Development of Context-based Sentiment Classification for Intelligent Stock Market Prediction

Nurmaganbet Smatov , Ruslan Kalashnikov , Amandyk Kartbayev *

Posted Date: 8 April 2024

doi: 10.20944/preprints202404.0563.v1

Keywords: sentiment analysis; neural networks; stock price prediction; text-mining; deep learning.

Article

Development of Context-Based Sentiment Classification for Intelligent Stock Market Prediction

Nurmaganbet Smatov, Ruslan Kalashnikov and Amandyk Kartbayev *

Kazakh-British Technical University; nu_smatov@kbtu.kz; r_kalashnikov@kbtu.kz; a.kartbayev@kbtu.kz
* Correspondence: a.kartbayev@gmail.com; Tel.: +7(705)7888330

Abstract: This paper presents a novel approach to sentiment analysis specifically customized for predicting stock market movements, bypassing the need for external dictionaries which are often unavailable for many languages. Our methodology directly analyzes textual data, with a particular focus on context-specific sentiment words within neural network models. This specificity ensures our sentiment analysis is both relevant and accurate in identifying trends in the stock market. We employ sophisticated mathematical modeling techniques to enhance both the precision and interpretability of our models. Through meticulous data handling and advanced machine learning methods, we leverage large datasets from Twitter and financial markets to examine the impact of social media sentiment on financial trends. We achieved an accuracy exceeding 75%, highlighting the effectiveness of our modeling approach, which we further refined into a convolutional neural network model. This achievement contributes valuable insights into sentiment analysis within the financial domain, thereby improving the overall clarity of forecasting in this field.

Keywords: sentiment analysis; neural networks; stock price prediction; text-mining; deep learning.

1. Introduction

The growth of online services has given people the chance to share their thoughts on a broad range of subjects, such as products, services, movies, companies, and political figures. This widespread sharing of personal views has resulted in the gathering of large amounts of information, requiring efficient techniques to assess and understand people's perspectives on different topics. A significant challenge in this process is the subjective nature of emotional evaluation in textual content. Studies have shown that the same text can mean varying interpretations among different individuals, sometimes leading to completely opposite assessments.

As technological advancements continue, the significance of public sentiment in influencing a wide range of decisions has become increasingly apparent. This is particularly evident in the field of behavioral finance, where financial decisions are often seen as being driven by emotional factors[1]. It is posited that public sentiment, as reflected in social media outputs like Twitter, may have a comparable impact on stock market prices. For instance, tweets carrying positive or negative sentiments, especially those with financial hashtags, are thought to potentially influence stock movements. The prevailing sentiment of the day is hypothesized to affect stock prices the following day – negative sentiment could lead to a decrease in prices, whereas positive sentiment might cause an increase. Additionally, the influence of tweets on stock prices is believed to be proportional to the number of followers of the Twitter account, suggesting a greater impact from accounts with larger followings.

Furthermore, natural language texts, inherently unstructured, present additional difficulties for processing. These texts often include elements such as sarcasm, humor, and typographical errors, which can be challenging for both human and machine interpretation. Another layer of complexity arises from the language-specific nature of sentiment analysis methods. For instance, techniques developed for English texts may not be directly applicable to texts in Kazakh, highlighting a limitation in the universality of these methods.

 doi:10.20944/preprints202404.0563.v1

The tonality of text, crucial in sentiment analysis, is also context-dependent. The emotional connotation of specific words can vary significantly across different domains, affecting the interpretation of sentiment. Therefore, a critical task in sentiment analysis is the automatic extraction and classification of opinions from texts. This involves determining the presence of a subjective component in the text and categorizing the text based on its tonality, which may range from positive to negative and possibly include additional classes. Tonality is defined here as the emotional evaluation that the author of the text expresses about a particular object or subject.

There has been considerable research in this area, particularly focusing on the prediction of stock prices using Twitter sentiment analysis. [2] and [3] investigated the impact of tweet sentiment on the Dow Jones Industrial Index (DJII), utilizing Granger causality to unveil a significant association between the mood of calmness in tweets and DJII values. Their approach, informed by these insights, yielded considerable gains within a 40-day period. Further, [4] examined how tweet sentiment correlates with stock price fluctuations and trading volumes, employing diverse models to forecast stock prices for entities like General Electric, Intel, and IBM, with an approximate 70% accuracy rate. These findings collectively underscore a substantial link between Twitter sentiment and stock market dynamics. Sentiment analysis, especially when applied to tweets, has found utility in forecasting outcomes in political and social arenas. [5] created a framework for learning public sentiment towards the 2020 U.S. presidential election using Twitter data. Additionally, [6] deployed classification techniques, such as Random Forest, SVM, and Naive Bayes, to ascertain stock directions from news sentiment, securing an accuracy between 85% to 94% across various scenarios.

The introduction of contextualized embedding has significantly influenced sentiment analysis, particularly for social media content such as tweets. The work of [7] stands out in this field; it assesses a range of word representation models, including Transformer-based auto-encoder models like RoBERTa, and showcases their effectiveness in capturing the intricacies of the informal and evolving language found in tweets. This research highlights the advantages of contextualized models over static ones for sentiment analysis, aligning with the findings of [8], [9], and [10]. These studies have proven the effectiveness of models like BERT across various NLP tasks, affirming the resilience of these advanced models. [11] explores the complexities of automating tweet analysis, a crucial aspect for understanding the inherent challenges of sentiment analysis. Their findings further support [7]'s conclusions, underscoring the importance of a profound comprehension of natural text to effectively navigate the informal syntax characteristic of tweets.

The groundwork for modern sentiment analysis methodologies was significantly influenced by the development of Vector Space Models (VSMs), as highlighted by [12] and [13]. Their pioneering efforts in exploring semantic similarities within VSMs and introducing count-based approaches like Bag-of-Words (BoW) and TF-IDF paved the way for the evolution of more complex word representation models [14]. This shift from basic VSMs to advanced embedding strategies represents a major leap forward in the field of sentiment analysis. The introduction of Word2Vec and FastText by [15], which proposed the application of dense vectors for word representation, was a pivotal moment. The advancements by [16-18] further addressed the complexities introduced by Twitter's ever-changing language, showcasing the significance of these developments in tackling the nuances of social media text analysis.

The significance of creating sentiment-specific embedding for sentiment analysis is underscored by [19], who introduced the Sentiment-Specific Word Embedding model (SSWE). This innovative approach integrates sentiment information within embeddings to solve the problem of words that share syntactic similarities yet exhibit divergent sentiment polarities being closely positioned in vector space. Similarly, [20] developed an attention-based LSTM framework aimed at forecasting the directional trends of major indices and individual stock prices, leveraging headlines from financial news. This model demonstrated competitive performance against advanced models that blend knowledge graphs for event embedding learning. In a related work, [21] applied Random Matrix Theory (RMT) and information theory to dissect the correlation and flow of information between The New York Times' publications and global financial indices. Their findings reveal a profound

 doi:10.20944/preprints202404.0563.v1

connection between news content and global markets, positioning news as a pivotal influence on market dynamics.

In the development of sentiment lexicons we have explored several approaches, notably dictionary-based and corpus-based methods. The dictionary-based methods have been extensively discussed in literature, with notable contributions from [22,23]. While these methods face challenges in processing social media content, mainly due to misspellings and the use of out-of-vocabulary words, this opens up exciting opportunities for further exploration and the development of more robust techniques capable of understanding the nuances of social media language. Corpus-based methods, in contrast, are better suited for handling social media data. These methods utilize a range of statistical and linguistic features to distinguish opinion words from other words, as demonstrated in the works of [24,25]. Another key category of methods encompasses both dictionary-based and corpus-based approaches and involves graph-based techniques. [26] introduced an innovative strategy for building a lexical network by using a lot of unlabeled data, followed by the application of a graph propagation algorithm. This approach, alongside similar strategies that utilize graph or label propagation for the extraction of opinion words, has been further investigated by researchers like [27,28], underscoring the versatility and efficacy of graph-based methods in sentiment lexicon construction.

In this study, we've deliberately chosen an approach that bypasses the need for external dictionaries, acknowledging that such resources are not universally available across languages. This choice leads us to concentrate directly on the text itself, with a particular emphasis on identifying sentiment-related terms pivotal for the context at hand as we construct our neural network model. This method serves to bridge the gap of knowledge by ensuring our sentiment analysis is both contextually relevant and accurately reflective of stock market trends. Our investigation employs a range of strategies, enhancing not just the precision of our models but also their interpretability. We aim to achieve a high level of clarity in our findings, utilizing sophisticated mathematical modeling techniques. A significant strength of our study lies in its thorough data handling and the employment of machine learning techniques, particularly how we process extensive datasets from Twitter and financial records. By leveraging advanced data preparation and machine learning methods, this work enriches the growing field of sentiment analysis in finance, marking a key contribution by focusing on raw text analysis to derive insights into market sentiments.

2. Materials and Methods

2.1. Data Preprocessing

Our work commenced with the utilization of Python libraries to gather datasets containing references to stock names from various sources. One primary dataset originated from the Twitter Sentiment Analysis Dataset available on Huggingface, comprising 1,578,627 classified tweets[29]. Each tweet is labeled as '1' for positive sentiment and '0' for negative sentiment. In addition, we incorporated the Twitter7 dataset[30], a substantial collection of approximately 476 million tweets amassed from June to December 2009. This dataset, part of the Stanford Large Network Data Collection (SNAP), is approximately 25GB in size and encompasses data from 17 million users, including 476 million tweets, 181 million URLs, 49 million hashtags, and 71 million retweets. The structure of each data entry includes time, user, and tweet content.

Furthermore, we sourced stock prices data from Yahoo Finance. This dataset features a subset of US-listed instruments, updated daily based on trading volume and information availability[31]. It's important to note that this dataset may have gaps due to its selection criteria, implying that certain instruments may temporarily appear or disappear from the dataset. The market data includes various return calculations over different timespans. The Table 1 summarizes the datasets used in our study, focusing on their sources and a brief description, along with a mention of machine learning methods and algorithms that have been previously applied to similar data in the field.

 doi:10.20944/preprints202404.0563.v1

Table 1. Description of the datasets.

Dataset Name	Source	Description	Size	Applied methods
Twitter Sentiment Analysis	Kaggle, Huggingface	Classified tweets with sentiment analysis annotations.	About 60GB, 1,578,627 tweets	Random Forest, SVM, Naive Bayes
Twitter7	Stanford Large Network Data Collection (SNAP)	Collection of tweets capturing user interactions like URLs, hashtags, and retweets, with temporal and user data.	Approximately 25GB, 476 million tweets	Neural Networks, Decision Trees
Stock Prices (Yahoo Finance)	Yahoo Finance	Subset of US-listed stock instruments, reflecting daily updates based on trading volume and market dynamics.	Daily updated, Subset of US-listed instruments	LSTM, Gradient Boosting, Regression

The preprocessing stage involved several steps using Python, focusing on data cleaning and filtering to prepare the data for analysis. The process included:

- Removing XML/JSON Characters: We eliminated irrelevant characters (e.g., &gt, &amp) using a parser, as they hold no value for sentiment analysis.
- Decoding Data: Complex symbols in tweets were decoded into simple, understandable characters using UTF-8 encoding, the most widely accepted method for data decoding.
- Standardizing Apostrophes and Slangs: Apostrophes were uniformed to avoid ambiguity, and slangs were standardized for computational understanding.
- Converting Created Words: User-generated words in tweets were reformatted into a standard format for better computational interpretation.

We opted for a Regex Tokenizer over the standard tokenizer due to its effectiveness in handling the less standardized nature of tweet data, which often includes extra spaces and symbols. This tokenizer uses regular expressions to determine split positions in the text. For feature vectorization, we employed Hashing TF-IDF, a method commonly used in text mining to reflect the importance of a term in a document relative to the corpus. This approach helps in converting words into vectors for subsequent sentiment classification and prediction. Additionally, we have merged various tweets containing sentiment messages with price information, as illustrated in Figure 1, to make it accessible for training.

Figure 1. Integration of sentiment-based tweets and price data.

The processed dataset was then divided into two parts: 85% for training and 15% for testing a hybrid neural network classification model. The training involved generating feature vectors for each tweet, enabling the classification of tweets as positive or negative. The model achieved an accuracy of 0.848362375637 in the best case. Considering that human sentiment classification can be subjective with about 10% debatable, this accuracy is a commendable starting point.

 doi:10.20944/preprints202404.0563.v1

2.2. Context-Oriented Sentiment Analysis

The task of context-oriented tone analysis of text documents (reviews) can be described as follows: for each review d_i from the available set of reviews $D = \{d_1, d_2, \dots, d_n\}$ it is necessary to find a subset $A_i = \{a_{i1}, a_{i2}, \dots, a_{i|A_i|}\}$ contexts of $A = \{a_1, a_2, \dots, a_k\}$ that are mentioned in this review, and for each one $a_{ij} \in A_i$ define a tone from the set $Y = \{-1,0,1\}$: "negative", "neutral" and "positive" respectively. For the case when a review contains contradictory judgments for one context, a special label is specified for it. C - conflict. I.e. for each context it is necessary to choose a label from the set of Y. This task can be broken down into three separate subtasks: context extraction, tone detection, and opinion abstracting for the review. Let us write down the formal formulation of each subtask.

2.2.1. Subtask 1: Context Extraction

This subtask can be viewed as the task of classifying objects (reviews) into overlapping classes $S = \{s_1, s_2, \dots, s_p\}$ - set of feedback sentences $d \in D$. $A = \{a_1, a_2, \dots, a_k\}$ - finite set of contexts known for the given subject domain. $A^* = \{0,1\}^k$ - the set of admissible responses of the classifier. $a^*: S \to A^*$ - an unknown target dependency whose values are known only on the objects of the finite training sample $S_m = \{(s_1, A_1^*), \dots, (s_m, A_m^*)\}$. We need to build an algorithm $a: D \to A^*$, capable of classifying an arbitrary object $s_i \in S$.

2.2.2. Subtask 2: Identifying tone in relation to contexts

Can be viewed as a classification problem into non-overlapping classes. $S = \{s_1, s_2, \dots, s_p\}$ - set of sentences of some review $d \in D$ and for each $s_i \in S$ is defined $A_i^* = \{a_{i1}^*, a_{i2}^*, \dots, a_{ir_i}^*\}, r_i \le k$ - the set of contexts mentioned in this paper. $Y = \{-1,0,1\} -$ a set of tone labels that correspond to the scale "negative" - "neutral" - "positive". $y^*: S \to Y^p$ an unknown target dependence whose values are known only on the objects of a finite training sample $S_m = \{(s_1, a_{11}^*, y_{11}^*), \dots, (s_m, a_{mr_m}^*, y_{mr_m}^*)\}$. It is required to construct an algorithm $y: S \to Y^p$ capable of classifying an arbitrary object. $s_i \in S$.

2.2.3. Subtask 3: Abstracting an opinion for a review of a $D = \{d_1, d_2, \dots, d_n\}$ - a set of text documents, with each review d_i consists of multiple sentences $S_i = \{s_{i1}, s_{i2}, \dots, s_{i|s_i|}\}$.

For each sentence $s_{ij}, i \in [1, n], j \in [1, |Si|]$ there is a set of pairs $\{(a_{ijl}^*, y_{ijl}^*)\}_{l=1}^{L_{ij}}$ where $a_{ijl}^* - l$-th context of the sentence s_{ij}, y_{ijl}^* - context tone a_{ijl}^*. It is required to construct an algorithm capable for each $d_i \in D$ specify a set of pairs (a_{ih}, y_{ih}) such that, $\forall s_{ij} \in d_i, \forall (a_{ijl}^*, y_{ijl}^*), l \in [1, L_{ij}] \exists (a_{ih}, y_{ih}): a_{ih} = a_{ijl}^*, y_{ih} \in Y^* \cup C$.

$$Y^* = \{y_{ijl}^*, y_{\text{qwe}}^*, C\}, (q, w, e): \exists (a_{\text{qwe}}^*, y_{\text{qwe}}^*): a_{\text{qwe}}^* = a_{ijl}^* \quad (1)$$

C - conflict label. $h \in [1, H_i]$ where H_i - is the total number of contexts encountered in the text $d_i \in D$, not including repeated contexts, $H_i = \sum_{j=1}^{|S_i|} L_{ij} - L_i^*$.

2.2.4. Context extraction task as a set of binary classification tasks

The context extraction task, considered as a multiclass classification task into overlapping classes, can be reduced to several binary classification tasks. It was decided to use one-versus-all strategies: for each of the contexts declared for the subject domain, a separate classifier is built, trained on data related to one context against data for all other contexts.

Formally, if $D = \{d_1, d_2, \dots, d_n\}$ - a set of reviews, and each text $d_i \in D$ consists of a set of sentences: $d_i = \{s_1, s_2, \dots, s_n\}$, which are subject to classification, $A = \{a_1, a_2, \dots, a_k\} -$ a finite set of contexts, $p^* = \{0,1\}^k -$ a set of admissible responses, $S = \{(s_1, p_1^*), \dots, (s_m, p_m^*)\}$ - a finite training sample, then for each context $a_i \in A: z_i = \{z_{i1}, z_{i2}, \dots, z_{im}\}$ - a new vector of labels, and $z_{ij} = 1$ if $p_{ji}^* = 1$ otherwise $z_{ij} = 0$. $S_i = \{(s_1, z_{i1}), \dots, (s_m, z_{im})\}$ - new training sample.

We get a set of k of training samples, one for each context. Then $c = \{c_1, c_2, \dots, c_k\}$ - set of classifiers, where $c_i, i \in [1; k]$ has been trained on the corresponding sample S_i. For each context under consideration, reference terms can be selected from a marked-up training collection: nouns, verbs, adjectives, and adverbs that are characteristic in describing the context. Once the dictionary of reference terms has been compiled, each new verifiable term that has a distributed representation of the $\vec{a} = (a_1, \dots, a_n)$ can be mapped to a specific context A^* in one of two ways:

- element-by-element comparison with each reference term $\overrightarrow{b_l} \in B_{A^*}$ of the context A^*;

 doi:10.20944/preprints202404.0563.v1

- by calculating the cumulative similarity to the context A^*. B_{A^*} - set of context reference terms A^*. Each $\overrightarrow{b_l} \in B_{A^*}$ has a distributed vector representation $\overrightarrow{b_t} = (b_1, \dots, b_n)$. Cosine similarity is used as a measure of proximity between vectors in both cases.
  For the first method(2):

$$\text{sim}_1(\vec{a}, A^*) = \max_{i=1,\dots,k} \frac{\vec{a} \cdot \overrightarrow{b_l}}{\| \vec{a} \| \cdot \| \overrightarrow{b_l} \|} \quad (2)$$

  $\overrightarrow{b_l} \in B_{A^*}, k = |B_{A^*}| -$ number of reference terms.
  For the second method(3):

$$\text{sim}_2(\vec{a}, A^*) = \sum_{i=1}^{k} \frac{\vec{a} \cdot \overrightarrow{b_l}}{\| \vec{a} \| \cdot \| \overrightarrow{b_l} \|'} \quad (3)$$

$\overrightarrow{b_l} \in B_{A^*}, k = |B_{A^*}| -$ number of reference terms. If the obtained proximity value exceeds some threshold, the tested term is considered contextual. The threshold value in each case can be determined experimentally.

The tone for each of the retrieved texts can be determined by its context. In order to give an interpretation of the context in a numerical format, a dictionary for the subject area under consideration is used. It is then used to generate a set of features for each evaluated context, which can be used as input to a neural network based classifier. To compile the dictionary, candidates for emotional expressions are selected from the texts under consideration: all nouns, adjectives, verbs and adverbs, as well as individual text fragments. Next, it is necessary to numerically determine the emotional coloring of each term. For this purpose, semantic similarity is used again (4-5).

$$\text{sim}^+(\vec{a}, B^+) = \sum_{i=1}^{k} \frac{\vec{a} \cdot \overrightarrow{b_l}}{\| \vec{a} \| \cdot \| \overrightarrow{b_l} \|} \quad (4)$$

$$\text{sim}^-(\vec{a}, B^-) = \sum_{i=1}^{k} \frac{\vec{a} \cdot \overrightarrow{b_l}}{\| \vec{a} \| \cdot \| \overrightarrow{b_l} \|} \quad (5)$$

Here $B^+ n B^-$- sets of reference emotional terms of positive and negative tonalities, respectively. The composition of these sets is determined by the expert, each of them contains expressions for positive and negative tonalities. Each element of these sets has a distributed representation $\overrightarrow{b_l} = (b_1, \dots, b_n). \vec{a} = (a_1, \dots, a_n)$ - distributed representation of the term under consideration. Finally, $\text{sim}^+(\vec{a}, B^+) n \text{sim}^-(\vec{a}, B^-)$-are the values of the total similarities. The tone whose total similarity is greater modulo is chosen as the tone of the word under consideration.

Once the emotional color of each expression has been determined, a complete tone dictionary is obtained, matching each lexical expression with a tone score based on semantic similarity. Determining tone separately for each context is complicated by the fact that, generally speaking, there may be several of them in a sentence. This is supposed to be done in two steps: first, find all contexts mentioned in the message, then, for each of these contexts, determine the total tonality.

Even though we depend only on our model, we also integrate the financial sentiment dictionary developed by Loughran and McDonald[32] to enhance and validate the outcomes of our classification model. This English sentiment lexicon is specifically designed for analyzing financial documents, categorizing words into six sentiments critical in financial contexts: negative, positive, litigious, uncertainty, constraining, or superfluous. By utilizing this lexicon, we can more accurately interpret the emotional tone of financial texts, which is pivotal in predicting stock market movements. A higher prevalence of words labeled as "positive" within the analysis suggests an increasing trend in stock prices. Using the financial sentiment dictionary significantly augments our model's capability to dissect and comprehend financial documents with remarkable precision, thereby improving the reliability and effectiveness in forecasting stock price movements based on the sentiment in the text messages.

 doi:10.20944/preprints202404.0563.v1

The final stage of our methodology, as shown in Figure 2, involved data analysis using machine learning techniques to predict stock prices based on sentiment data and historical price trends, employing algorithms such as linear regression, random forest, and neural networks. This approach aimed to establish a correlation between stock market sentiment and price movements and showcased the practical applications of machine learning models in financial analysis.

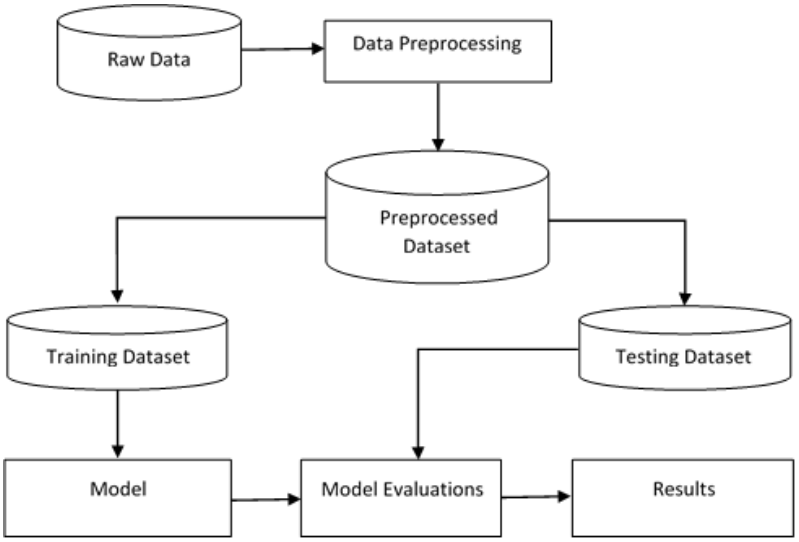

Figure 2. Overview of proposed methodology.

3. Results

An intriguing outcome of our research is the adaptation of context-based Convolutional Neural Network (CNN) models within the task of sentiment analysis. Where previous studies have generally limited their scope to a fewer set of messages, resulting in high variance and insufficient accuracy for dependable investment decisions, our approach extends it to a broader portfolio of stocks. While focusing on a few stocks can provide deep insights into individual market behaviors, extending the analysis to a larger set of messages can capture a wider array of market dynamics and reduce the risk associated with anomalous movements in any single stock.

In this phase of the work, we applied our mathematical models using a CNN, which was implemented through the TensorFlow framework. We began by setting the operational hyperparameters of the neural network as suggested by best practices[33]. Specifically, we utilized around 100 filters for dimensions, and set the dropout probability during the regularization phase at 0.5 to prevent overfitting. For the training of the neural network, we employed batch gradient descent with a batch size of 64, across 8 training epochs. Additionally, we explored the effects of various filter combinations through multiple experiments.

The best result was achieved with a combination of filters of size 7. This can be explained by the need to take into account both near and far contexts of a word without overloading the network with additional information in the form of duplicate or close in size filters. Increasing the number of filters allows to slightly influence the result, but after reaching a certain threshold, the growth of accuracy and completeness stops (See Figure 3). The best results are achieved when the probability of neuron dropout is kept around 0.5, because in this case there are enough neurons for classification, and, at the same time, regularization of the network avoids the problem of overtraining.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 April 2024 doi:10.20944/preprints202404.0563.v1

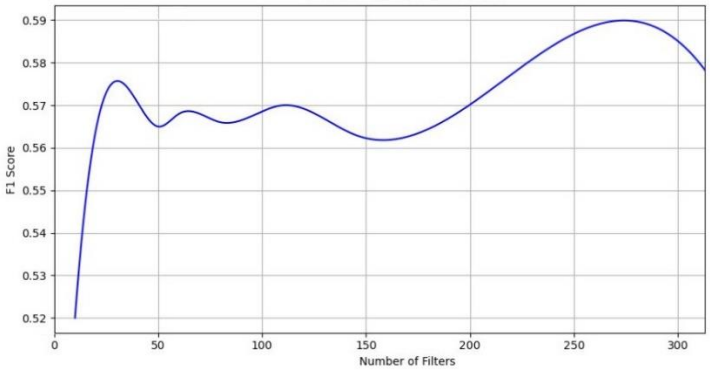

Figure 3. Impact of the number of filters on the performance of CNN.

Batch gradient descent is considered to be a faster and more stable implementation of the error back propagation method compared to stochastic gradient descent, but it tends to stop and get stuck in local minima, especially at high batch sizes, as demonstrated by the results obtained. As a compromise, smaller batch sizes can be used. Increasing the number of training epochs has a negative effect on the results, since on a small training sample the neural network becomes more sensitive to the overtraining problem.

The architecture of the convolutional neural network is important in the extraction and processing of textual data. The arrangement of the convolutional layers, coupled with filter sizes that capture varying contextual lengths, enables the network to discern subtle nuances in tone and sentiment. This flexibility is crucial for tasks such as sentiment analysis, where the emotional undertone of a text can significantly influence its interpretation. The ability of the CNN to adapt its filters to the specific demands of the dataset emphasizes the importance of an approach to neural network design, where parameters are not arbitrarily chosen but are instead intentionally selected to optimize performance. The performance on these parameters are shown on the Figures 4–6.

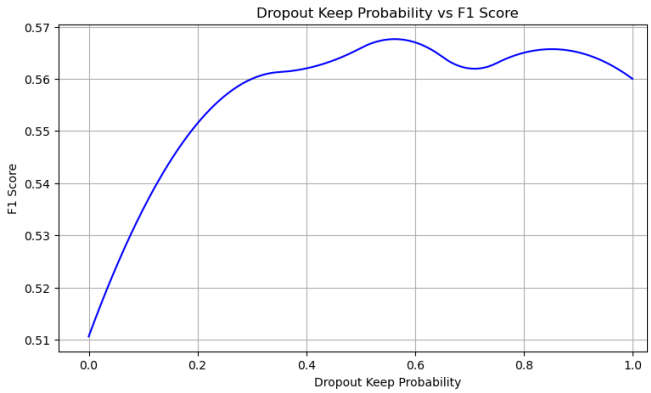

Figure 4. Impact of dropout probability on the performance of CNN.

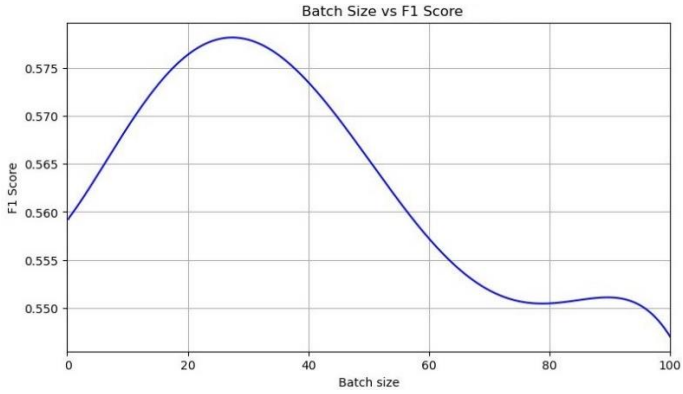

Figure 5. Impact of the batch size on the performance of CNN.

 doi:10.20944/preprints202404.0563.v1

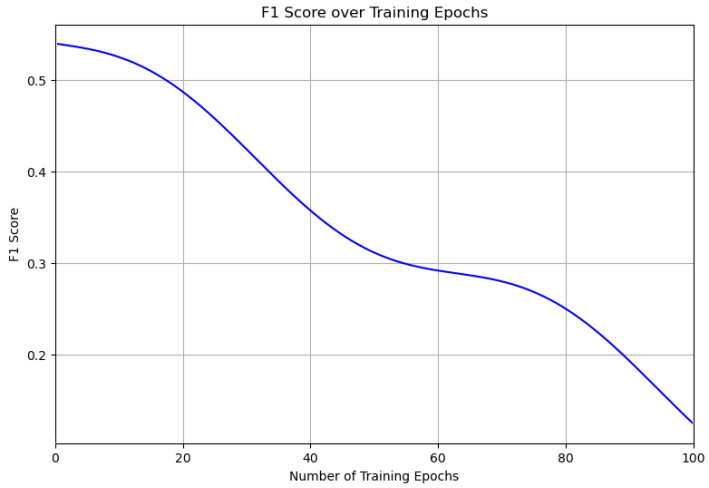

Figure 6. Impact of the number of training epochs on the performance of CNN.

Moreover, the integration of dropout as a regularization technique demonstrates a strategic balance between learning complexity and model generalizability. By randomly omitting neurons during the training phase, dropout prevents the network from becoming overly dependent on any single neuron, thus mitigating the risk of overfitting. This technique ensures that the model remains robust and capable of generalizing from the training data to unseen examples, which is essential for maintaining high levels of accuracy in real-world applications.

Our strategy involved adapting the models to each assessed dataset during a 5-fold cross-validation process(See Table 2). By incorporating tweets from the training folds with additional datasets, we enriched the adaptation process of our model, a technique that was rigorously tested using diverse seeds and parameter tuning. The culmination of these tests yielded an average prediction accuracy of over 90%, helping for the significant correlation between tweets and market behavior and validating the sufficiency of our sample size.

Table 2. Performance metrics of CNN across 5-fold cross-validation.

Cross-Validation	Accuracy	Precision	Recall	F1 Score
Fold 1	0.9941	0.9940	0.9942	0.9941
Fold 2	0.9939	0.9941	0.9938	0.9940
Fold 3	0.9942	0.9943	0.9941	0.9942
Fold 4	0.9938	0.9937	0.9940	0.9939
Fold 5	0.9940	0.9942	0.9941	0.9941
Average	0.9940	0.9941	0.9940	0.9941

It should be noted that using two convolutional neural networks in sequence yields inferior results compared to employing just one network. However, when used separately, each neural network in the ensemble demonstrates significantly better sentiment classification. This improvement is attributed to the dominance of one sentiment (positive) over others, leading to high accuracy rates by predominantly categorizing it into the most common sentiment. The neural network designed to distinguish emotional tonality from neutral tonality outperforms others, as aspects with a neutral tonality are relatively uncommon.

Summarizing the above, the best results for context extraction were achieved using filters of dimensionality 3-7 of 100 each, with a probability of neuron dropout equal to 0.5. The successful application of CNNs in this domain highlights their potential as a powerful tool for processing and analyzing textual data, offering insights that can significantly enhance our understanding of language and its emotional undertones[34].

Our model has produced Table 3, which presents our findings on stock market data. This table doesn't just list stock prices; it also includes sentiment analysis to show how people's reactions on

 doi:10.20944/preprints202404.0563.v1

social media might affect stock trends. By including this information, we can better understand the success of different prediction models. Such integration enables detailed evaluation of the predictive model's performance, and provides a more dimensional assessment, considering not just the statistical outcomes but also the psychological undercurrents that drive market behaviors.

Table 3. Stock prices correlated with social media sentiments.

	Date	Tweet	Prices	Negative	Neutral	Positive
1	20090801	Microsoft's free anti-malware beta to arrive ...	23.52	0.07173	0.066	0.807
2	20090802	Microsoft invites some of its bestest OEM bud...	23.52	0.09593	0.082	0.766
3	20090803	Outstanding insight of the ramifications of to ...	23.83	0.08674	0.081	0.831
4	20090803	Yahoo and Microsoft picked the wrong fight: I...	23.77	0.08114	0.068	0.781
5	20090802	Get Rich on Microsoft Search engine Bing http...	23.81	0.06255	0.071	0.820

After testing our classification model, we conducted data analysis using the pre-processed sentiment data and stock prices. The employed algorithms included linear regression, random forest, and Support Vector Machines (SVM). The objective was to leverage machine learning techniques to predict stock prices based on sentiment data and historical price trends.

Given that our primary focus is on supervised learning algorithms and then our dataset spans six months, we opted to allocate the initial five months of data for training purposes and the final month for testing. The figures presented clearly indicate that SVM and linear regression outperform random forest in terms of prediction accuracy. Figures 7–9 display the prediction outcomes from linear regression, random forest, and SVM, respectively.

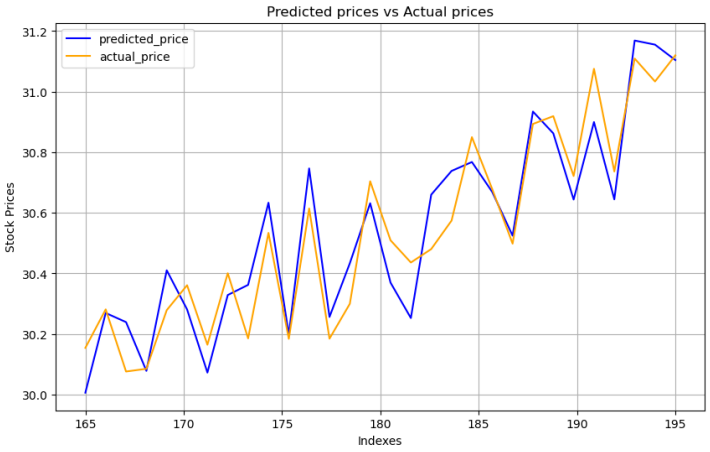

Figure 7. Prediction result of linear regression.

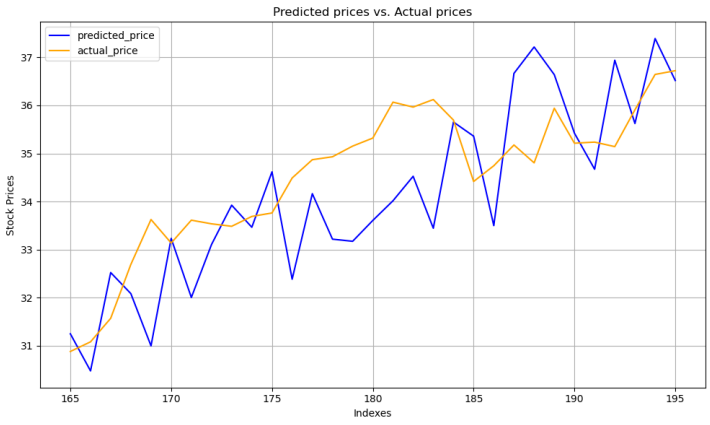

Figure 8. Prediction result of random forest

 doi:10.20944/preprints202404.0563.v1

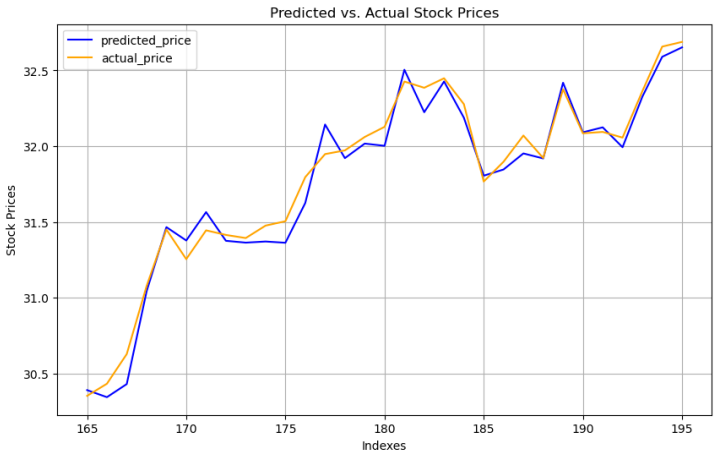

Figure 9. Prediction result of SVM

There linear regression showed satisfactory prediction capabilities but tended to mirror the prior day's actual stock prices closely. This pattern could potentially result in substantial losses in the volatile stock market, where prices can shift abruptly. On the other hand, SVM predictions demonstrated a closer alignment with actual prices, managing to capture the market's trend more accurately despite a slight prediction delay. This indicates SVM's superior adaptability to market fluctuations compared to linear regression.

To refine our models further, we implemented a feature selection method aimed at identifying and utilizing the most crucial features from a reduced size of the training set. This process involved assessing the impact of different sizes of important features on the performance of our models, especially in handling imbalanced data. By ranking the features according to their mutual information values, we were able to optimize our training data, thereby enhancing the precision and reliability of our predictions. This strategic approach to feature selection underscores our commitment to improving model performance by focusing on the most influential factors.

While numerous studies employ the accuracy of trend prediction as the benchmark for evaluating prediction models, this work adopts a different approach[35]. Considering that stock investment decisions are not solely based on the directional trend of stock movements but also on seasonal factors and other complexities, a mere qualitative assessment is insufficient. Therefore, we opt for Root Mean Square Error (RMSE) and Mean Square Error (MSE) as our evaluation metrics. These quantitative measures are widely recognized and provide a more comprehensive assessment of a prediction model's performance. The outcomes of our evaluation, based on these criteria, are detailed in Table 4.

Table 4. The MSE/RMSE result of three methods.

Methods	MSE	RMSE
Linear Regression	0.082878409	0.287886104
Random Forest	0.5054099	0.711377248
SVM	0.071215228	0.269700311

Our analysis, shown in Figure 10, highlights how well the linear regression and SVM models can predict with low errors. The SVM model, in particular, stands out for its accuracy, matching what we expected[36]. Its predictions are very close to the real market prices and it's good at following the ups and downs of the market, even if there's a slight delay in its predictions. This shows that the SVM model is quite robust and flexible, able to keep up with market changes better than the linear regression model. We've put a lot of effort into making our model as accurate as possible by focusing on the most important factors that affect market movements. By carefully choosing which features to include in our model, we've been able to make our predictions more precise.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 April 2024 doi:10.20944/preprints202404.0563.v1

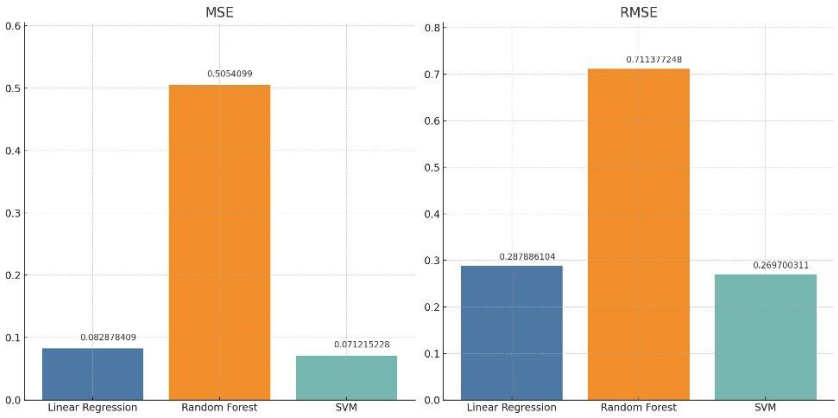

Figure 10. Evaluation of prediction using three models.

Furthermore, a deeper examination of market behaviors is detailed in Figure 11, which illustrates the relationship between Microsoft's stock price and the prevailing sentiment trends on social media platforms. The figure specifically showcases the correlation between the positive trends in public sentiment, as expressed through social media, and subsequent movements in stock prices.

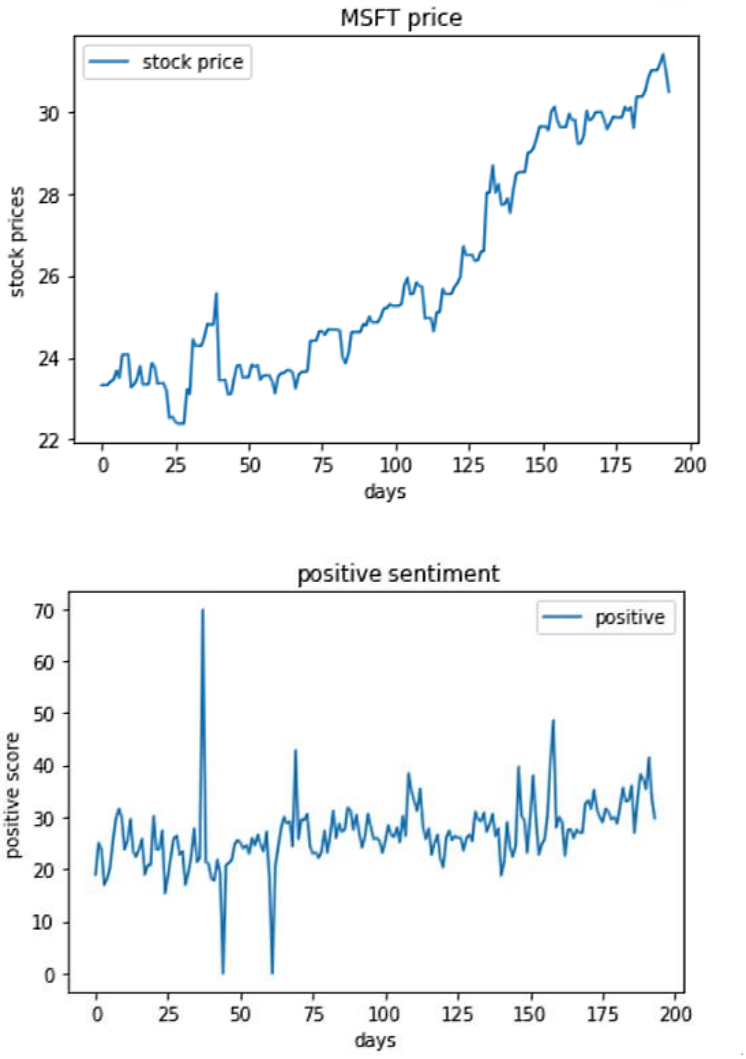

Figure 11. Correlation between Microsoft's stock price and positive sentiment trends.

As a result, we could apply our model to predict NASDAQ stock trends by analyzing sentiments from daily Twitter posts (see Table 5). With so many tweets out there, and many not related to the

 doi:10.20944/preprints202404.0563.v1

stock market, we made sure to only use tweets that mention stock hashtags. This way, we have a wide range of data that's meaningful and large enough to be statistically reliable, helping us analyze trends more accurately and keep unnecessary noise to a minimum.

Table 5. Comparative performance metrics for LR, RF, and SVM Models across stocks.

Stocks	AAPL				AMZN				MSFT			
Model	Acc.	F1-score	Precision	Recall	Acc.	F1-score	Precision	Recall	Acc.	F1-score	Precision	Recall
LR	0.720	0.766	0.735	0.800	0.700	0.795	0.789	0.801	0.737	0.708	0.684	0.736
RF	0.542	0.496	0.451	0.551	0.530	0.548	0.528	0.570	0.466	0.504	0.457	0.561
SVM	0.781	0.810	0.747	0.884	0.776	0.745	0.707	0.788	0.782	0.768	0.716	0.828

For sentiment extraction, tweets were classified into positive, neutral, or negative categories. The predictive model designed to forecast stock price fluctuations demonstrated an accuracy exceeding 75% on the test set. This result is indicative of the efficacy of our proposed strategy, which consistently outperformed alternative methods over the observed period. The models were trained using offline data, with the dataset divided into a training set, constituting 80% of the data, and a test set, comprising the remaining 20%. Each dataset entry includes feature vectors encapsulating sentiment scores and the prior day's stock price change rate. With over 20k records in the dataset, the models underwent initial training followed by evaluation against the test set.

Comparative analysis across three distinct models revealed that the SVM yielded the highest performance metrics on the test data, establishing it as the selected model for our final system implementation. This chosen model demonstrates the practical application of sentiment analysis in financial market prediction, solidifying its relevance and utility in the domain of quantitative finance.

4. Discussion

In our investigation, we encountered findings that notably diverge from previous studies. Initially, we conducted a thorough preprocessing of both Twitter and stock market data, ensuring precise alignment by date for a cohesive analysis. However, unlike some earlier research that suggested a more uniform distribution of sentiment[37], our sentiment analysis, applied after the preprocessing phase, revealed an unexpected pattern: the sentiment scores across stocks formed a left-skewed distribution, hinting at a subtler sentiment polarity than previously thought. Contrary to the common narrative, our data suggested that extremely negative sentiments had a more pronounced impact on stock price declines than previously reported, while significantly positive sentiments were closely tied to stock price increases. This observation directly challenges the notion that public sentiment has a stochastic effect on stock prices, underscoring the hidden influence of extreme sentiments on market dynamics. Further diverging from past findings[38-39], we posited that Twitter influencers might have significant sway over market movements, a hypothesis not extensively explored in prior work[40]. We explored sentiment fluctuations over time, adopting an hourly classification to uncover potential impacts on stock performance. This detailed approach, focusing on temporal sentiment variations, proposes a novel way for sentiment analysis in finance, setting the stage for the deeper incorporation of time series models in future investigations[41].

Through our investigation, we discerned that pre-training context-based CNN models improve the accuracy of classification. However, we identified that amassing larger sets of tweets does not invariably enhance predictive performance, particularly for models trained from scratch on tweets. Moreover, the employment of the short strategy, bolstered by spread return calculations, not always mirrors the complex nature of market trading. Our findings suggest an optimal tweet sample size of 40,000 or fewer, beyond which the model's adaptation becomes less effective. This decline in adaptation efficacy may stem from the over-adjustment of model weights during back-propagation, potentially undermining the intrinsic semantic and syntactic knowledge previously encoded within the model's layers.

 doi:10.20944/preprints202404.0563.v1

The model, while is a significant step toward understanding stock price movements through sentiment analysis, encounters several limitations that currently impede its real-world application. One of the prominent limitations is the tendency of the model to favor positive sentiment terms over negative ones. This bias may stem from the overall upward trend in stock prices observed within the dataset's time frame. Consequently, the model's predictive accuracy is skewed, reflecting the prevailing positive market conditions rather than a balanced sentiment assessment.

Additionally, the model's training on multiple stocks and the potential cross-sentiment influence among users could introduce systematic bias, affecting the generalizability of the predictions. The impact of breaking news on subsequent days could affect predictions, which suggests a need for further study. The reliance on bigram frequency as the primary feature further constrains the model's capacity to encapsulate complex sentiment expressions, as it overlooks the potential richness of sentiment conveyed in longer N-gram terms. The choice of the lagging parameter, while based on comparative performance metrics, lacks a robust selection algorithm that could potentially enhance the model's predictive capability. While our model has shown improvement in performance, the error margin remains too high for practical use. We're considering the application of a non-linear model to the entire feature set as a possible improvement.

Future iterations of this model would benefit from the application of more sophisticated algorithms for parameter selection, likely leading to improved performance outcomes. Another challenge is the integration of multiple software tools, which has led to complications in achieving a seamless combination. This technical hurdle has notably restricted our ability to access real-time data and accurately predict stock price movements as they unfold. These limitations outline the areas for improvement and future research directions. Advancements in feature selection, algorithmic development, and software integration are critical next steps to refine the model and achieve a level of accuracy that is applicable to real-world financial markets.

5. Conclusions

In conclusion, the study highlights the viability of employing SVM and linear regression as classifiers following the extraction of features from context-based CNN models. These classifiers emerge as judicious selections, complementing the intricate modeling process and reinforcing the veracity of sentiment analysis as a pivotal tool in stock price prediction. We've taken a distinctive path by forgoing traditional external dictionaries in favor of direct textual analysis, a decision driven by the limited availability of such dictionaries across various languages. This approach allowed us to zero in on sentiment-linked phrases crucial within specific scenarios, thereby crafting a neural network model tailored to contextually significant sentiment analysis. The method effectively narrows the knowledge gap, ensuring our sentiment analysis is not only relevant to specific contexts but also mirrors stock market trends with high fidelity. Our exploration incorporates diverse strategies to not merely refine our models' accuracy but also their comprehensibility. By harnessing advanced mathematical modeling techniques, we have strived to ensure our findings are crystal clear.

Our study's insights into sentiment polarity and the influential role of Twitter users offer a fresh perspective that contradicts some established beliefs, suggesting that the relationship between public sentiment and stock market trends is more complex than previously understood. Future work will focus on enhancing feature selection, refining algorithms, and better software integration to improve the model’s accuracy for real-world financial applications. The transition from theory to practice in our methodology paves the way for a more realistic and actionable set of guidelines for market participants. As such, our work serves as a viable template for developing advanced predictive models that can be directly applied to stock market investing, transcending the role of mere academic reference to become a practical tool in the arsenal of investors.

Author Contributions: Conceptualization, N.S. and A.K.; methodology, A.K. and N.S.; validation, N.S. and R.K.; formal analysis, A.K.; investigation, N.S., A.K. and R.K.; data curation, N.S. and R.K.; writing—original draft preparation, N.S. and A.K.; writing—review and editing, N.S., R.K., A.K.; visualization, N.S., R.K.; supervision, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The used data are publicly available at https://huggingface.co/datasets/carblacac/twitter-sentiment-analysis (accessed on 25 March 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gao, Y.; Zhao, C.; Sun, B.; Zhao, W. Effects of Investor Sentiment on Stock Volatility: New Evidences from Multi-Source Data in China's Green Stock Markets. *Financ. Innov.* 2022, *8*(1), 77, 1–30. https://doi.org/10.1186/s40854-022-00381-2.
2. Nakhli, M.S.; Dhaoui, A.; Chevallier, J. Bootstrap rolling-window Granger causality dynamics between momentum and sentiment: Implications for investors. Ann. Financ. 2022, 18, 267–283. https://doi.org/10.1007/s10436-021-00399-z.
3. Pagolu, V.S.; Reddy, K.N.; Panda, G.; Majhi, B. Sentiment Analysis of Twitter Data for Predicting Stock Market Movements. In Proceedings of the 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), Paralakhemundi, India, 2016; pp. 1345–1350. https://doi.org/10.1109/SCOPES.2016.7955659.
4. Ranco, G.; Aleksovski, D.; Caldarelli, G.; Grčar, M.; Mozetič, I. The Effects of Twitter Sentiment on Stock Price Returns. PLOS ONE, 2015, 10, 1–21. https://doi.org/10.1371/journal.pone.0138441.
5. Chaudhry, H.N.; Javed, Y.; Kulsoom, F.; Mehmood, Z.; Khan, Z.I.; Shoaib, U.; Janjua, S.H. Sentiment Analysis of before and after Elections: Twitter Data of U.S. Election 2020. Electronics 2021, 10, 2082. https://doi.org/10.3390/electronics10172082.
6. Xiao, Q.; Ihnaini, B. Stock Trend Prediction Using Sentiment Analysis. PeerJ Computer Science 2023, 9, e1293, 1-18. https://doi.org/10.7717/peerj-cs.1293.
7. Barreto, S.; Moura, R.; Carvalho, J.; Paes, A.; Plastino, A. Sentiment Analysis in Tweets: An Assessment Study from Classical to Modern Word Representation Models. *Data Min. Knowl. Disc.* 2022, *37*, 1–63. https://doi.org/10.1007/s10618-022-00853-0.
8. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2019; pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.
9. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. CoRR, 2019. http://arxiv.org/abs/1907.11692.
10. Nguyen, D.Q.; Vu, T.; Nguyen, A.T. BERTweet: A Pre-trained Language Model for English Tweets. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, October 2020; pp. 9–14. https://doi.org/10.18653/v1/2020.emnlp-demos.2.
11. Pathak, A.R.; Agarwal, B.; Pandey, M.; Rautaray, S. Application of Deep Learning Approaches for Sentiment Analysis. In Deep Learning-Based Approaches for Sentiment Analysis; Agarwal, B., Nayak, R., Mittal, N., Patnaik, S., Eds.; Springer: Singapore, 2020; pp. 1–31. https://doi.org/10.1007/978-981-15-1216-2_1.
12. Salton, G.; Wong, A.; Yang, C.S. A Vector Space Model for Automatic Indexing. *Commun. ACM* 1975, *18*, 613–620. https://doi.org/10.1145/361219.361220.
13. Turney, P.; Pantel, P. From Frequency to Meaning: Vector Space Models of Semantics. *J. Artif. Intell. Res.* 2010, *37*. https://doi.org/10.1613/jair.2934.
14. Manning, C.D.; Raghavan, P.; Schütze, H. Introduction to Information Retrieval. Cambridge University Press: Cambridge, England, 2008; 482 pp.
15. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and Their Compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems, NIPS'13, 2013; pp. 3111–3119.
16. Agrawal, A.; An, A.; Papagelis, M. Learning Emotion-Enriched Word Representations. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, 2018; Association for Computational Linguistics: pp. 950–961.
17. Felbo, B.; Mislove, A.; Søgaard, A.; Rahwan, I.; Lehmann, S. Using Millions of Emoji Occurrences to Learn Any-Domain Representations for Detecting Sentiment, Emotion, and Sarcasm. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 2017; Association for Computational Linguistics: pp. 1615–1625. https://doi.org/10.18653/v1/D17-1169.

 doi:10.20944/preprints202404.0563.v1

18. Xu, P.; Madotto, A.; Wu, C.S.; Park, J.H.; Fung, P. Emo2Vec: Learning Generalized Emotion Representation by Multi-Task Training. In Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, Brussels, Belgium, 2018; Association for Computational Linguistics: pp. 292–298. https://doi.org/10.18653/v1/W18-6243.
19. Tang, D.; Wei, F.; Yang, N.; Zhou, M.; Liu, T. Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Baltimore, 2014, 1, 1555–1565. https://doi.org/10.3115/v1/P14-1146.
20. Kang, Q.; Chen, E.J.; Li, Z.-C.; Luo, H.-B.; Liu, Y. Attention-based LSTM Predictive Model for the Attitude and Position of Shield Machine in Tunneling. *Underground Space* 2023, *13*, 335–350. https://doi.org/10.1016/j.undsp.2023.05.006.
21. García-Medina, A.; Sandoval, L.; Urrutia Bañuelos, E.; Martínez-Argüello, A.M. Correlations and Flow of Information between the New York Times and Stock Markets. *Physica A: Stat. Mech.* 2018, 502, 403–415. https://doi.org/10.1016/j.physa.2018.02.154.
22. Taboada, M.; Brooke, J.; Tofiloski, M.; Voll, K.; Stede, M. Lexicon-Based Methods for Sentiment Analysis. *Comput. Linguist.* 2011, 37, 267–307. https://doi.org/10.1162/COLI_a_00049.
23. Ojeda-Hernández, M.; López-Rodríguez, D.; Mora, Á. Lexicon-Based Sentiment Analysis in Texts Using Formal Concept Analysis. *Int. J. Approx. Reason.* 2023, *155*, 104–112. https://doi.org/10.1016/j.ijar.2023.02.001.
24. Rice, D.R.; Zorn, C. Corpus-Based Dictionaries for Sentiment Analysis of Specialized Vocabularies. *Political Science Research and Methods* 2021, 9(1), 20–35. https://doi.org/10.1017/psrm.2019.10.
25. Feng, J.; Gong, C.; Li, X.; Lau, R.Y.K. Automatic Approach of Sentiment Lexicon Generation for Mobile Shopping Reviews. *Wireless Commun. Mob. Comput.* 2018, 9839432, 1-13. https://doi.org/10.1155/2018/9839432.
26. Velikovich, L.; Blair-Goldensohn, S.; Hannan, K.; McDonald, R. The Viability of Web-Derived Polarity Lexicons. In Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, California, June 2010, 777–785. https://aclanthology.org/N10-1119.
27. Hamilton, W.L.; Clark, K.; Leskovec, J.; Jurafsky, D. Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora. *Proc. Conf. Empir. Methods Nat. Lang. Process. 2016*, 2016, 595–605. https://doi.org/10.18653/v1/D16-1057.
28. Yang, Y.; Eisenstein, J. Overcoming Language Variation in Sentiment Analysis with Social Attention. *Trans. Assoc. Comput. Linguist.* 2017, *5*, 295–307. https://doi.org/10.1162/tacl_a_00062.
29. Naji, I. TSATC: Twitter Sentiment Analysis Training Corpus. In Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL), 2012. http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-2012-09-22.
30. Yang, J.; Leskovec, J. Patterns of Temporal Variation in Online Media. In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM '11), Hong Kong, China, 2011; Association for Computing Machinery: New York, NY, USA, pp. 177–186. https://doi.org/10.1145/1935826.1935863.
31. Zhang, F.; Ding, Y.; Liao, Y. Financial Data Collection Based on Big Data Intelligent Processing. *Int. J. Inform. Technol. Syst. Approach 2023*, 16(3), 1–13. http://doi.org/10.4018/IJITSA.320514.
32. Loughran, T.; McDonald, B. When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance 2011*, 66, 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x.
33. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *J. Big Data 2021*, 8(1), 53. https://doi.org/10.1186/s40537-021-00444-8.
34. Krichen, M. Convolutional Neural Networks: A Survey. Computers 2023, 12, 151. https://doi.org/10.3390/computers12080151.
35. Susanto, H.; Sari, A.; Leu, F.-Y. Innovative Business Process Reengineering Adoption: Framework of Big Data Sentiment, Improving Customers' Service Level Agreement. *Big Data Cogn. Comput. 2022*, 6, 151. https://doi.org/10.3390/bdcc6040151.
36. Sonkavde, G.; Dharrao, D.S.; Bongale, A.M.; Deokate, S.T.; Doreswamy, D.; Bhat, S.K. Forecasting Stock Market Prices Using Machine Learning and Deep Learning Models: A Systematic Review, Performance Analysis and Discussion of Implications. Int. J. Financial Stud. 2023, 11, 94. https://doi.org/10.3390/ijfs11030094.
37. Singh, L.G.; Mitra, A.; Singh, S.R. Sentiment Analysis of Tweets Using Heterogeneous Multi-layer Network Representation and Embedding. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 2020; Association for Computational Linguistics, pp. 8932–8946. https://doi.org/10.18653/v1/2020.emnlp-main.718.
38. Sailunaz, K.; Alhajj, R. Emotion and Sentiment Analysis from Twitter Text. *Journal of Computational Science 2019*, 36, 101003. https://doi.org/10.1016/j.jocs.2019.05.009.

 doi:10.20944/preprints202404.0563.v1

39. Jahanbin, K.; Chahooki, M.A.Z. Aspect-Based Sentiment Analysis of Twitter Influencers to Predict the Trend of Cryptocurrencies Based on Hybrid Deep Transfer Learning Models. *IEEE Access 2023*, 11, 121656-121670. https://doi.org/10.1109/ACCESS.2023.3327060.
40. Kalashnikov, R.; Kartbayev, A. Assessment of the Impact of Big Data Analysis on Decision-Making in Stock Trading Processes. In Proceedings of the 13th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare, 2024; *Procedia Computer Science, Volume 231*, pp. 786-791. https://doi.org/10.1016/j.procs.2023.12.137.
41. Saadatmand, F.; Zare Chahoki, M.A. Time Series Analysis by Bi-GRU for Forecasting Bitcoin Trends Based on Sentiment Analysis. In Proceedings of the 2023 13th International Conference on Computer and Knowledge Engineering (ICCKE), Mashhad, Iran, 2023; pp. 323–328. https://doi.org/10.1109/ICCKE60553.2023.10326259.