	49.75%	upvote weighted	upvote weighted
investing	51.92%	45.19%	regular	upvote weighted
options	52.27%	49.03%	upvote weighted	regular
pennystocks	54.45%	54.80%	upvote weighted	upvote weighted
personalfinance	52.04%	50.10%	upvote weighted	regular
stockmarket	51.54%	54.41%	regular	regular
stocks	51.64%	51.31%	regular	regular
wallstreetbets	48.54%	49.89%	regular	upvote weighted
all	50,44%	49.93	upvote weighted	regular
baseline	52.44%			

Table 3: Summary of the results of using moving sentiment scores from VADER and FinBERT on the Reddit posts which could be linked to listed NASDAQ stocks. The accuracy and moving average method correspond to the combination of sentiment moving average window size and price look-ahead horizon that yielded the highest accuracy for each subreddit. The last row also presents the accuracy of the baseline model

had a 54.43% accuracy when sentiment scores were estimated with VADER and 54.80% accuracy with FinBERT. This accuracy is about 2% higher than the baseline method's accuracy of 52.44%. There is no clear indication that using an upvote weighted moving average of sentiment scores performs better than a regular moving average, as the two methods each contribute to half of the top scores. The best accuracy for each sentiment analysis model and subreddit is also close to 50%($\pm$5%) in all cases.

Table 4 shows that for 6 out of 8 subreddits, sentiment scores using VADER was a better predictor for stock growth. The biggest difference in accuracy happens when using posts from r/investing, where there is a 6.97% difference between the two models. We can also see that r/pennystocks not only has the highest maximum accuracy, but also the highest average accuracy independent of sentiment analysis model.

Stock Prediction using Trained Classifiers

A summary of results from training a classifier to predict stock prices movements on Reddit posts from subreddits r/gme, r/stocks, r/wallstreetbets and the three subreddits combined is presented in table 5.

Table 5 shows that the trained classifiers, which are implemented with scikit-learn[16] manages to beat the baseline model by a significant amount in all cases. The combination of subreddit and classifier that performed best when predicting AMC is r/stocks with a linear SVM. For AMZN, its r/wallstreetbets with a KNN and for GME its r/stocks.

Average accuracy subreddit	VADER	FinBERT	average difference(VADER,FinBERT)
gme	50,62%	49,19%	1,42%
investing	50,48%	43,51%	6,97%
options	51,01%	47,69%	3,33%
pennystocks	53,92%	52,98%	0,94%
personalfinance	50,68%	49,02%	1,65%
stockmarket	50,59%	51,51%	-0,92%
stocks	50,89%	50,68%	0,21%
wallstreetbets	48,26%	49,43%	-1,17%
all	50,19%	49,75%	0,45%

Table 4: The average accuracies and differences in average accuracy between VADER and FinBERT moving averages (average between all combinations of window sizes and time horizons).

Furthermore, we note that in all cases, the best combination of price horizon (label window) and sentiment window size seems to be a combination of 7days and 14days. Label windows or sentiment windows shorter than 7d does not give the highest accuracy for any stock and subreddit. In most cases, a linear SVM performed best in terms of accuracy, followed by a KNN classifier. The KNN classifier looks at the 3 nearest neighbors, and the linear SVM uses a regularization parameter ('C' in scikit-learn) of 0.025.

To compare the result of the classifiers with the result of the moving average sentiment score approach (that does not train a classifier), it is worth adding that the accuracy for individual stocks has also been calculated (as part of calculating the weighted means that are presented in table 3 and 4). The scores for subreddits r/gme, r/wallstreetbets and r/stocks never achieved an accuracy higher than 60%. In other words, the trained classifiers outperform the non-trained approach by a minimum of 29%.

Discussion

The purpose of this project is to examine if Reddit posts can be used to predict if stock prices will go up or down in the future. As a first investigation, it has been evaluated if moving average sentiment scores based on VADER or FinBERT can be used to predict stock price movements. The experimental results seem to indicate that solely using a moving average of sentiments scores from Reddit posts does not work well for predicting future price movements, as most accuracies were close to 50%. If we are only correct in that the stock will go up or down 50% of the times, we will likely experience losses 50% of the times and not make a profit. Furthermore, weighting sentiment scores based on upvotes does not seem to yield any significant improvements in terms of accuracy. An

Trained classifiers

stock	subreddit	best model	acc.	baseline	label window	sentiment window
AMC	GME	linear SVM	90.7%	65.1%	14d	14d
AMC	stocks	linear SVM	91.7%	69.2%	14d	14d
AMC	wallstreetbets	KNN	87.5%	56.8%	14d	7d
AMC	all	KNN	86.25%	52.5%	7d	7d
AMZN	GME	linear SVM	88.0%	49.0%	14d	14d
AMZN	stocks	linear SVM	87.9%	57.6%	14d	14d
AMZN	wallstreetbets	KNN	91.8%	57.5%	14d	7d
AMZN	all	linear SVM	88.1%	54.8%	14d	7d
GME	GME	linear SVM	88.6%	64.4%	14d	14d
GME	stocks	linear SVM	89.7%	73.3%	7d	14d
GME	wallstreetbets	linear SVM	87.1%	62.4%	14d	14d
GME	all	linear SVM	86.2%	62.8%	7d	7d

Table 5: Shows which classifier performs best on Reddit posts from each of the three subreddits r/gme, r/stocks and r/wallstreetbets, each of the three stocks AMC, AMZN and GME. It also shows the best baseline accuracy and which time horizon the labels are based on as well as which moving average window size this classifier was trained on.

interesting observation to make is that there is some differences in performance between different subreddits, as r/pennystocks has an accuracy of 54.5%, and r/wallstreetbets has an accuracy of 49%. This could indicate that the sentiment of certain subreddits are better than others at predicting stock prices. At least, it is the case in the experimental results, but due to the small size of the dataset, it is hard to determine if this is generally true or just a random coincidence - if Reddit posts from 2020 would be used instead, it could be possible that another subreddit than r/pennystocks is the top performer. Two possible ways to improve the method and make it more rigorous would therefore be to use more data, e.g. scrape a dataset with Reddit posts from more years than just 2021. Additionally, one could increase the dataset size simply by changing the evaluation window from a daily basis to an hourly basis. This would be possible since the posts in the dataset are labeled with exact times of when they are created, and we can still use a moving average approach, but on a hourly basis instead of daily. This would increase the dataset size by a factor of 8 (the exchanges are open 8 hours a day). Long et. al. [13] investigated the causality between GME stock prices and r/wallstreetbets' sentiment and claim to see a connection between the two, so hourly evaluation could be possible. At the same time, it introduces many practical complications related to collection of the data and calculating the rolling averages.

Another observation is that, as seen in table 4, there is a positive difference between the accuracies of VADER sentiments and FinBERT sentiments, which could mean that VADER is better suited for this task than FinBERT. Since FinBERT is trained on financial corpuses and VADER is "trained" on social media corpuses, one interpretation of these results is

that the language in the subreddits with a positive difference use a language that is more "social media-like" and that subreddits with a negative difference use a more domain-specific, financial language that FinBERT works better for. This is however a very vague interpretation that would need to be investigated further.

The results from training classifiers on document embeddings, Reddit post features and sentiment scores as features show that a high accuracy can be achieved, especially when using one to two-week moving averages of the features and using binary price growth labels that compare today's price with the price 1-2 weeks from now. The results also show that the best classifier for each subreddit and each of the three stocks was either a linear SVM or a KNN classifier, which consistently beat the naïve baseline method that selected the most occurring class label each time by close to 30%. In this case as well, it has to be pointed out that the dataset on which the classifier was trained on is limited to a small amount of observations (¡365). To combat the small dataset size, classifiers that traditionally perform well on smaller datasets were chosen, but since the test set only consists of less than 150 samples, the generalization properties of the classifiers are hard to evaluate. Since the experiments related to the classifiers were limited to these three stocks, it is difficult to determine how the method would generalize to more stocks and longer time horizons. The very high accuracy is promising however and shows that the classifiers manage to utilize the combination of document embeddings and sentiment scores to perform better than the baseline by a large margin. The use of Reddit posts and sentiment analysis could therefore be incorporated into more complex stock price prediction models and contribute to an increase in accuracy. For instance, Liu et al. [12] show that there are co-movements between company-specific social media metrics and the companies' stock price movements. As future work, it would be interesting to investigate if the information embedded in Reddit posts could be combined with information from other social media sites, news sites and real-time financial data to create a more robust predictive model.

The results in this paper are by no means an indicator of the best possible accuracy when predicting stock prices with Reddit posts. Additional hyperparameter tuning and model variations should be evaluated before using this approach in real time tasks. The "document embedding approach" in this project is very simple as it is takes the average of all word embeddings computed for each Reddit post. A more advanced document embedding model could be used, for instance a transformer, to possibly get better results. Additionally, more sophisticated preprocessing could be done: as mentioned in Limitations & Assumptions, multiple stocks can be assigned to a single Reddit post. In reality, separate sentences in the post are likely to be related to one stock at a time, and assuming that the whole post is equally important to all mentioned stocks is not realistic. Instead a method to divide each post into sections based on which stock they are about could be developed. The first priority when further investigating the approach used in this project should be to solidify the results by increasing the dataset size, investigated models and hyperparameters. As [6] and [12] points out, it is generally believed that companies' stock prices are largely affected

by the general public's perception about the companies. The results in this project are in line with this belief, but are not themselves strong enough to prove it as a fact.

Conclusion

The results in this report indicate that using information from Reddit posts together with sentiment analysis and document vectorization could be a valid approach to achieve strong predictive performance when trying to predict whether stock prices will go up or down in the next 7 or 14 days. The results also seem to indicate that there is no strong connection between moving average sentiment about stocks on Reddit and their future price developments, the best accuracy achieved was at 55%, and if there is a difference in different subreddits' performance in this regard, it is by a small margin. But, classifiers as KNN models and support vector machines can achieve a test accuracy of 87% when trained and evaluated on document embeddings and sentiment scores extracted from Reddit posts created in 2021, which significantly outperforms the used baseline method. Before the conclusion that Reddit posts can be used to predict stock prices, it does however have to be tested whether the method used in this report can be successfully generalized to other stocks than those listed on NASDAQ and to Reddit posts and prices from other years than 2021.

References

[1] Abhinav Anand and Jalaj Pathak. "Wallstreetbets against wall street: The role of reddit in the gamestop short squeeze". In: *IIM Bangalore Research Paper* 644 (2021).

[2] Dogu Araci. "Finbert: Financial sentiment analysis with pre-trained language models". In: *arXiv preprint arXiv:1908.10063* (2019).

[3] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: `1810.04805 [cs.CL]`.

[4] Ronen Feldman. "Techniques and applications for sentiment analysis". In: *Communications of the ACM* 56.4 (2013), pp. 82–89.

[5] Tim Hasso et al. "Who participated in the GameStop frenzy? Evidence from brokerage accounts". In: *Finance Research Letters* (2021), p. 102140.

[6] Wu He et al. "Social media-based forecasting: A case study of tweets and stock prices in the financial services industry". In: *Journal of Organizational and End User Computing (JOEUC)* 28.2 (2016), pp. 74–91.

[7] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[8] Matthew Honnibal and Ines Montani. "spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing". 2017.

[9] Jeremy Howard and Sebastian Ruder. *Universal Language Model Fine-tuning for Text Classification*. 2018. arXiv: 1801.06146 [cs.CL].

[10] Danqi Hu et al. "The rise of reddit: How social media affects retail investors and short-sellers' roles in price discovery". In: *Available at SSRN 3807655* (2021).

[11] Clayton Hutto and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 8. 1. 2014.

[12] Ling Liu et al. "A social-media-based approach to predicting stock comovement". In: *Expert Systems with Applications* 42.8 (2015), pp. 3893–3901.

[13] Cheng Long, Brian M Lucey, and Larisa Yarovaya. ""I Just Like the Stock" versus" Fear and Loathing on Main Street": The Role of Reddit Sentiment in the GameStop Short Squeeze". In: *SSRN Electronic Journal* (2021), p. 31.

[14] Michael Lubitz. "Who drives the market? Sentiment analysis of financial news posted on Reddit and Financial Times". In: *University of Freiburg: http://ad-publications. informatik. uni-freiburg. de/theses/Bachelor_ Michael_Lubitz_2018. pdf* (2017).

[15] NASDAQ. *Nasdaq Listings*. https://datahub.io/core/nasdaq-listings#resource-nasdaq-listed. [Online; accessed 2021-12-23]. 2018.

[16] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[17] Matthew E. Peters et al. "Deep contextualized word representations". In: *CoRR* abs/1802.05365 (2018). arXiv: 1802.05365. URL: http://arxiv.org/abs/1802.05365.

[18] Hu Xu et al. "BERT post-training for review reading comprehension and aspect-based sentiment analysis". In: *arXiv preprint arXiv:1904.02232* (2019).

[19] Yahoo. *GameStop Corp. (GME)*. https://finance.yahoo.com/quote/GME?p=GME&.tsrc=fin-srch. [Online; accessed 2021-12-13]. 2021.