Libera Università Internazionale
degli Studi Sociali Guido Carli

PREMIO TESI D'ECCELLENZA

Reddit sentiment and the US stock market: a Deep Learning based analysis

Alessandra Zannella

2022-2023

LUISS

Libera Università Internazionale
degli Studi Sociali Guido Carli

Working Paper n. 6/2022-2023
Publication date: Dicember 2024
Reddit sentiment and the US stock market: a Deep Learning based analysis

ISBN 979-12-5596-207-6

Luiss Academy is an imprint of
Luiss University Press – Pola Srl
Viale Pola 12, 00198 Roma
Tel. 06 85225485
E-mail lup@luiss.it
www.luissuniversitypress.it

Reddit sentiment and the US stock market: a Deep Learning based analysis

By Alessandra Zannella

ABSTRACT

In this study, we analyze the effects of discussions within the highly-subscribed Reddit community known as *WallStreetBets* (WSB), on US stock market returns and trading volumes. WSB is a platform, boasting nearly 15 million members, where users engage in informal discussion and share content on financial markets. Our investigation primarily revolves around two key factors: stock sentiment and attention. Stock sentiment refers to the tone or attitude, either bullish or bearish, toward a specific stock. We gauge sentiment through a range of three analytical techniques, namely the Valence Aware Dictionary and sEntiment Reasoner (VADER), Support Vector Machine (SVM), and Convolutional/Recurrent Neural Networks (CNN/RNN). Stock attention signifies the level of interaction on WSB regarding a specific stock. In this study, we focus our analysis on a carefully chosen set of stocks: TSLA (Tesla), AMZN (Amazon), GOOGL (Alphabet Inc., Google), NFLX (Netflix), and AMD (Advanced Micro Devices). These stocks have been specifically selected due to their high liquidity, in terms of annual turnover.

The results reveal a statistically significant and positive relationship between attention and volume, suggesting that increased Reddit interactions on WSB often lead to hikes of trading activity. In addition, higher bullishness, in WSB content, tends to coincide with higher trading volumes, though with varying significance levels. However, the impact on stock logarithmic returns varies widely across our set of stocks, suggesting that their characteristics play a pivotal role. Notably, the study focuses on five of the most liquid American stocks. Hence, the likelihood of drastic price fluctuations caused by retail traders is reduced. Only TSLA and AMD are among the most-mentioned stocks, with TSLA potentially classified as a meme stock.

Keywords: Reddit, *WallStreetBets*, meme stocks, sentiment, VADER, Natural Language Processing, Support Vector Machine, Convolutional Neural Network, Long Short Term Memory layer, abnormal volume, returns, Newey-West standard errors.

Introduction

In recent years, social media has become an integral part of people's daily lives. These platforms distinguish themselves from traditional media through their rapid global reach, immediate accessibility, user-friendly interfaces, and constant interactivity. As a result, people are now exposed to a continuous stream of news and, most importantly, opinions. Among all social media platforms, Reddit is one of the primary aggregators of opinions and discussions. Indeed, Reddit pools together many communities, called subreddits, where individuals discuss specific topics of interest. One of the most highly subscribed subreddits is *WallStreetBets*, founded in January 2012, where users discuss financial markets and share investment insights and opinions. Because these have reshaped both the landscape of financial investing and the type of news that retail investors pay attention to, researchers are now concerned with analysing the impact of social network interactions on financial markets. However, investor sentiment on market dynamics has been a longstanding subject of interest, even before the use of social media as a mean to share investment ideas and financial advice. Historically, the focus primarily revolved around professional sources. For instance, the Wall Street Journal, where institutional investors, brokerage firms, and stock analysts provided recommendations and shared their perspectives on future market trends. Research, like Tetlock (2007), explored the impact of Wall Street Journal advice. Greene and Smart (1999) revealed links between recommendations and trading volume shifts. With the rise of the internet, retail investors gained the ability to share their own investment ideas and insights online, rather than solely relying on traditional journals for information. Consequently, numerous research studies began to focus on the impact of online investment advice and microblog conversations on trading activities. For instance, Antweiler and Frank (2004) discovered statistically significant relationships between discussions on platforms like Yahoo! Finance and Raging Bull, and both market volatility and returns. Their findings raised questions about whether the internet's increasing influence and global reach would empower noisy trading to potentially influence financial markets. With a similar approach, Sprenger et al. (2014) delved into the influence of microblogging, specifically tweets, on some market variables. Their analysis encompassed not only factors like traffic and the volume of shared content but also sentiment and disagreement among users. Their findings revealed significant effects on share volatility. Numerous researchers have directed

their efforts toward Twitter analysis, with studies conducted by Gu and Kurov (2020), Sul et al. (2017), Behrendt and Schmidt (2018), Bollen et al. (2011), among others. In the case of Behrendt and Schmidt (2018), they discovered some statistically significant effects of information sourced from stock-related Tweets on intra-day volatility, across all components of the Dow Jones Industrial Average. Reddit analysis has a relatively limited presence in this context. Interest shifted significantly to Reddit, particularly following the so called *Game Stop short squeeze* (Anand and Pathak, 2021). In that occasion, a group of social mobilized retail traders organized themselves on WSB to push up GameStop stock price. This event served as compelling evidence of a new era in financial markets: that of social media platforms. Many studies aimed at comprehending its dynamics and assessing the potential for similar occurrences. For example, Long et al. (2023) described the role of Reddit in the Game Stop short squeeze, analyzing the impact of almost 11 million WSB comments from January 1st to February 28th, 2021, on Game Stop's intra-day returns. Similarly, Betzer and Harries (2022) provided empirical evidence of the co-movement between Reddit posts and various trading measures in the subsequent 30-minute window. Other authors, such as Umar et al. (2021), also incorporated an analysis of social media activity alongside the short ratio. In fact, one of the conditions that made the short squeeze possible was the significant level of short interest by institutional investors, including hedge funds, in GME stock. The Game Stop case was nothing but tip of the iceberg. Indeed, other stocks like AMC Entertainment and Blackberry experienced similar events. Frequently discussed stocks subject to these swings are commonly referred to as "meme stocks" due to their association with social media-driven trading and subsequent market manipulation events.[1] The majority of studies, such as Chacon et al. (2023), have focused on meme stocks and have discovered evidence of Reddit's influence on returns, volatility, and trading volumes. Gianstefani et al. (2022) developed an alert system designed to signal the presence of echo chambers where investors could pool themselves to influence financial markets. They examined the effects on abnormal returns, focusing on specific periods and stocks that triggered their alarm system. The stocks they studied were divided into two categories: meme stocks and non-meme stocks. Unsurprisingly, the alert system was triggered on numerous occasions for meme stocks, while it only activated once or twice for non-meme stocks.

Our research question explores whether the level of interactions and the sentiment conveyed in opinions regarding stocks on social media exert an influence on the stock market. In particular, we examine the volume and the tone of shared content on Reddit. After organizing and extracting valuable

[1] Typically, these stocks are characterized by lower trading volumes and liquidity levels. These particular market inefficiencies create opportunities for market manipulations.

insights from *WallStreetBets* submissions, we assess their impact on five selected stocks. Specifically, we measure these effects on five well-known and liquid technology stocks (TSLA, AMZN, GOOGL, NFLX, AMD). We use Reddit submissions to construct proxies of attention and sentiment on the stocks of interest. We define attention as the volume of Reddit posts concerning a given stock; the tone of these submissions is described by the *sentiment*. Section 1 presents the data we retrieve from WSB and introduces the methodologies employed throughout the analysis. These comprehend three sentiment analysis techniques, as well as the econometric techniques applied to assess WSB influence on the five selected stocks' returns and trading volumes. The first sentiment analysis tool, Valence Aware Dictionary and sEntiment Reasoner, relies on Natural Language Processing (NLP) techniques, which have been previously used by Long et al. (2023) and the majority of the aforementioned researchers. The second tool utilizes a support vector machine, as explored by Kharde and Sonawane (2016). The third tool involves a deep network with a combination of convolutional and LSTM layers, as in Zhao et al. (2021), Minaee et al. (2019), Gandhi et al. (2021), and Usama et al. (2020). Additionally, other techniques, such as keyword searches, have been employed, as highlighted in Chacon et al. (2023). However, these may lead to approximate sentiment labels. We examine how social activity among retail traders influences financial markets. We adopted an augmented version of the Capital Asset Pricing Model (CAPM), a method akin to the approach employed by Broadstock and Zhang (2019) for intraday returns. Additionally, we analyzed the influence of WSB on trading activity using a multiple linear regression model, incorporating control variables as per the methodology outlined by Duz Tan and Tas (2021) and Tetlock et al. (2008). Section 2 reports the econometric analysis results alongside with their interpretation. In line with Duz Tan and Tas (2021), this thesis finds a positive and significant relationship WSB informal discussion and abnormal turnover for most of the selected stocks. Hence, for those stocks higher discussion leads to increased trading activity. Positive net sentiment or bullishness also correlates, with a confidence level higher than 90%, with higher trading volumes for some stocks. In addition, this study incorporates sentiment variables in the context of the Capital Asset Pricing Model for analyzing Reddit contents' effects on logarithmic returns. The outcomes highlighted variability among stocks, confirming no universal rule for Reddit's impact on stock markets (Gianstefani et al., 2022). In addition, most of these results lack of statistical significance. Long et al. (2023) obtained significant relationship only at higher and intraday frequencies. Overall, it is noteworthy to emphasize that the stocks under consideration are highly liquid, reducing the likelihood of significant price fluctuations and manipulations events. Furthermore, only TSLA and AMD emerge as heavily discussed stocks from 2018 to 2023, with TSLA being considered a "meme stock".

Data and Methodology

1. REDDIT DATA

Reddit is a social media organized in various communities, known as subreddits. Relevant interactions to this research happen on WSB where people informally discuss and share opinions on financial markets. WSB was created in 2012 and it became one of the most subscribed Reddit communities boasting an impressive membership of 14.16 million subscribers, as per Reddit *WallStreetBets* Community (2023).

Our WSB data range from January 2018 to March 2023, included. We retrieved them through python Reddit API wrappers, namely, PRAW and PMAW. An API wrapper is a tool that simplifies the process of sending requests to an API (Application Programming Interface). PRAW and PMAW are python libraries that enable the user to interact with Reddit's API. In particular, PMAW uses multithreading, allowing to send multiple requests to the Push Shift API. The Pushshift Reddit API is a particular alternative to the Reddit's official API, over which it is preferred due to its unique advantages. It has the ability to retrieve historical data, while the official API has limitations when accessing older contents. Additionally, the Pushshift API allows users to retrieve large datasets in bulk, resulting in more efficiency. Lastly, it is easier to use since it offers a more flexible way to search for specific content. To download the desired data from WSB we followed The PRAW Quick Start Guide (2023), specifying as subreddit '*wallstreetbets*'. We have obtained a dataset containing Reddit posts, also referred to as submissions, along with associated information. This dataset is structured as a DataFrame, where each row represents a single post, and each column corresponds to a specific attribute or feature. In total, the dataset comprises 2,015,951 submissions. Each submission is characterized by the following features:

`id`: The unique code that identifies the submission.
`title`: The title of a Reddit post is a brief summary or a header of content of the submission.
score: The score of Reddit posts is the difference between the number of up-votes and down- votes. It indicates how well-received has been the submission.
num comments: The number of comments that the submission has received.
selftext: Self-text refers to the main body of the post. It contains the content of the submission.
`flair`: A flair is used to categorize submissions within a subreddit. Each community

has its own 'flairs'. In the case of WSB the possible ones are: YOLO (You Only Live Once), DD (Due Diligence), Gain/Loss, News, Discussion, Options, Shitpost.
author: The user, author of the submission.
date: The timestamp indicating the date when the user posted the submission.

id	title	score	num comments	selftext	flair	author	date
b2p9nc	$TLRY $15 million rev- enue, 1 million net loss, missed EPS by 175% = $7 billion market cap. Calls to the moon!	4	1	[removed]	shitpost	[deleted]	18/03/2019

Table 1. Example of WSB submission

Note: This table provides an example of a submission on WSB. Notably, the content of the submission, identified as `selftext`, and the information pertaining to the author are empty, appearing as 'removed' and 'deleted,' respectively. The reason why `selftext` appears as 'removed' is often attributed to content moderation. In fact, when a submission violates community guidelines, moderators may remove a part of or the entire content. The absence of information in the author field can occur for various reasons. For instance, when a post is removed due to violations of guidelines, the author's details may no longer be visible. Additionally, when Reddit users delete their accounts or specific posts the author information is automatically hidden. Furthermore, members can intentionally choose to leave the author field empty or use an anonymous account.

1.1 Data cleaning and pre-processing

Because both the 'selftext' and 'title' fields contain information that is relevant for this thesis, we have combined them into a new column, named 'text'. Subsequently, we cleaned 'text' column, removing undesired sub-strings[2] and removing all NaN values. We pre-processed 'text', using an ad-hoc built function, pre-proc(). It performs two operations: it filters out stop words; and it removes punctuation and digits. In NLP, stop words are commonly occurring words. They are removed for various reasons. Firstly, they usually contribute a little to the overall meaning of a sentence. Secondly, filtering them out improves computational efficiency. To perform this task, we used a list, containing stop words from two python libraries for textual analysis, namely Natural Language Toolkit (NLTK) and Gensim.

The next pre-processing step is to assess whether a submission contains content on one or more stocks. Hence, following the procedure of Witts et al.

[2] When 'selftext' is [deleted] or [removed] the concatenation of 'selftext' and 'title' yields in sentences with unwanted words.

(2021), we isolate the list of mentioned stocks in a separate column, 'mention'. The purpose is to simplify the process of filtering the DataFrame for specific stocks. To perform this operation, we use python set operations. In particular, we intersect the set of words belonging to each element of the 'text' column with a list containing stock tickers, stock names, bi-gram, tri-grams, uni-grams.[3] The intersection yields in a list that contains different name format,[4] depending on how the user mentioned a specific stock. Hence, we uniform them by converting all of them into stock tickers.

1.2 Summary Statistics

Table 2 shows several statistics computed on daily basis for each of the five years. The use of WSB has increased over time. In particular, both average and total submission increased reaching their peak in 2021 and then slowly decreasing. In 2021, there was a considerable surge in total submissions compared to previous years, as evident from the high value of both total submissions and average daily submissions. This year also exhibits a substantial standard deviation in daily submissions, indicating significant day-to-day fluctuations in social activity. Moreover, the high maximum daily submission value in 2021 suggests a particularly exceptional day (or period) with a large number of submissions. In fact, these days correspond to the second half of January 2021, during the so called "Game Stop Frenzy". When the online community managed to short squeeze Hedge Funds and other institutional investors that were shorting Game Stop. They organized on WSB to push GME price up. On January 22nd the stock closed at 65 dollars. In accordance to Banerji and McCabe (2021) and Pedersen (2021), the peak came on January 28th when the shares reached a price of 483 dollars and Robinhood restricted trading of the stock. The success of the GME short squeeze contributed to increase WSB popularity. In fact, its membership more than doubled in size, from almost 2 million to over 6 million subscribers (Times, 2021). In 2022 and the first quarter of 2023, the submission numbers decreased compared to 2021 but still remained significantly higher than in previous years.

[3] n-grams are sequences of n items or words extracted from a longer text or string.
[4] Tickers, stock names, bi-gram, tri-grams, uni-grams.

Year	Total Submissions	Average Daily Submissions	Standard Deviation Daily Submissions	Maximum Daily Submission
2018	88,646	242.87	121.44	727
2019	70,477	195.23	82.63	527
2020	304,121	830.93	486.56	3,443
2021	1,271,377	3492.79	11,152.79	138,814
2022	240,413	658.67	453.01	4,679
2023 Q1	40,917	454.63	163.00	986

Table 2 Submissions' statistics

Note: This table presents key statistics related to submissions over the years. The data spans from 2018 to the first quarter of 2023, providing insights into the submission patterns over time.

2. SENTIMENT ANALYSIS

The next stage involves associating a sentiment label, either positive (denoted as 1) or negative (de- noted as 0), to each submission. We base our sentiment classification on three methodologies. Two of them are Machine Learning (ML) approaches while the other is a dictionary-based approach. The primary objective is to assess their performances and accuracy. We place significant emphasis, particularly in the case of machine learning models, on mitigating overfitting. Overfitting[5] occurs when a model is excessively tailored to the specific characteristics of the training dataset, resulting in a model that is unable to make accurate predictions on new, unseen data. Conversely, if a model is overly simplistic, you might not be able to capture all the aspects of and variability in the data. In this situation, referred to as underfitting, the model performs poorly even on the training dataset. It is crucial that the chosen model does not overfit the data, as obtaining a reliable and generalizable solution is paramount to the success of our analysis. At the same time, we also need to avoid overly simplistic models. Finding the optimal level of complexity is crucial to capture the key aspects of the data while maintaining generalizability. Before deepening further on the sentiment analysis models we have built and implemented, we introduce briefly the concept of *Fanatic Submission* explaining their relevance to understand redditors' sentiment.

[5] Müller and Guido (2016).

2.1 Fanatic Submissions

WSB, as any other community, has its own "language". Indeed, members usually make use of slangs to garner attention. These expression are crucial to this study, not only because they are some of the most written words, but also because they are crucial to understand the author's sentiment on a stock. Table 12 in Appendix A.1 shows the most typed words.[6] Amongst these stand out typical WSB slangs such as 'moon' and 'yolo'. Table 13, in Appendix A.1, shows several fanatic words and slangs (Long et al., 2023) along their occurrences between 2018 and March 2023. Slangs can consist of a phrase rather than a single word.[7] We define Fanatic Submissions as Reddit posts that contain popular slang terms or phrases. Table 3 reports some summary statistics on their daily counts. They follow a similar pattern of the total submissions, characterized by an upsurge in all metrics until the year 2021, followed by a gradual decrement thereafter. Their total daily count has generally increased over the years. The average daily count of Fanatic Submissions also exhibited an upward trend. The standard deviation of daily Fanatic Submissions reached its maximum value in 2021 at 1,222.73, indicating a considerable variation in the use of slang throughout that year. The maximum daily *Fanatic Submission* hit its peak in 2021 with 12,382 submissions. While there was a decrease in Fanatic Submissions in 2022 and the first quarter of 2023 compared to 2021, the use of investment-related slang remained considerable. Their daily frequency[8] is plotted in Figure 1. As can be observed, the share of fanatic submissions reached a maximum of almost 50% during 2020.

[6] We removed common language words

[7] We used a list of phrases and words as a filter to gather all the "fanatic submissions". Its components are: 'yolo', 'to the moon', 'moon', 'hold the line', 'hodl', 'tendies', 'tendie', 'dd', 'diamond hands', 'diamonds', 'apes together strong', 'apes', 'paper hands', 'paper', 'stoncks', 'loss porn', 'gain porn', 'bagholder'.

[8] Computed considering the total number of submission per day.

Year	Total Fanatic Submissions (% of Total)	Average Daily Fanatic Submissions	Standard Deviation Daily Fanatic Submissions	Maximum Daily Fanatic Submission
2018	11,675 (13.17%)	31.99	16.73	92
2019	9,292 (13.18%)	25.74	12.73	88
2020	43,317 (14.24%)	118.35	68.95	499
2021	179,296 (14.10%)	492.57	1,222.73	12,382
2022	24,878 (10.34%)	68.16	71.13	781
2023 Q1	3,715 (9.08%)	41.28	18.96	103

Table 3. Fanatic Submissions' statistics

Note: The table reports statistics on data that spans from 2018 to the first quarter of 2023.

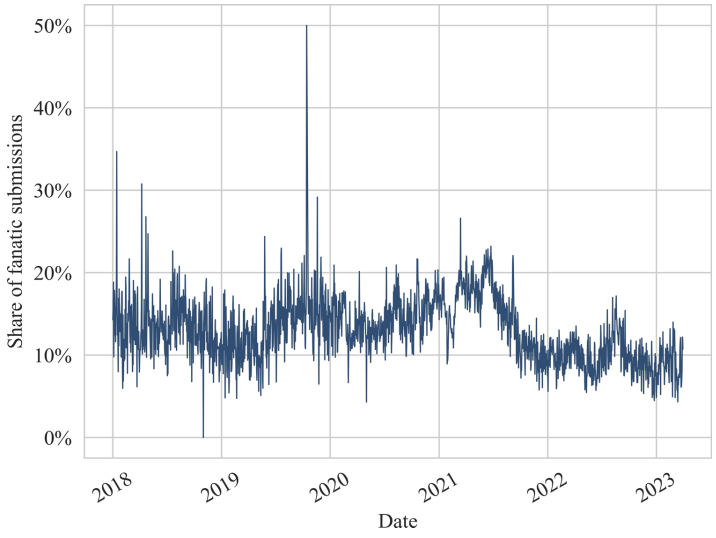

Figure 1: Fanatic Phrases and Words Frequency

Note: The figure illustrates the trajectory of Fanatic expressions' frequency from 2018 to the first quarter of 2023. The frequency is calculated as the percentage of Fanatic submissions on a daily basis relative to the total number of daily submissions.

2.2 Vader Intensity Analyzer

The VADER stands for Valence Aware Dictionary and sEntiment Reasoner and it is a lexicon and rule-based tool particularly useful when analysing short informal texts. For this reason, as soon as Hutto and Gilbert (2014) presented it, it became very popular for social conversation's sentiment analysis. As Witts et al. (2021) highlight, VADER offers a significant advantage over ML models since it does not require training. Hence, it eliminates the need for a training set and potential biases that may arise during data collection. Moreover, it considers both individual word meanings and their context. Nevertheless, its effectiveness heavily relies on the quality and coverage of its dictionary and pre-built sentiment lexicon. In scenarios involving domain-specific terms, VADER may lead to potentially inadequate results. Furthermore, numerous researchers, such as Long et al. (2023), have pointed out that when dealing with Reddit posts, incorporating subreddit-specific slangs and idioms into the VADER dictionary improves its accuracy. In fact, it is possible to achieve a substantial improvement in sentiment analysis performance. This is why we augment VADER's lexicon with the terms reported in Appendix A.1.

VADER provides its outcomes in the form of a dictionary (key-value pairs). The keys include 'positive', 'negative', and 'neutral', representing the weights associated with positive, negative, and neutral words, respectively; 'compound' corresponding to an overall sentiment score. For the purposes of this study, the submissions are categorized as follows: those with a compound score less than 0 are assigned the label 0, indicating a negative sentiment, while submissions with a compound score higher than zero are labeled 1, indicating a positive sentiment. Posts with a neutral sentiment are appropriately labeled as such. We quantify Vader's accuracy using a test data set,[9] with manually annotated sentiment polarity labels.

We evaluate the accuracy of out sentiment analysis models making use of *Confusion Matrices*. Confusion matrices are among the most comprehensive means of presenting the results of binary classification models. They are 2x2 arrays[10] where the rows correspond to the true classes and the columns correspond to the predicted classes. The confusion matrix is shown in Figure 3 in Appendix A.1. The top left quadrant represents cases where the actual class is negative, and the model correctly predicted it. The bottom right quadrant represents cases where the actual class is positive, and the model correctly predicted it. The bottom left quadrant and the top right one are the

[9] It consists of manually labelled Reddit post and tweets on stocks and financial instruments. The length of the data set is 2,415.

[10] In general, if they are square matrix and the dimensions are equal to the number of classes.

complementary set of False Positive and False Negative, respectively. In Python, we set norm='pred' so that each quadrant provides the normalized category, relative to the predictions made by the model.

2.3 Support Vector Machine

A Support Vector Machine is a supervised machine learning algorithm, developed in AT&T Bell Laboratories by Boser et al. (1992). It is used mainly for classification purposes, but it can be used also for regression tasks. In this research, we utilize it for binary classification. SVM operates following a geometric approach. Indeed, it searches for the hyperplane that best separates two or more classes of data points. On each of the axes of the vector space, a feature of the data points is measured. Since SVM are supervised ML algorithms, they require a training data set. In addition, it is necessary to evaluate the model's out of sample behaviour, using a validation and test set. These data sets consists of manually labelled[11] WSB submissions and tweets.

Prior to training SVM, we go through some preliminar steps. Firstly, we re-balance the labelled dataset[12] and split it[13] into training and validation set. Secondly, we pre-processed them using the ad-hoc built function preproc(). Then, we apply tokenization,[14] breaking down the text into a set of unique words (tokens) and constructing a dictionary associating to each of them its frequency. The output is a matrix where each row stands for a submission, each of the columns represents a token and the values are the frequency of the specified token or word in the submission text. Subsequently, we implement term frequency-inverse document frequency (TF-IDF) vectorization. [15] It converts text data into a numerical representation that reflects the importance of the words in a submission. The output is a matrix with the same shape, columns and rows of the CountVectorizer output. The values of the TF-IDF are computed for each token as the product between Term Frequency[16] and Inverse Document Frequency, which is the logarithm of the total number of submissions divided by the number of submissions containing the term.

[11] They are marked as either positive or negative.
[12] The dataset comprises 14,037 sentiment-labelled submissions, with 6,037 classified as negative and 8,900 classified as positive. We rebalanced the dataset, resulting in an equal number of positive and negative submissions.
[13] The training set comprises 80% of the original dataset, with the remaining part, 20%, being the validation set.
[14] For tokenization we used CountVectorizer(max features=5,000) from the sklearn.feature extraction.text module. Where 5,000 is the maximum number of words for which we count the frequency.
[15] The class we used in python is TfidfTransformer() from sklearn.feature extraction.text.
[16] The one obtained with CountVectorizer.

After completing the aforementioned stages, we start training our Support Vector Machine. Intuitively, when performing binary classification, the SVM's goal is to find the best possible dividing hyperplane that effectively separates a set of elements into two classes. For the sake of comprehensibility, the SVM optimization problem will be firstly presented in the linear case. The training set, T, is a set of tuples, each comprehending a tokenized and TF-IDF vectorized submission, sub_i, with its associated sentiment label, y_i. The number of tuples of the training set.[17]

$$\text{is } m \text{ an it is equal to 9,660 and } i = 1,2...m.$$
$$T = \{(sub_1, y_1), (sub_2, y_2), ..., (sub_m, y_m)\} \text{ where } y_i \in \{-1, +1\}.$$

In our study, y_i identifies the class of the i^{th} submission and it is either -1, when the submission expresses a bearish view, or $+1$, when the content is bullish. The dimension of sub_i is dx1 for any i where $d = 5,000$. In fact, the number of features we have set, in the vectorization stages, are 5,000. In general, a separating hyperplane in the sample space can be expressed as a linear function:

$$\mathbf{w}^T \mathbf{sub} + b. \qquad (1)$$

Where $w = (w_1, w_2, ..., w_d)$ is the normal vector[18] (1xd, with $d = 5,000$), controlling for the direction of the hyperplane; $sub = (sub^{'}, sub^{'}, ..., sub^{'})$ the submissions matrix (mxd, where $m = 9,660$); and b is the bias, the distance between the hyperplane and the origin. When it correctly classifies the data points:

$$\mathbf{w}^T \mathbf{sub} + b > 0 \Leftrightarrow y_i = +1;$$

$$\mathbf{w}^T \mathbf{sub} + b > 0 \Leftrightarrow y_i = -1. \qquad (2)$$

[17] We have previously split the labeled dataset, resulting in 80% of 12,074 elements for the training set.

[18] In geometry, the normal vector is a vector orthogonal to the hyperplane. A hyperplane can be defined with its normal vector and the distance from the origin.

Then, it must hold that:

$$\mathbf{w}^{\mathrm{T}}\mathbf{sub}+\mathrm{b}\geq +1\Leftrightarrow y_i=+1;$$

$$\mathbf{w}^{\mathrm{T}}\mathbf{sub}+\mathrm{b}\geq \text{-}1\Leftrightarrow y_i=\text{-}1. \qquad (3)$$

Equation 3 can be re-written as:

$$y_i(\mathbf{w}^{\mathrm{T}}\,\mathbf{sub}+b)\geq +1. \qquad (4)$$

The sample points closest to the hyperplane are called support vectors. The margin is the total distance from two (belonging to different classes) of them to the hyperplane:

$$Margin=\frac{2}{|\mathbf{w}|}. \qquad (5)$$

The goal of the SVM is to find the optimal separating hyperplane, that respect the constraint in Equation 4 with the maximum margin. From a mathematical standpoint:[19]

$$\begin{aligned}&\max_{\mathbf{w},b}\ \frac{2}{|\mathbf{w}|};\\ s.t.\quad &y_i(\mathbf{w}^T\,\mathbf{sub}+b)\geq +1.\end{aligned} \qquad (6)$$

Usually, this is solved by minimizing $\frac{1}{2}|w|^2$ subject to the same constraints and using the Langrange Theorem. All the aforementioned steps have been conducted under the assumption of linear separability in training sample data.

[19] Zhou (2021)

However, when we apply the GridSearchCV class[20] from sklearn.model selection to our Reddit labelled dataset, it reports that the optimal model is the non-linear variant, utilizing the Radial Basis Function (RBF) kernel. When data points aren't linearly separable, they are mapped from the original feature space to an higher dimensional one. Nevertheless, this can be computationally inefficient due to the very high number of dimensions of the mapped feature space and to the inner product in the Lagrange equation. The hyperplane equation is:

$$\mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}(\mathbf{sub})+\mathrm{b}. \qquad (7)$$

The mapped model's Lagrange function is:

$$\frac{1}{2}(|\mathbf{w}|)^2 + \sum_{i=1}^{m} \lambda_i(1 - y_i(\mathbf{w}^T \boldsymbol{\varphi}(\mathbf{sub}_i) + b)). \qquad (8)$$

Where $\boldsymbol{\varphi}(\mathbf{sub}_i)$ is the i^{th} column of the submission matrix mapped in the higher dimensional feature space. And λ_i is the Lagrange multiplier corresponding to the i^{th} constraint. Setting the partial derivatives of the Lagrange function with respect to w and b to zero, yields:

$$\mathbf{w} = \sum_{j=1}^{m} \lambda_i y_i \boldsymbol{\varphi}(\mathbf{sub}_i); \qquad (9)$$

$$\sum_{j=1}^{m} \lambda_i y_i = 0. \qquad (10)$$

[20] This is a class that allows to find the best hyperparameters and Kernel function given a training set.

Substituting 9 in the Lagrange function and considering 10 as our constraint:[21]

$$\max_{\lambda} \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j y_i y_j \quad \varphi(\mathbf{sub}_i), \varphi(\mathbf{sub}_j) \quad ;$$

$$s.t. \sum_{i=1}^{m} \lambda_i y_i = 0; \tag{11}$$

$$\lambda_i \geq 0, i = 1, 2, \ldots, m.$$

Using an RBF Kernel implies that the inner product $\boldsymbol{\varphi}(\mathbf{sub_i}),\boldsymbol{\varphi}(\mathbf{sub_j})$ is expressed as:

$K(\mathbf{sub_i},\mathbf{sub_j}) = \varphi(\mathbf{sub_i}),\boldsymbol{\varphi}(\mathbf{sub_j})$ where $K(\mathbf{sub_i},\mathbf{sub_j})$ is the RBF Kernel function:

$$K(\mathbf{sub}_i, \mathbf{sub}_j) = exp\left(\frac{|\mathbf{sub}_i - \mathbf{sub}_j|^2}{2\gamma^2} \right). \tag{12}$$

Where $\boldsymbol{\gamma}$ is the width of the RBF Kernel. In the model described it is set to 1.[22] Then, the equation of the hyperplane is:

$$f(\mathbf{sub}) = \langle \mathbf{w}, \varphi(\mathbf{sub}) \rangle + b = \sum_{i=1}^{m} \lambda_i y_i K(\mathbf{sub}_i, \mathbf{sub}_j) + b. \tag{13}$$

Another hyper parameter is C, it controls the trade-off between maximizing the margin between classes and minimizing the classification error. Therefore, it is referred as regularization parameter. It is introduced in the model by substituting the so called hard margin with the soft margin, that allows for violation of the constraints. The modified objective function in the constrained optimization problem becomes:

$$\min_{\mathbf{w},b} \frac{1}{2} |\mathbf{w}^2 \quad C \sum_{i=1}^{m} Loss(\mathbf{w} \; b \quad | \; + \quad , \;). \tag{14}$$

[21] For all the mathematical steps, please see Zhou (2021).
[22] We select the hyperparameter considering the GridSearchCV outputs.

In this way the loss[23] is minimized while maximizing the margin. High values of C can lead to a tighter fit to the training data, potentially implying to overfitting. Conversely, lower values of "C" may result in a more lenient fit and misclassifications, as the model sacrifices predictive accuracy. In this case C is set to 1.

We build the confusion matrix of the SVM with the same test set used for evaluating VADER's accuracy. It is in Figure 4 in Appendix A.1. Another metrics to evaluate the performance of SVM is the f1score:

$$f1score = 2\frac{Precision \times Recall}{Precision + Recall}. \tag{15}$$

Where precision is the ratio of true positives to the sum of true positives and false positives. While recall is the ratio of true positives to the sum of true positives and false negatives. The svm has a f1score equal to 80% and the classification report is available in table 14 in Appendix A.1.

[23] Python class SVC utilizes the Hinge Loss by default: $max(0, 1 - yif$ (**subi**)

2.4 Convolutional Neural Network and LSTM

Building on the methodology proposed by Zhao et al. (2021) and the recent work of Ma- hadevaswamy and Swathi (2023), we employ a deep learning model for categorizing the sentiment of the WSB submissions. As previously done for SVM, before applying the Neural Network we perform rebalancing, pre-processing, vectorization and tokenization on the data set. We split the labelled dataset into training set, comprising 80% of the data, and validation set, which comprises the remaining 20%.

Neural Networks have been widely used text analysis and classification tools because, contrar- ily to traditional text classifiers, they do not rely on human-designed features such as dictionaries. Furthermore, they can be designed to learn contextual information rather than solely considering words and their semantics. They were described by Kohonen (1988) as "massively parallel interconnected networks of simple (usually adaptive) elements and their hierarchical organizations which are intended to interact with the objects of the real world in the same way as biological nervous systems do". Their basic constituents are the neurons, in this definition "simple elements". They receive input signals, subsequently process them through an activation function and then produce outputs. Their activation dynamics depend on their input weights and thresholds, parameters that are learned directly from the data during the training phase. A neural network consists of a connection of neurons organized in layers. The training process is carried out with learning algorithms. Among them, backpropagation is widely used. Intuitively, it iteratively searches for the weights and thresholds that minimize the learning Mean Squared Error (MSE). Supposing a training sample where inputs are denoted by x and corresponding output by y. $\hat{y}$ is the vector of estimated outputs. The MSE is:

$$\frac{1}{l}\sum_{i=1}^{l}(\mathbf{y}-\hat{\mathbf{y}})^2$$

Here, l is the number of output layers. This algorithm tunes the parameter following the gradient descent method.[24]

[24] To deepen further the gradient descent method please refer to Zhou (2021)

The architecture used for this task comprehends a CNN joined with a bidirectional LSTM. Figure 2 from Alayba et al. (2018) displays the central part of our architecture. A convolutional network is a particular deep learning algorithm that, by combining convolutional and pooling layers, manages to capture the most important features of an input reducing its dimensionality. It is particularly well suited for images pattern recognition and text processing and classification. Long Short Term Memory layers are advantageous when dealing with sequential data, such as text. In fact, their ability lies in how they manage Long-Term dependencies learning, carrying forward only useful information while disregarding the superfluous one. We analyze the Convolutional layer and the Long Short Term Memory layer in details in Appendix A.1.

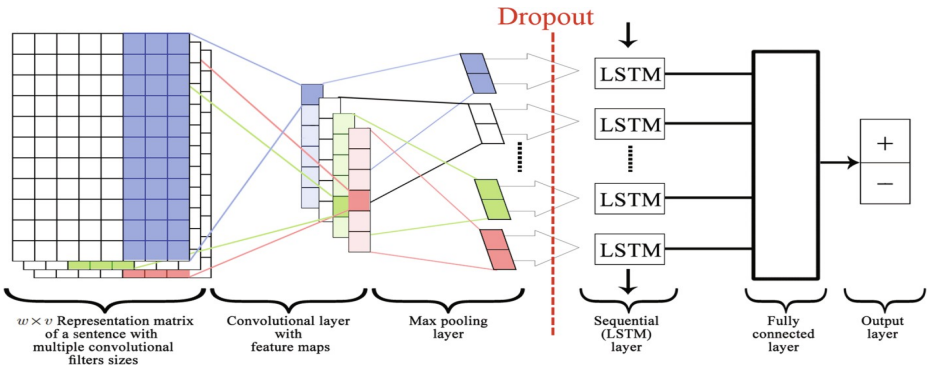

Figure 2: An example of a CNN-LSTM architecture

Note: The figure displays the core part of our deep learning network. The input enters the model and passes through a convolutional layer where a feature map, consisting of multiple neurons, serve as filters to perform convolutional operations. Subsequently, a max pooling layer reduces the dimensionality of our data by selecting the maximum value from a specified window of elements. The dropout is a regularization technique that prevents overfitting by randomly deactivating a user-specified proportion of neurons. The LSTM comprehends four main layers and manages to carry forward only relevant information. Our output layer consists of a fully connected layer with sigmoid activation functions. Their output range from zero to one.

Source: Alayba et al. (2018)

In Appendix A.1 Table 15 summarizes the model and reports the output size for each layer. Furthermore, A.1 contains the detailed sequence of the NN's components alongside with a brief description of each layer.

Some of the hyperparameters[25] in the model are chosen according to a *Random Search*[26] algorithm that maximizes validation accuracy. We trained our neural network for 100 epochs and we inserted the *early stopping* feature to mitigate overfitting. While the training set serves to calculate the gradient and to update the weights, the validation set is used to assess the error. The early stopping feature interrupts the training process when the training error decreases but the validation error starts to increase. The confusion matrix of the CNN-LSTM is displayed in figure 5 in Appendix A.1. It is computed by fitting the model on the test set.[27]

3. STOCKS DATA

We consider stocks traded either on the New York Stock Exchange or on the NASDAQ. Because the majority of WSB users are from the United States, a significant portion of contents pertains to American stocks. We collect tickers and extended names of each stock using the Nasdaq Screener tool. In doing it, we remove type C shares, rights, ETFs and warrants. In addition, we create a stock dictionary, using the stock tickers as keys and providing various ways to refer to the instruments themselves. These includes long names, short names, tickers with the dollar sign at the beginning, uni-grams, bi-grams, and tri-grams.[28] The dictionary proved to be particularly useful when associating each submission with one or more relevant stocks. Table 4 displays the three most discussed stocks per year. We sort the mentions, based on the frequency of submissions containing specific tickers. An additional column indicates the total number of comments received by each of these submissions. Tesla (TSLA) appears as one of the most mentioned stocks in each of the years listed, with a consistent presence from 2018 to 2023. This indicates the enduring attention that Tesla has received. Moreover, GameStop (GME) experienced a significant surge in mentions in 2021, with 4,692,780 comments, likely due to the Reddit-driven short squeeze phenomenon that year. Similarly, AMC Entertainment (AMCX) had a notable peak in mentions in 2021, likely associated with the meme stock mania that year.

[25] Namely, l2 regularization strength, dropout rate and the number of filters of the convolutional layer.
[26] The Random Search is an algorithm from keras library whose task is hyperparameters tuning.
[27] The same used on SVM and VADER.
[28] Names that may include, for example, “Inc.”, “Corp”, “Investment”, “Group”, “Corporation”.

Year	Ticker	Number of Comments	Number of Submissions
2023	TSLA	39,907	915
	BBBY	18,469	854
	SYBT	7,193	475
2022	BBBY	344,796	8,040
	GME	144,380	3,336
	TSLA	146,091	2,886
2021	GME	4,692,780	103,584
	AMCX	432,925	64,596
	RH	61,839	9,469
2020	TSLA	201,634	10,593
	GME	65,445	2,451
	SYBT	100,901	2,062
2019	TSLA	51,418	1,827
	CHN	20,742	691
	AMD	21,984	623
2018	MU	61,700	3,304
	TSLA	41,513	2,367
	AMD	26,873	1,652

Table 4. Most Mentioned Stocks per year

Note: The table provides the ticker of the most mentioned stocks per year, along with number of submissions mentioning them and the number of comments they have received.

3.1 Stocks statistics

For the analysis conducted in this research, we select five stocks from the twenty most liquid ones traded between 2018 and March 2023. The criterion we use for liquidity assessment is the total turnover.[29]

[29] We downloaded stocks historical data using both *yfinance* Python library and the tickers in the stocks dictionary.

4. ECONOMETRIC ANALYSIS

4.1 Attention and Sentiment Time Series

For each of the five stocks, we generate multiple time series. The attention time series aims to measure how much the stock is being discussed within a three days period.

$$Att_t = \sum_{i=0}^{2} Mentions_{t-i}. \tag{16}$$

Here, $Mentions_t$ is the count of daily submissions mentioning the stock at day t. We formulate the positive and negative sentiment time series[30] using the same approach. These series are generated by calculating a rolling sum over a 3-day window, capturing the count of positive and negative submissions separately.

$$Positive_t = \sum_{i=0}^{2} PositiveMentions_{t-i}; \tag{17}$$

$$Negative_t = \sum_{i=0}^{2} NegativeMentions_{t-i}. \tag{18}$$

Where $PositiveMentions_t$ and $NegativeMentions_t$ are the counts of how many times the stock appeared within a bullish and bearish, respectively, submission at day t. Additionally, we construct a time series consisting of the difference between positive and negative submissions. It can be interpreted as the level of bullishness on the specific mentioned stock.

$$Sent_t = Positive_t - Negative_t. \tag{19}$$

In Appendix A.2, Figure 6, 7, 8, 9 display the aforementioned time series.

4.2 WSB Sentiment and Trading Activity

To evaluate the impact of WSB content on trading activity we adopt a methodology similar to that employed by Tetlock et al. (2008) and Sprenger et al. (2014). The dependent variable is the daily abnormal volume, defined as follows:

[30] The chosen sentiment analyzer tool is SVM.

$$AV_t = \ln(V_t) - \frac{1}{3}\sum_{i=0}^{2} \ln(V_{t-i}). \tag{20}$$

Where V_t is daily volume at time t. It represents a proxy from abnormal trading activity since it quantifies how far the daily logarithmic turnover deviates from its three-day rolling window average. This definition was previously used by Duz Tan and Tas (2021). The control variables are size, Parkinson volatility, cumulative return at time t − 1. They are denoted as S_{t-1}, PV_{t-1}, CR_{t-1} respectively. Parkinson volatility is defined as in Parkinson (1980).

$$S_{t-1} = ln(MarketCap_{t-1}); \tag{21}$$

$$PV_{t-1} = \sqrt{\frac{1}{4 \cdot 3ln2}\sum_{i=1}^{3} ln\left(\frac{H_{t-i}}{L_{t-i}}\right)}; \tag{22}$$

$$CR_{t-1} = \sum_{i=1}^{3} LogRet_{t-i}. \tag{23}$$

Where H_{t-i} represents the highest price observed on day $t-i$, and L_{t-i} represents the lowest price observed on day $t-i$.
The explanatory variables are Att_{t-1} as defined in and $Sent_{t-1}$, attention and sentiment, respec- tively, at day t − 1.

$$Att_{t-1} = \sum_{i=0}^{2} Mentions_{t-1-i}; \tag{24}$$

$$Sent_{t-1} = Positive_{t-1} - Negative_{t-1}. \tag{25}$$

Where $PositiveMentions_t$ and $NegativeMentions_t$ are the counts of how many times the stock appeared within a bullish and bearish, respectively, submission at day t. The following equations describe the relationship between abnormal volume, as defined in equation 20, attention, and sentiment, respectively:

$$Positive_{t-1} = \sum_{i=0}^{2} PositiveMentions_{t-1-i}; \tag{26}$$

$$Negative_{t-1} = \sum_{i=0}^{2} NegativeMentions_{t-1-i}. \tag{27}$$

Here $X_{t-1} = [S^*_{t-1}, PV^*_{t-1}, CR^*_{t-1}]$ is a $nx3$ matrix and γ is a 1x3 vector of parameters. The symbol asterisk denotes the standardization applied to all these time series.

4.3 WSB Sentiment and Stock Return

This part of the study aims to illustrate whether positive and negative sentiment on WSB have impact on the chosen stocks' daily logarithmic returns. Similarly to what Broadstock and Zhang (2019) have done, we extend the framework of the Capital Asset Pricing Model. Contrarily to this previous research that dealt with intra-day data, we consider Reddit and stocks data at daily frequency. In this analysis, the logarithmic stock return serves as the dependent variable. As in CAPM, the logarithmic return of the S&P 500 is used as a proxy for the market return. We introduced sentiment-related time series as supplementary explanatory variables. The following equations describe them:

$$Pos_{t-1} = ln(Positive_{t-1} + 1); \quad (30)$$

$$Neg_{t-1} = ln(Negative_{t-1} + 1). \quad (31)$$

Similarly to Padalkar (2021), we apply the logarithmic transformation to the sentiment related time series. We estimate the coefficients of our sentiment-augmented CAPM with an ordinary least square (OLS) model. Because we are dealing with time series data, we compute the Newey-West, heteroskedasticity and autocorrelation corrected, standard errors. The model is specified by the following equation:

$$Ret_t = \alpha + \beta_{market} Mkt_t + \beta_{pos} Pos_{t-1} + \beta_{neg} Neg_{t-1}. \quad (32)$$

Ret_t denotes the logarithmic return of the stock for the period from day $t-1$ to t, while Mkt_t represents the logarithmic return of the S&P 500 index during the same time interval, from t − 1 to t.

We expect to find out that higher Pos_{t-1}, which serves as a proxy for bullish interactions, will be reflected in an higher logarithmic return the on following day, all else remaining the same. Conversely, we imagine that as Neg_{t-1}, that represents WSB bearishness, increases, the stock's logarithmic return decreases.

2. Empirical Results

1. ABNORMAL VOLUME

We implemented the econometric analysis described in Equation 28 and Equation 29 on MATLAB. The coefficients are estimated adopting Ordinary Least Squares (OLS) method. For this purpose, we make use of the hac function which stands for Heteroskedasticity and Autocorrelation Consistent. It is intended to adjust the biases in the OLS estimation of the covariance matrix, arising when dealing with serial correlated and heteroskedastic time series data. It is built around Newey-West (Newey and West, 1986) definition of covariance matrix and standard errors (SE). Appendix A.2 shows how we obtain them. In this case, OLS coefficients remain unchanged while their SE, and hence the significance tests associated with them, become more reliable.

1.2 Abnormal Volume and WSB Stock Attention

Table 5 provides information on the relationship between abnormal volume and stock attention for the five selected stocks. $\hat{\beta}_{att}$ is the estimated coefficient of the explanatory variable Att^{*}_{t-1} for each stock. Its value indicates the impact of a one-unit change in stock attention on the abnormal volume, keeping other factors constant. This coefficient is positive for almost all the stocks, except from GOOGL. Hence, overall, it suggests a positive relationship between WSB activity and abnormal traded volumes. Standard Errors provide a measure of the variability in the coefficient estimates.[31] We conduce a two tailed test to assess whether the relationship between abnormal volume at day t, AV^{*}, and WSB attention at $t-1$, Att^{*}_{t-1}, is statistically significant. The test statistic is:

$$TestStatistic = \frac{\hat{\beta}_{t}^{at}}{\overline{SE_{t}^{at}}}, \tag{33}$$

[31] The smaller SE values are, the greater is the precision in the estimates.

and $TestStatistic \sim t(n-k)$ where n is the number of observation[32] and k is the number of explanatory variables.[33]

The value of $\hat{\beta}_{att}$, 0.058, for TSLA indicates that a unit increase in WSB stock attention corresponds to a 0.058 increase in abnormal volume on the subsequent day. The coefficient is statistically significant, having a p-value smaller than 10%. Analogously, an increase in AMZN mentions within WSB submissions has a positive impact on abnormal traded volumes. Indeed, $\hat{\beta}_{att}$ is equal to 0.135 and it has a statistical significance at 5.7%. Also NFLX trading volumes appear to be influenced by WSB interaction. In this case, $\hat{\beta}_{att}$ is 0.090 and its p-value is 6.3%. Similarly the $\hat{\beta}_{att}$ for AMD is positive. However, it isn't statistically different from zero with 90% confidence, as the p-value is 15%. The model when applied to GOOGL's WSB attention data reveals a negative impact on abnormal turnover. However, it is not found to be statistically significant.

The results are overall in line with what Duz Tan and Tas (2021) highlighted with their analysis on Twitter stock discussion volume. However, contrary to Witts et al. (2021) that discovered a negative lagged relationship between WSB discussion and stock trading volume, we demonstrated a positive one. This findings can be motivated using Sprenger et al. (2014) argument that social media platforms and micro blogs that host financial discussion are used by some investors to confirm their investment decisions. These investors, defined by Cao et al. (2001), wait for signals that confirms their investment opinions to actually trade on them. This theory could apply to our case since we observe a positive lagged relationship between WSB and trading activity.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{att}$	0.058*	0.135*	-0.006	0.090*	0.049
SE	0.034	0.071	0.015	0.048	0.034
Test statistics	1.697	1.899	-0.384	1.862	1.442
p-value	0.090	0.057	0.701	0.063	0.150

Table 5. Abnormal Volume and Attention

Note: This table presents results from regression in Equation 28. The dependent variable

[32] It is equal to 1.314.
[33] It is equal to 4.

is the abnormal volume, measured as the logarithm of total volume on day t minus the average of logarithmic volume from day t until t-2. The explanatory variable of interest is stock attention, computed as the rolling sum over three days of the number of WSB submissions mentioning that stock. * indicate statistical relevance at the 10% level.

1.2 Abnormal Volume and WSB Stock Sentiment

Tables 6, 7, 8 provide insights into the relationship between abnormal volume at day t, AV_{t^*}, and stock sentiment at day $t-1$, $Sent^*_{t-1}$, using the three different sentiment analysis techniques, namely VADER, SVM, CNN-LSTM architecture.

The model described in (29) when Valence Aware Dictionary and sEntiment Reasoner is implemented, unveils that bullishness on WSB and trading activity are positively related. However, none of the $\hat{\beta}_{sent}$ shows statistical significance when considering a 90% confidence interval. Only the regressions for TSLA and AMZN lead to coefficients with associated p-values lower than 20%. For instance, the p-value of WSB TSLA attention is equal to 12.6%.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{sent}$	0.037	0.073	0.033	0.050	0.038
SE	0.024	0.055	0.034	0.043	0.034
Test statistics	1.531	1.303	0.986	1.185	1.124
p-value	0.126	0.193	0.324	0.236	0.261

Table 6. *Abnormal Volume and VADER Sentiment*

Note: This table presents results from regression in Equation 29. The dependent variable is the abnormal volume, measured as the log of total volume on day t minus the average of log volume from day t until t-2. The explanatory variable of interest is stock sentiment, computed as the rolling sum over three days of the difference between positive and negative WSB submissions mentioning that stock. The sentiment time series is obtained with VADER sentiment analyzer. None of the coefficients are statistically significant.

When considering SVM-based sentiment time series, results vary across the five regressions. For instance, TSLA has a positive and statistically different from zero, at almost 93% confidence interval, $\hat{\beta}_{sent}$. In particular, it is estimated that to a one point increase in TSLA's bullishness, $Sent^*_{t-1}$, it corresponds an increase by 0.048 in the level of abnormal trading activity. AMD's results resemble those of TSLA, with a $\hat{\beta}_{sent}$ of 0.054 significant

with almost 90% confidence. While, AMZN and GOOGL WSB bullishness do not have have a significant impact on abnormal turnover, having p-values equal to 0.799 and 0.757, respectively. Finally, the model when implemented with NFLX time series reveals a negative relationship with a coefficient equal to -0.042 and associated p-value of 20%. This suggests that trading activity for NFLX is stimulated more by negative submissions mentioning the stock than positive ones. These findings are consistent with the theory developed by Gianstefani et al. (2022). They distinguished the analysis applied to "meme stocks", the ones triggering their alert system, from the one applied to non-meme stocks. Our HAC model reports statistically significant results only for TSLA and AMD, the ones that can be considered to some extent, as "meme stocks". Indeed, they are the ones falling into the category of most-mentioned stocks (Table 4).

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{sent}$	0.048*	0.008	0.006	-0.042	0.054
SE	0.027	0.030	0.020	0.033	0.033
Test statistics	1.805	0.254	0.308	-1.277	1.610
p-value	0.071	0.799	0.757	0.202	0.108

Table 7. Abnormal Volume and SVM Sentiment
Note: This table presents results from regression in Equation 29. The dependent variable is the abnormal volume, measured as the log of total volume on day t minus the average of log volume from day t until t-2. The explanatory variable of interest is stock sentiment, computed as the rolling sum over three days of the difference between positive and negative WSB submissions mentioning that stock. The sentiment time series is obtained with the support vector machine. * indicate statistical relevance at the 10% level.

The $\hat{\beta}_{sent}$ coefficient for each of the five selected stocks are substantially different when using a CNN-LSTM architecture for sentiment analysis. WSB bullishness on AMZN negatively influence trading activity with a $\hat{\beta}_{sent}$ equal to -0.098 and p-value of 1.9%. Likewise, NFLX Sent_{t-1} coefficient is negative with a 3.2% p-value. In addition, both GOOGL and AMD have a $\hat{\beta}_{sent}$ which is positive but not statistically significant. However, even when using CNN-LSTM, TSLA WSB bullishness and abnormal trading activity have a positive significant relationship with $\hat{\beta}_{sent}$ equal to 0.048.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{sent}$	0.040	-0.098**	0.014	-0.104**	-0.006
SE	0.028	0.042	0.025	0.049	0.029
Test statistics	1.459	-2.337	0.543	-2.148	-0.216
p-value	0.145	0.019	0.587	0.032	0.829

Table 8. Abnormal Volume and CNN-LSTM Sentiment

Note: This table presents results from regression in Equation 29. The dependent variable is the abnormal volume, measured as the log of total volume on day t minus the average of log volume from day t until t-2. The explanatory variable of interest is stock sentiment, computed as the rolling sum over three days of the difference between positive and negative WSB submissions mentioning that stock. The sentiment time series is obtained with a CNN-LSTM. *,** indicate statistical relevance at the 10% and 5% levels, respectively.

2. STOCK RETURN AND WSB SENTIMENT

We employed an augmented CAPM to analyze the relationship between stock logarithmic return on day t, Ret_t, and sentiment on the previous day, Pos_{t-1} and Neg_{t-1}. Positive and negative sentiment are derived from the three distinct methodologies: VADER sentiment analysis, Support Vector Machine, and Convolutional Neural Network-Long Short-Term Memory model. Regarding sentiment impact, it can be observed variation in results, reported in table 9, 10 and 11, across the three methods. The variables in Equation 32 aren't standardized. Hence, they are on different scales. This explains why $\hat{\beta}_{pos}$ and $\hat{\beta}_{neg}$ have a much smaller magnitude compared *to* $\hat{\beta}_{market}$.

In all three tables, we observe negative coefficients denoted as $\hat{\beta}_{neg}$ and positive coefficients for $\hat{\beta}_{pos}$, indicating that a negative sentiment submission related to a particular stock may contribute to a decrease in its return, while a positive sentiment submission may lead to an increase in its return. However, the overall impact of sentiment analysis on the five selected stocks, regardless of the methodology, remained generally weak and lacked statistical significance. Nevertheless, it's noteworthy that only TSLA's beta coefficients exhibit statistical significance, implying that the influence of Reddit sentiment can vary depending on the specific stock, and its impact may not be uniform across all stocks. It is noteworthy that, among these five stocks, TSLA stands out as the sole "meme stock" that consistently garnered significant attention from 2018 to March 2023. We encourage further research to explore the effects of sentiment on returns in different industries and market conditions to gain a deeper understanding.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{market}$	1.4680***	1.1107***	1.1534***	1.1271***	1.5879***
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
$\hat{\beta}_{pos}$	0.0034*	0.0002	0.0008	0.0015	0.0004
	(0.0341)	(0.3869)	(0.7711)	(0.1989)	(0.6781)
$\hat{\beta}_{neg}$	-0.0025*	-0.0007	-0.0011	-0.0017	0.0001
	(0.0999)	(0.8647)	(0.3055)	(0.2901)	(0.9056)

Table 9. Stock Return and VADER Sentiment

Note: This table presents results from regression in Equation 32. The dependent variable is the stock log return between t-1 and t. The explanatory variables are the market return, positive and negative WSB sentiment, computed as the logarithm of 1 + rolling sum over

three days of positive and negative, respectively, WSB submissions mentioning that stock. The sentiment time series are obtained with VADER sentiment analyzer. *,*** indicates statistical relevance at the 10% and 1% levels, respectively, and the numbers in parenthesis are p-values.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{market}$	1.4722***	1.1097***	1.1539***	1.1265***	1.5876***
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
$\hat{\beta}_{pos}$	0.0001	0.0002	0.0001	0.0009	0.0012
	(0.1629)	(0.9000)	(0.9080)	(0.4088)	(0.2284)
$\hat{\beta}_{neg}$	-0.0010	-0.0017	-0.0004	-0.0016	-0.0010
	(0.1797)	(0.2143)	(0.5391)	(0.2179)	(0.4486)

Table 10. Stock Return and SVM Sentiment

Note: This table presents results from regression in Equation 32. The dependent variable is the stock log return between t-1 and t. The explanatory variables are the market return, positive and negative WSB sentiment, computed as the rolling sum over three days of positive and negative, respectively, WSB submissions mentioning that stock. The sentiment time series are obtained with a Support Vector Machine. *,*** indicates statistical relevance at the 10% and 1% levels, respectively, and the numbers in parenthesis are p-values.

	TSLA	AMZN	GOOGL	NFLX	AMD
$\hat{\beta}_{market}$	1.4754***	1.1104***	1.1539***	1.1266***	1.5866***
	(0.0000)	(0.0000)	(0.0000)	(0.0000)	(0.0000)
$\hat{\beta}_{pos}$	0.0035	0.0006	0.0004	0.0013	0.0003
	(0.0272)	(0.3553)	(0.4212)	(0.2992)	(0.8345)
$\hat{\beta}_{neg}$	-0.0029	-0.0002	-0.0005	-0.0013	-0.0009
	(0.1225)	(0.8136)	(0.6950)	(0.2684)	(0.4663)

Table 11. Stock Return and CNN-LSTM Sentiment

Note: This table presents results from regression in Equation 32. The dependent variable is the stock log return between t-1 and t. The explanatory variables are the market return, positive and negative WSB sentiment, computed as the rolling sum over three days of positive and negative, respectively, WSB submissions mentioning that stock. The

sentiment time series are obtained with a CNN-LSTM. *,*** indicates statistical relevance at the 10% and 1% levels, respectively, and the numbers in parenthesis are p-values.

Our findings are consistent with Long et al. (2023). They analyzed a "meme" company, GME, and concluded that from a statistical standpoint, meaningful relationship between Reddit sentiment and stock return start to be observed at higher frequencies. For instance, looking at intra-day WSB activity and returns. Hence, further research could investigate their co-movement at higher frequencies considering non "meme" and highly liquid stocks.

Conclusions

The primary objective of this research is to uncover and extract valuable insights from the Reddit community *WallStreetBets*. Subsequently, the goal is to assess the potential influence of this information on highly liquid and actively traded stocks. The analysis focuses on two key variables, attention and sentiment, and their impact on the logarithmic returns and abnormal turnover of these stocks. Notably, the dataset utilized in this study consists solely of submissions, in contrast to most of the previous research that has incorporated both submissions and comments from the subreddit.

By employing NLP and ML techniques, the data extracted from Reddit and re-elaborated is categorized into two distinct groups. The first category quantifies the level of attention garnered by a particular stock over the previous three days. The second category encompasses the sentiment expressed toward that specific stock during the same time span, classifying sentiment as either positive or bullish, and negative or bearish. The obtained time series data serve as explanatory variables within various linear multiple regression models. The influence of attention on abnormal turnover is statistically significant for most of the selected stocks, signifying a positive relationship between the two variables. In simpler terms, when there is a higher volume of Reddit discussions about a stock in the preceding three days, it often leads to an increase in trading activity by retail investors. Similarly, the positive coefficients linked to sentiment suggest that when positive sentiment surpasses negative sentiment, there tends to be a corresponding rise in trading volumes. However, it's important to note that the impact of sentiment on abnormal turnover have statistical significance levels below 90%. Concerning the influence on the logarithmic returns of stocks, this study explores the topic by introducing two sentiment variables into the Capital Asset Pricing Model. These variables account for both positive and negative sentiment observed over the preceding three days. This model's results show variability among the stocks, stressing the fact that that there is no universal or broadly applicable rule regarding the impact of Reddit discussions on stock markets. The effects appear to depend on the specific characteristics of each stock.

It's worth emphasizing that the stocks under consideration in this study are highly liquid. Consequently, the likelihood of retail traders causing significant price fluctuations or drastic movements is relatively low.

Furthermore, it's worth noting that only TSLA and AMD fall into the category of most-mentioned stocks and only TSLA could be classified as meme stocks to some extent. This distinction is crucial, as many researchers typically focus on meme stocks or tickers associated with less efficient and less liquid markets. In such market environments, the potential for market manipulation is more pronounced and Reddit effects are easier to isolate.

These findings have significant implications. They suggest that social media might be powerful enough to influence investment decisions, even potentially affecting financial markets directly. In traditional financial theory, retail traders are often seen as noisy traders, lacking the capacity to significantly impact market variables. However, the widespread use of social media to discuss financial matters and the rise of topic-specific communities that bring like-minded individuals together, enabling coordinated actions and cooperation, are changing this. Social media like Reddit are blurring the lines around the concept of noisy traders. They have the potential to create echo chambers, self induced consensus, Mancini et al. (2022), and trigger herd behavior. This phenomenon effectively brings together a diverse group of investors, leading them to make shared investment decisions that are substantial and far from being considered marginal or mere noise. All of this underscores the need for a deep understanding of these dynamics, not just for price discovery and trading strategies but also for regulatory purposes. Currently, there are no clear rules explicitly prohibiting retail investors from organizing themselves through forums, communities, or group chats for coordinated market actions.

This research provides a valuable set of tools and variables for monitoring and analyzing inter- actions on financial communities, particularly those like WSB. These variables may increasingly influence stock markets and in the next future could emerge as significant drivers of stock price fluctuations. To gain deeper insights into the implications of these findings, future research might consider exploring option markets. This is particularly relevant as a significant portion of retail investors opt for options trading[34] over trading the underlying stocks.

Additionally, investigating intra-day effects could offer further insights into the short-term consequences of social media discussions on stock movements within the same trading day. For enhancements to this study, incorporating comments from Reddit discussions could provide a more realistic representation, as a substantial portion of the conversation often unfolds within comment threads below the main submissions. Furthermore, improving the accuracy of sentiment classification in the ML models could be achieved by expanding the training dataset, thereby enhancing their ability

[34] The rationale behind it is that it enables to gain from leverage effects.

to classify posts. To upgrade the research, it is advisable to bifurcate the analysis into two distinct phases. The initial phase involves identifying whether a stock became a “meme” during a specific period. Subsequently, in the second phase, the analysis should focus on examining the impact of sentiment during that same timeframe.

A. Data and Methodology Appendix

All the Python and MATLAB codes will be made available upon request on a GitHub Repository.

A.1 SENTIMENT ANALYSIS

A.1.1 Fanatic Submissions

Word	Occurrences	% of Total Occurrences
gme	158,227	0.0152
buy	125,777	0.0121
stock	115,807	0.0111
market	97,184	0.0093
amc	80,767	0.0078
going	79,810	0.0077
short	77,003	0.0074
time	69,218	0.0066
money	63,903	0.0061
shares	61,937	0.0059
hold	59,602	0.0057
price	57,034	0.0055
moon	54,389	0.0052
earnings	50,597	0.0048
trading	49,967	0.0048
stocks	49,190	0.0047

robinhood	46,576	0.0045
sell	46,315	0.0044
calls	45,537	0.0043
company	43,183	0.0041
week	41,762	0.0040
wsb	40,902	0.0039
bought	39,256	0.0037
right	39,034	0.0037
options	38,878	0.0037
xb	37,138	0.0035
long	36,860	0.0035
puts	33,898	0.0032
share	32,905	0.0031
holding	32,399	0.0031
squeeze	28,620	0.0027
bb	28,601	0.0027
yolo	27,298	0.0026
high	26,323	0.0025
spy	26,195	0.0025

Table 12. Most frequent words
Note: This table displays the most frequent words in WSB submission from 2018 until March 2023. In bold some fanatic expressions.

Fanatic word	Occurrencies	% Occurrencies
yolo	27,298	0.0026
hodl	4,626	0.0004
tendies	15,656	0.0015
tendie	1,059	0.0001
dd	25,666	0.0025
diamond	14,772	0.0014
diamonds	607	0.0001
apes	22,292	0.0022
moon	54,389	0.0053
paper	7,967	0.0008

Table 13. Fanatic Words and Slangs

Note: The table reports some examples of fanatic words and the count of submission containing at least one of them.

A.1. 2 VADER Sentiment Analyzer New Words Dictionary

New Words Dictionary

new words = {‘available’: 0.8, ‘awesome’: 3.7, ‘baby’: 1.2, ‘ball’: 0.4, ‘bull’: 2.9, ‘bullshit’: -2.4, ‘buy’: 4.0, ‘future’: 1.1, ‘gain’: 2.2, ‘gamma’: 0, ‘gang’: -0.3, ‘gold’: 2, ‘good’: 2.5, ‘great’:
3.1, ‘green’: 1.9, ‘hand’: 0.1, ‘party’: 0.8, ‘penny’: -0.2, ‘poor’: 1.9, ‘possible’: 0.8, ‘potential’:
1.4, ‘power’: 2.2, ‘pretty’: 2.3, ‘probably’: 0.4, ‘top’: 2.4, ‘trade’: 0.6, ‘value’: 1.3, ‘win’: 2.7, ‘worth’: 1.9, ‘diamond hand’: 3, ‘wrong’: -1.8, ‘yolo’: 2.4, ‘dip’: -0.4, ‘dumb’: -1.9, ‘earning’:
1.8, ‘easy’: 1.6, ‘end’: -0.8, ‘hype’: 1.2, ‘idiot’: -2.6, ‘illegal’: -3.2, ‘interest’: 1.1, ‘issue’: -1.1,
‘joke’: -0.5, ‘jump’: 1.4, ‘least’: -0.4, ‘legal’: 1.9, ‘manipulation’: 2.3, ‘margin’: -0.1, ‘moment’:
0.7, ‘movement’: 0.9, ‘naked’: -1.1, ‘nice’: 2, ‘order’: 0.4, ‘panic’: -3, ‘straight’: 1, ‘strong’: 2.1,
‘stupid’: -2.1, ‘support’: 2.2, ‘target’: 1.3, ‘tendie’: 2.0, ‘to the moon’: 4, ‘cash’: 0.6, ‘concern’:
-1.3, ‘crash’: -4, ‘damn’: -1.7, ‘diamond’: 2.9, ‘hard’: -1.1, ‘hell’: -2.5, ‘high’: 2.4, ‘hodl’: 2.8,
‘hold’: 2, ‘holding’: 1.6, ‘limit’: -0.4, ‘low’: -1.7, ‘loss’: -2.5, ‘rocket’: 2.2, ‘sale’: -1, ‘scare’: -2.3, ‘scared’: -2.6, ‘seller’: -1.3, ‘selling’: -2, ‘buying’: 2, ‘attack’: -1.9, ‘problem’: -2.3, ’profit’:
2.5, ‘purchase’: 1.3, ‘shorting’: -4, ‘stonk’: 1.9, ‘star’: 2.4, ‘stop’: -0.8, ‘citron’: -4.0, ‘hidenburg’:
-4.0, ‘moon’: 4.0, ‘highs’: 2.0, ‘mooning’: 4.0, ‘long’: 4.0, ‘short’: -4.0, ‘call’: 4, ‘calls’: 4, ‘put’:
-4.0, ‘puts’: -4.0, ‘break’: 2.0, ‘tendies’: 2.0, ‘town’: 2.0, ‘overvalued’: -3.0, ‘undervalued’: 3.0,
‘sell’: -4.0, ‘gone’: -1.0, ‘gtfo’: -1.7, ‘paper’: -3, ‘bullish’: 3.7, ‘bearish’: -3.7, ‘bagholder’: -1.7,
‘money’: 1.2, ‘print’: 2.2, ‘bear’: -2.9, ‘pumping’: -1.0, ‘sus’: -3.0, ‘offering’: -2.3, ‘rip’: -4.0, ‘downgrade’: -3.0, ‘upgrade’: 3.0, ‘maintain’: 1.0, ‘pump’: 1.9, ‘hot’: 1.5, ‘drop’: -2.5, ‘rebound’:
1.5, ‘crack’: 2.5, ‘bankruptcy’: -1, ‘fuck put’: 2, ‘fuck puts’: 2, ‘fuck call’: -2, ‘fuck calls’: -2, ‘call bad’: -2, ‘put bad’: 2, ‘papers’: -3}

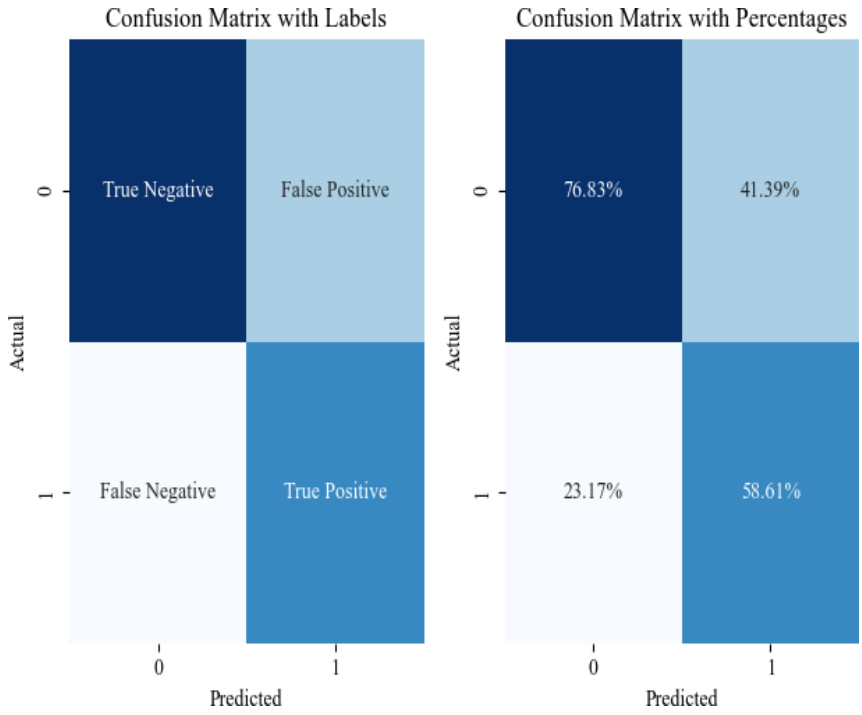

Figure 3: Confusion Matrix VADER

Note: The figure displays the confusion matrix obtained when we apply VADER to our test set. The quadrants are indicated in the right side of the figure. For instance, the True Negative quadrant refers to the percentage of
well-predicted negative submissions out of the total number of negative predicted submissions.

A.1. 3 Support Vector Machine

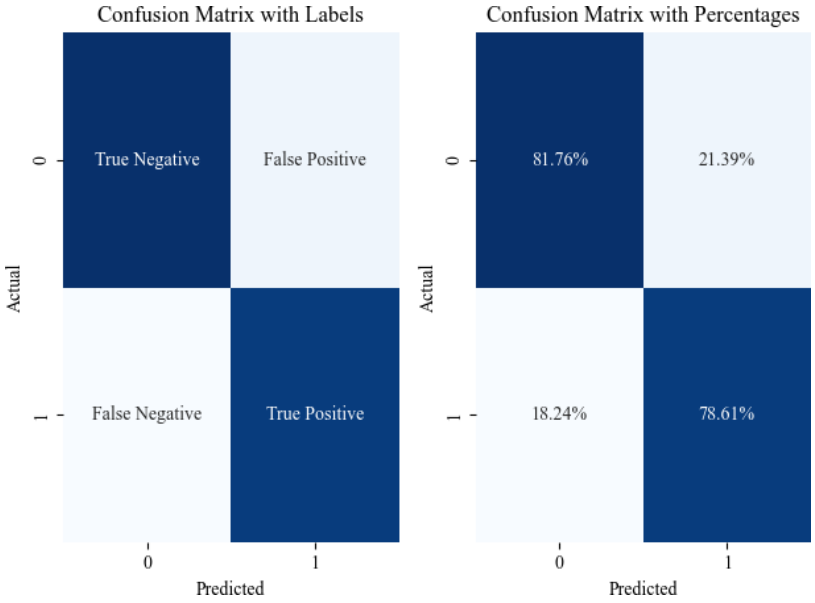

Figure 4: Confusion Matrix SVM

Note: The figure displays the confusion matrix obtained when we apply the trained SVM to our test set. The quadrants are indicated in the right side of the figure. For instance, the True Negative quadrant refers to the percentage of well-predicted negative submissions out of the total number of negative predicted submissions.

	Precision	Recall	f1-score	Support
Negative	0.818	0.789	0.803	1238
Positive	0.786	0.815	0.800	1177
Accuracy	0.802	0.802	0.802	0.802
Macro avg	0.802	0.802	0.802	2415
Weighted avg	0.802	0.802	0.802	2415

Table 14. Classification Report SVM

Note: The table summarizes the performance metrics of our Support Vector Machine. It provides reports important metrics: precision, recall, and f1-score. These are reported for each class, as well as for the overall dataset. The classification report is helpful when assessing the model's effectiveness in distinguishing between the classes, Negative and Positive.

A.1. 3 CNN-LSTM

Our architecture is build around a combination of a Convolutional network, comprehending a Convolutional and a max pooling layer, and a Long Short Term Memory layer. However, it includes other layers as well as techniques mitigating or preventing overfitting. The sequence of these layers is presented below.

Embedding Layer This layer is used to convert the tokenized and vectorized submissions into dense vectors of fixed size, the embedding size.[35] It is a commonly used NLP technique that maps tokens to vectors so that the model understands the context of a sentence. This layer requires the user to specify the input vocabulary size[36] and the maximum length of the input text[37] measured in number of tokens.

Batch Normalization Layer Batch normalization is used to stabilize and speed up the training process. It reduces the internal covariate shift[38] and mitigates the risks of issues such as vanishing gradient, that can become extremely small during backpropagation.

Dropout Layer We inserted a dropout layer to prevent overfitting. It is a regularization technique that randomly sets 30% of the input units to zero.

Convolutional Layer This is a 1D convolutional layer with 32 filters and a kernel size of 4. In brief, it applies a set of filters to the input sequence and performs element-wise products and sums, capturing local patterns in the data. The activation function is the Rectified Linear Unit[39] (ReLU). We choose it because, by introducing non-linearity, it allows the model to learn complex relationships. Furthermore, it creates sparse activations and mitigates the vanishing gradient issue.[40]

Dropout Layer Again, we set the dropout fraction to 30%.

---

[35] It is set to 64.
[36] In our case, it is equal to 16,028.
[37] It is set to 52, the maximum sequence length within the training set.
[38] Learning may slow down when the distribution of the input data to each layer changes as the parameters of the previous layer get updated. Batch normalization overcomes the problem, normalizing the inputs within a mini batch of data.
[39] $f(x) = max(0, x)$
[40] On the other hand, it can have the opposite problem, the exploding gradient. A way to improve the CNN is to use ReLU variants that address this potential issue.

Max Pooling Layer It performs 1D max pooling with a pool size of 2. We inserted it to reduce the dimension of the data by taking the maximum value within every 2-unit window. Hence, it captures the most important features.

Bidirectional LSTM Layer We introduce a bidirectional LSTM layer with 64 units. The bidi-rectional feature implies that the layer processes the sequential data both forwards and backwards. LSTM (Long Short-Term Memory) is a type of recurrent neural network, particularly suited for tasks involving language processing. Within this layer, we perform L2 regularization.[41] It mitigates the risk of overfitting by adding a penalty term to the loss function. The regularization term is equal to $\lambda ||w||^2$ where λ is the regularization strength, set to 0.02.

Dropout Layer This dropout layer sets 50% of the inputs to zero.

LSTM Layer This is another LSTM layer with 64 units. Similarly to the previous one, it applies L2 regularization with strength of 0.02.

Dropout Layer This dropout layer sets 50% of the inputs to zero.

Dense Layer This is the output layer with a single unit and a sigmoid[42] activation function. The sigmoid function outputs values between zero and one. Outputs close to zero are interpreted as negative sentiment while values close to one are classified as positive sentiment.

The model is compiled using the Adam (Adaptive Moment Estimation) optimizer[43] with a user-specified learning rate of 0.01 and the binary cross entropy[44] as loss function. The number of training epochs is 100. We insert the early stopping feature[45] to reduce the probability of overfitting.

Convolutional Neural Network

CNNs are mostly used in image processing. In particular for pattern recognition. However, following Usama et al. (2020) and Minaee et al.

[41] In pyhton, the kernel regularizer l2(0.02) applies L2 regularization with 0.02 strength.

[42] $\sigma(x) = \frac{1}{1+e^{-x}}$.

[43] It adapts the learning rates of the parameters during training and incorporates momentum. It has a better performance compared to the traditional gradient descent model.

[44] $L(y, \hat{y}) = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}))$ where y and $\hat{y}$ are the true and predicted label respectively.

[45] It has a patience of 10. Meaning that after ten epochs with no improvement on the validation loss, the training process will stop.

(2019) approach, this architecture can also be used for sentiment analysis. A CNN architecture consists of a convolutional layer followed by a pooling layer.

Convolutional Layer The convolutional layer slides its filters[46] (of dimension 1x4) along the input vector. In particular, it performs one-by-one multiplications and summations. After applying the filter, or kernel, each value is passed through a rectified linear unit function.

Max Pooling Layer Pooling layers aim to reduce the computational complexity of the model and the numbers of parameters. It accomplishes this by utilizing the "MAX" function to scale down the dimension. This layer comprehends a kernel of dimension 1x2. The stride, the step the kernel does along the input vector, is equal to 2 meaning that there is no overlapping.

Long Short-Term Memory Network

LSTM is a kind of Recurrent Neural Network that is well-suited to learn long term dependencies (Staudemeyer and Morris, 2019). The reason why it is widely use is that they overcome some of the traditional RNN's problems. In fact, they are not able to carry forward critical information. Another difficulty arises from the gradient vanishing problem. LSTM manages to maintain relevant context over time. They achieve this by discarding unnecessary information and retaining data that is essential for the decisions still to come. The recurring module in LSTM networks is made by four layers that interact with each other. The information is stored in the so called cell state and updated by four structures, the gates, described below.

Forget Gate Layer This layer handles the decision to whether keep or get rid of the previous information. The activation function is a sigmoid[47] and its output $f_t = \sigma (w_f [s_{t-1}, x_t] + b_f)$ can be interpreted as the fraction of information to carry forward.

Input Gate Layers This gate consists of two layers.A sigmoid function whose output $i_t = \sigma (w_i[s_{t-1}, x_t] + b_i)$ is the proportion of new information added to the cell state. The "new potential

[46] The number of filters can be easily set up on python.

[47] All the activation functions are applied to the weighted (with different weights for each layer) average between of the previous state's hidden layer s_{t-1} and the current input x_t plus a bias b.

information" $\hat{c}_t = \tanh(\mathbf{w}_c[s_{t-1}, x_t] + b_c)$is determined by a tanh activation function. The cell state is updated $c_t = f_t c_{t-1} + i_t \hat{c}_t$.

Output Gate Layer It consists of a sigmoid function whose value $o_t = \sigma(w_o[s_{t-1}, x_t] + b_o)$ is the fraction of cell state to output. Then, the cell state is passed through a tanh function. The final state is $s_t = o_t \tanh(c_t)$.

Layer (type)	Output Shape	Param
Embedding	(None, 52, 64)	1,025,856
Batch normalization	(None, 52, 64)	256
Dropout	(None, 52, 64)	0
Conv1D	(None, 52, 32)	8,224
Dropout	(None, 52, 32)	0
MaxPooling1D	(None, 26, 32)	0
Bidirectional LSTM	(None, 26, 128)	49,664
Dropout	(None, 26, 128)	0
Dense	(None, 1)	65

Table 15. Model Summary

Note: The table provides the summary of our CNN-LSTM architecture alongside with the input size and number of parameter to be optimized for each layer.

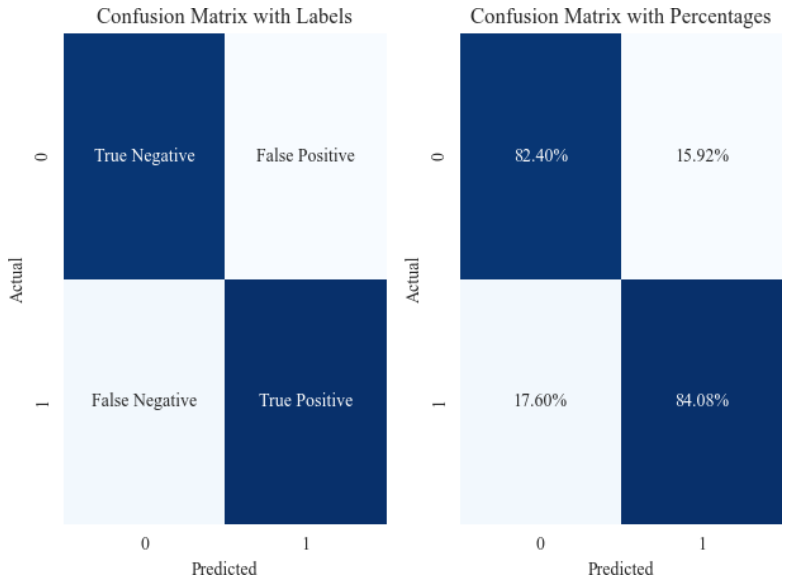

Figure 5. Confusion Matrix CNN-LSTM

Note: The figure displays the confusion matrix obtained when we apply the CNN-LSTM architecture to our test set. The quadrants are indicated in the right side of the figure. For instance, the True Negative quadrant refers to the percentage of well-predicted negative submissions out of the total number of negative predicted submissions.

A.2 ATTENTION AND SENTIMENT TIME SERIES

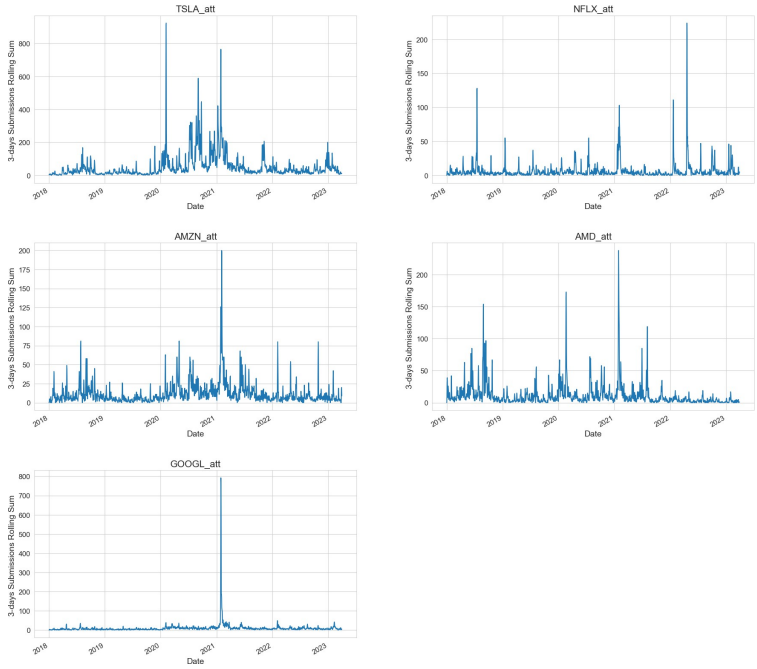

Figure 6. Attention Time Serie

Note: The figure shows, for each of the five stocks, the attention series. It is computed as the rolling sum of the number of submissions mentioning the specific stock over a three-day window.

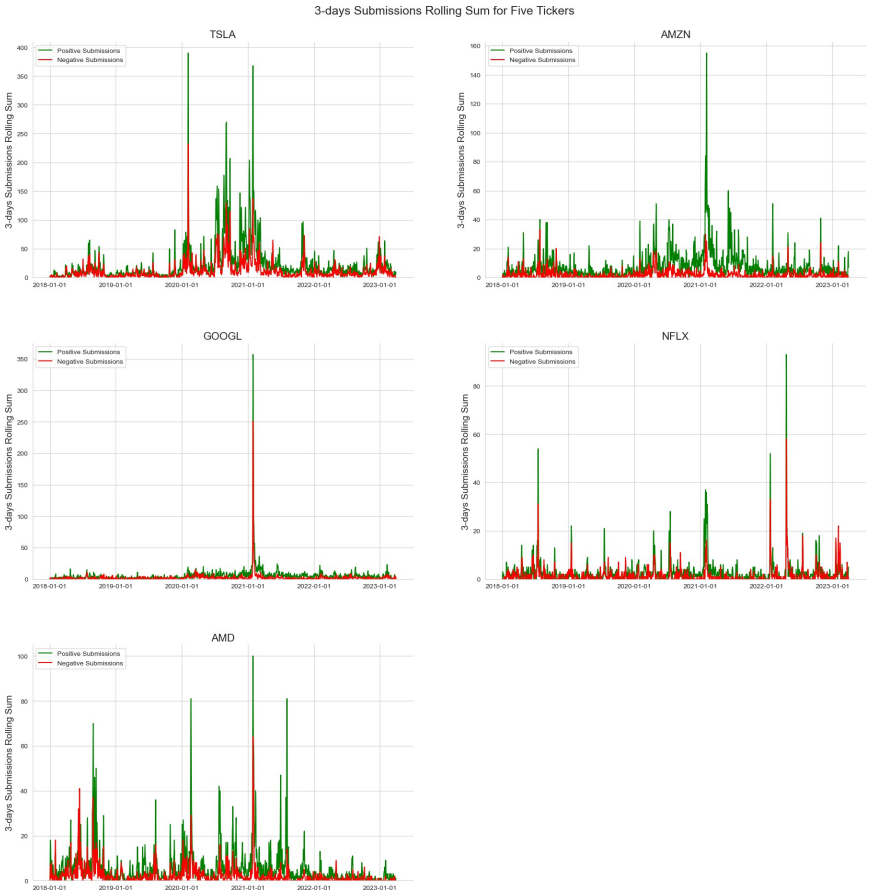

Figure 7: Sentiment Time Series VADER

Note: The figure shows, for each of the five stocks, the negative and positive sentiment series obtained adopting VADER as our classification tool. They are computed as the rolling sum of the number of positive and negative, respectively, submissions mentioning the specific stock over a three-day window.

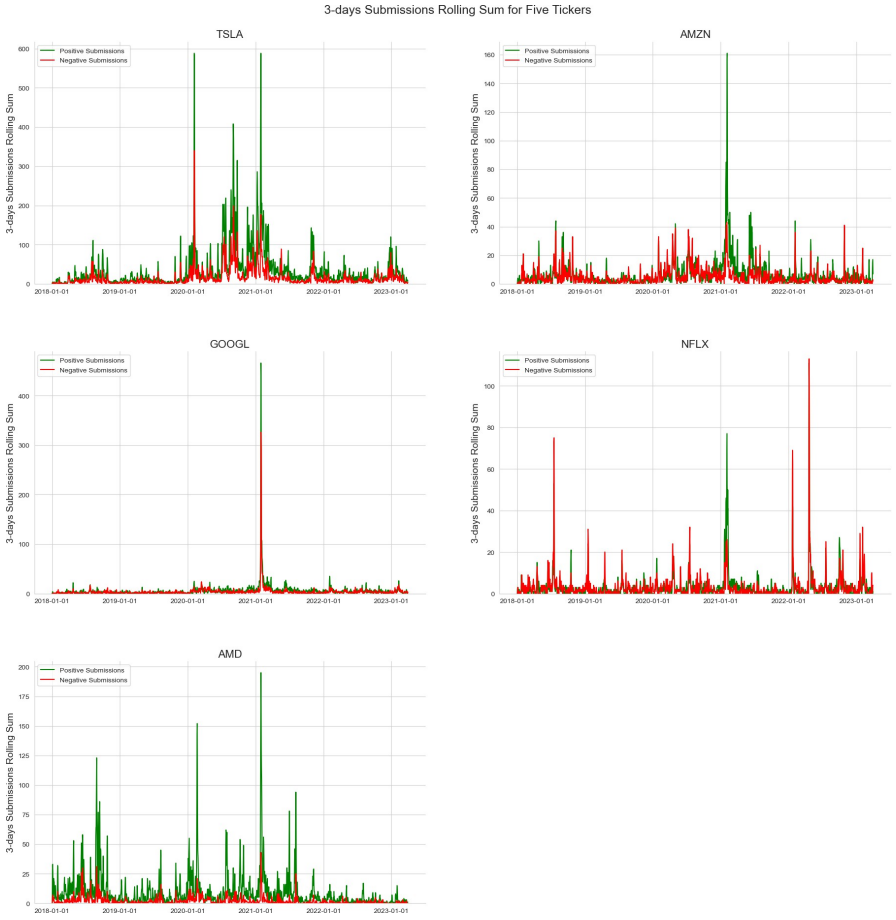

Figure 8: Sentiment Time Series SVM

Note: The figure shows, for each of the five stocks, the negative and positive sentiment series obtained adopting the SVM as our classification tool. They are computed as the rolling sum of the number of positive and negative, respectively, submissions mentioning the specific stock over a three-day window.

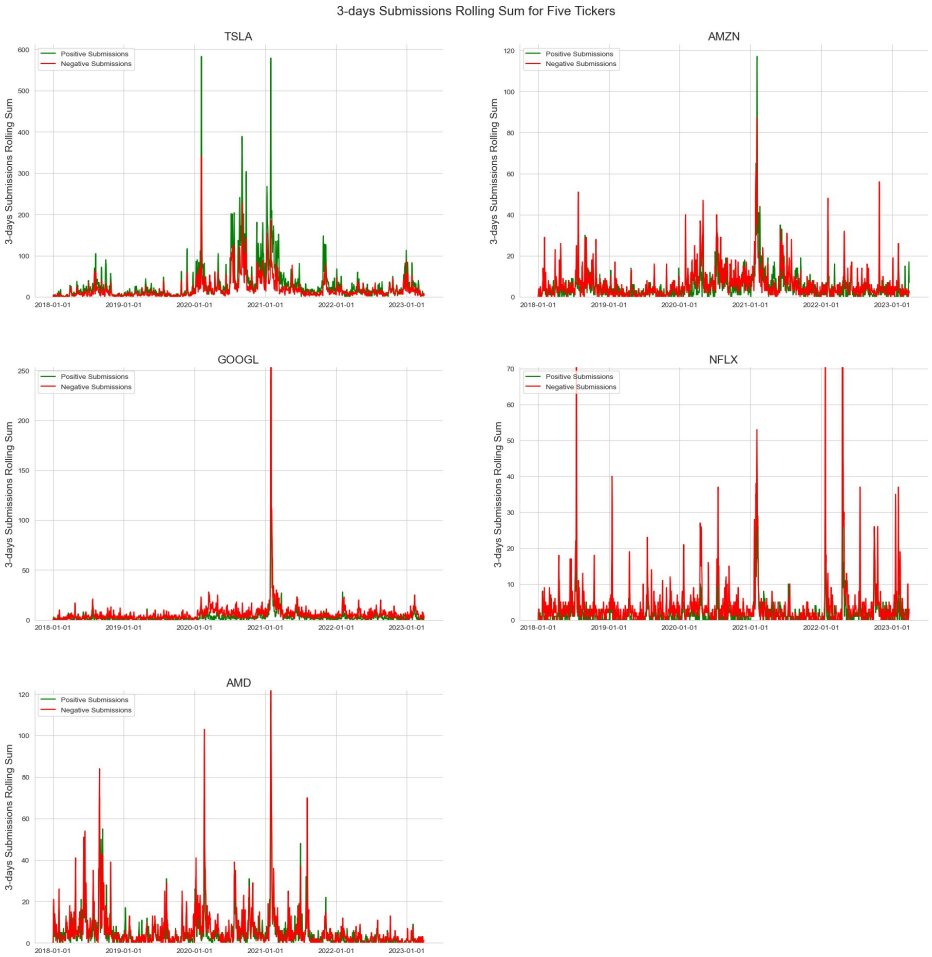

Figure 9: Sentiment Time Series CNN-LSTM

Note: The figure shows, for each of the five stocks, the negative and positive sentiment series obtained adopting the CNN-LSTM network as our classification tool. They are computed as the rolling sum of the number of positive and negative, respectively, submissions mentioning the specific stock over a three-day window.

B. Empirical Results Appendix

B.1 NEWEY-WEST STANDARD ERRORS

Newey and West (1986) described how to construct a heteroskedasticity and autocorrelation consistent covariance matrix, positive definite by construction. Given the regression's residuals $\varepsilon = y - X\beta$ with ε and yTx1 vectors, βkx1 vector and X a Txk matrix, the autocovariances matrix and their weights are:

$$S_l = \frac{1}{T} \sum_{t=l+1}^{T} \varepsilon_t \varepsilon_{t-l}'; \tag{34}$$

$$w_l = 1 - \frac{l}{L+1}. \tag{35}$$

They capture the autocovariances of the residuals at different lags l with a maximum lag of L. The Newey-West covariance matrix is obtained as:

$$\hat{\Sigma}_{NW} = S_0 + \sum_{l=1}^{L} (W_l S_l + S_l' W_l), \tag{36}$$

and the standard errors are:

$$SE_{NW} = \sqrt{diag(\hat{\Sigma}_{NW})}. \tag{37}$$

References

Alayba, A., Palade, V., England, M., and Iqbal, R. (2018). A combined CNN and LSTM model for Arabic Sentiment Analysis. In Holzinger, A., Kieseberg, P., Tjoa, A., and Weippl, E., editors, *Machine Learning and Knowledge Extraction*, volume 11015 of *Lecture Notes in Computer Science*. Springer.

Anand, A. and Pathak, J. (2021). Wallstreetbets against Wall Street: The role of Reddit in the GameStop short squeeze. *SSRN Electronic Journal*.

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.

Banerji, G. and McCabe, C. (2021). Reddit legend Keith Gill boosts stake in GameStop. *The Wall Street Journal*.

Behrendt, S. and Schmidt, A. (2018). The Twitter myth revisited: Intraday investor sentiment, Twitter activity, and individual-level stock return volatility. *Journal of Banking & Finance*, 96:355–367.

Betzer, A. and Harries, J. P. (2022). How online discussion board activity affects stock trading: The case of GameStop. *Financial Markets and Portfolio Management*, 36(4):443–472.

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

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92)*, pages 144–152, Pittsburgh.

Broadstock, D. C. and Zhang, D. (2019). Social-media and intraday stock returns: The pricing power of sentiment. *Finance Research Letters*, 30:116–123.

Cao, H. H., Coval, J. D., and Hirshleifer, D. A. (2001). Sidelined investors, trading-generated news, and security returns. Dice Working Paper No. 2000-2.

Chacon, R. G., Morillon, T. G., and Wang, R. (2023). Will the Reddit rebellion take you to the moon? Evidence from Wallstreetbets. *Financial Markets and Portfolio Management*, 37(1):1–25.

Duz Tan, S. and Tas, O. (2021). Social Media Sentiment in international stock returns and trading activity. *Journal of Behavioral Finance*, 22(2):221–234.

Gandhi, U. D., Malarvizhi Kumar, P., Chandra Babu, G., et al. (2021). Sentiment analysis on Twitter data by using Convolutional Neural Network (CNN) and long short term memory (LSTM). *Wireless Personal Communications*.

Gianstefani, I., Longo, L., and Riccaboni, M. (2022). *The echo chamber effect resounds on financial markets: A social media alert system for Meme Stocks*.

Glenski, M., Weninger, T., and Volkova, S. (2019). *Improved forecasting of cryptocurrency price using social signals*.

Greene, J. and Smart, S. (1999). Liquidity provision and noise trading: Evidence from the "Investment Dartboard" column. *The Journal of Finance*, 54(5):1885–1899.

Greene, W. H. (2019). *Econometric Analysis, Global Edition*. Pearson, 8th edition. Published on December 30, 2019.

Gu, C. and Kurov, A. (2020). Informational role of social media: Evidence from Twitter sentiment. *Journal of Banking & Finance*, 121:105969.

Hu, D., Jones, C. M., Zhang, V., and Zhang, X. (2021). *The rise of Reddit: How social media affects retail investors and short-sellers' roles in price discovery*.

Hutto, C. J. and Gilbert, E. E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. Eighth International Conference on Weblogs and Social Media (ICWSM-14).

Jin, Z., Yang, Y., and Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and LSTM. *Neural Comput & Applic*, 32:9713–9729.

Kecman, V. (2001). *Support Vector Machines – an introduction*. In Wang, L., editor, *Support Vector Machines: Theory and Applications*, volume 177 of Studies in Fuzziness and Soft Computing. Springer, Berlin, Heidelberg.

Kharde, V. and Sonawane, S. (2016). Sentiment Analysis of Twitter data: A survey of techniques. *International Journal of Computer Applications*, 139:5–15.

Kohonen, T. (1988). An introduction to neural computing. *Neural Networks*, 1:3–16.

Lai, S., Xu, L., Liu, K., and Zhao, J. (2015). Recurrent Convolutional Neural Networks for Text Classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1).

learn Contributors, S. (2023). *Support Vector Machines (SVM) - scikit-learn Documentation*.

Long, C., Lucey, B., Xie, Y., and Yarovaya, L. (2023). "I just like the stock": The role of Reddit sentiment in the GameStop share rally. *Financial Review*, 58:19–37.

Lopez, M. (2018). Advances in Financial Machine Learning. Wiley, 1st edition.

Lyócsa, S., Baumöhl, E., and Výrost, T. (2022). YOLO trading: Riding with the

herd during the GameStop episode. *Finance Research Letters*, 46, Part A:102359.

Mahadevaswamy, U. and Swathi, P. (2023). Sentiment analysis using bidirectional LSTM network. *Procedia Computer Science*, 218:45–56.

Mancini, A., Desiderio, A., Clemente, R. D., and et al. (2022). Self-induced consensus of Reddit users to characterise the GameStop short squeeze. *Scientific Reports*, 12:13780.

Minaee, S., Azimi, E., and Abdolrashidi, A. (2019). *Deep-Sentiment: Sentiment Analysis Using Ensemble of CNN and Bi-LSTM Models*.

Müller, A. C. and Guido, S. (2016). Introduction to Machine Learning with Python. O'Reilly Media, Inc.

Newey, W. K. and West, K. D. (1986). A simple, positive semi-definite, Heteroskedasticity and Autocorrelation-Consistent Covariance Matrix. Technical Working Paper 0055, National Bureau of Economic Research.

O'Shea, K. and Nash, R. (2015). *An introduction to Convolutional Neural Networks*. ArXiv e-prints.

Padalkar, N. R. (2021). "Stonks to the moon": Evidence from Reddit posts and corresponding market manipulation. In *AMCIS 2021 Proceedings*, page 4.

Parkinson, M. (1980). The Extreme Value Method for estimating the variance of the rate of return. *The Journal of Business*, 53:61–65.

Pedersen, L. (2023). Lasse Pedersen on GameStop and predatory trading. Princeton University, Department of Economics.

Pedersen, L. H. (2021). *Game on: Social networks and markets*. NYU Stern School of Business Forthcoming.

Reddit WallStreetBets Community (Accessed 2023). Reddit WallStreetBets.

Salvaris, M., Dean, D., and Tok, W. (2018). *Convolutional Neural Networks. In Deep Learning with Azure*. Apress, Berkeley, CA.

SimilarWeb (2023a). Regional distribution of desktop traffic to Reddit.com as of april 2023 by country. Retrieved September 2, 2023.

SimilarWeb (2023b). Worldwide visits to Reddit.com from november 2022 to april 2023 (in billions). Retrieved September 2, 2023.

Sprenger, T., Tumasjan, A., Sandner, P., and Welpe, I. (2014). Tweets and trades: The information content of stock microblogs. *European Financial Management*, 20:926–957.

Sprenger, T. O. and Welpe, I. M. (2010). *Tweets and trades: The information content of stock microblogs*.

Statista (2023). *Number of Monthly Active Reddit users worldwide from 2015 to 2022 (in millions)*. Retrieved September 2, 2023.

Staudemeyer, R. and Morris, E. (2019). *Understanding LSTM – a tutorial into Long Short-Term Memory Recurrent Neural Networks*.

Sul, H. K., Dennis, A. R., and Yuan, L. I. (2017). Trading on Twitter: Using social media sentiment to predict stock returns. *Decision Sciences*, 48(3):454–488.

Suthaharan, S. (2016). Support Vector Machine. In *Machine Learning Models and Algorithms for Big Data Classification*, volume 36 of Integrated Series in Information Systems, page Chapter 9. Springer.

Team, P. D. (2023). PRAW Quick Start Guide.

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *The Journal of Finance*, 62(3):1139–1168.

Tetlock, P. C., Saar-Tsechansky, M., and Macskassy, S. (2008). More than words: Quantifying language to measure firms' fundamentals. *The Journal of Finance*, 63(3):1437–1467.

Times, T. N. Y. (2021). *How GameStop became a red-hot investing idea on Reddit*.

Umar, Z., Gubareva, M., Yousaf, I., and Ali, S. (2021). A tale of company fundamentals vs sentiment driven pricing: The case of GameStop. *Journal of Behavioral and Experimental Finance*, 30:100501.

Usama, M., Ahmad, B., Song, E., Hossain, M. S., Alrashoud, M., and Muhammad, G. (2020). Attention-based sentiment analysis using Convolutional and Recurrent Neural Network. *Future Generation Computer Systems*, 113:571–578.

Witts, D. W., Tortosa-Ausina, E., and Arribas, I. (2021). The irrational market: Considering the effect of the online community Wall Street Bets on financial market variables. Technical Report 2021/13, Economics Department, Universitat Jaume I, Castelló n (Spain).

Wooley, S., Edmonds, A., Bagavathi, A., and Krishnan, S. (2019). Extracting Cryptocurrency Price Movements from the Reddit Network Sentiment.

Zhao, N., Gao, H., Wen, X., and Li, H. (2021). Combination of Convolutional Neural Network and gated recurrent unit for aspect-based sentiment analysis. IEEE Access, 9:15561–15569.

Zhou, Z.-H. (2021). *Machine Learning*. Springer Singapore, 1 edition.