User2Vec: A Novel Representation for the Information of the Social Networks for Stock Market Prediction Using Convolutional and Recurrent Neural Networks

Pegah Eslamieh, Mehdi Shajari and Ahmad Nickabadi

Abstract: Predicting stock market trends is an intriguing and complex problem, which has drawn considerable attention from the research community. In recent years, researchers have employed machine learning techniques to develop prediction models by using numerical market data and textual messages on social networks as their primary sources of information. In this article, we propose User2Vec, a novel approach to improve stock market prediction accuracy, which contributes to more informed investment decision making. User2Vec is a unique method that recognizes the unequal impact of different user opinions on specific stocks, and it assigns weights to these opinions based on the accuracy of their associated social metrics. The User2Vec model begins by encoding each message as a vector. These vectors are then fed into a convolutional neural network (CNN) to generate an aggregated feature vector. Following this, a stacked bi-directional long short-term memory (LSTM) model provides the final representation of the input data over a period. LSTM-based models have shown promising results by effectively capturing the temporal patterns in time series market data. Finally, the output is fed into a classifier that predicts the trend of the target stock price for the next day. In contrast to previous attempts, User2Vec considers not only the sentiment of the messages, but also the social information associated with the users and the text content of the messages. It has been empirically proven that this inclusion provides valuable information for predicting stock direction, thereby significantly enhancing prediction accuracy. The proposed model was rigorously evaluated using various combinations of market data, encoded messages, and social features. The empirical studies conducted on the Dow Jones 30 stock market showed the model’s superiority over existing state-of-the-art models. The findings of these experiments reveal that including social information about users and their tweets, in addition to the sentiment and textual content of their messages, significantly improves the accuracy of stock market prediction.

Keywords: stock market prediction; social network analysis; deep learning; user behavior modeling; financial market emotion analysis

1. Introduction

The stock market is one of the most critical financial markets in every country’s economy. Predicting changes in this market holds great value for both small and large investors, as well as financial analysts, since even minor fluctuations can cause significant profit or financial loss. However, the increasing interest of investors in short-term stock market investments calls for the replacement of traditional prediction methods, which were primarily focused on long-term investments, with faster and more cost-effective approaches. As prediction methods continue to advance, the information sources utilized by these models have also undergone changes over the past few decades. The widespread use of social media platforms and the dissemination of people’s thoughts, opinions, and experiences...
within these networks have motivated researchers to incorporate this valuable asset into their stock market prediction models. Deep neural networks have further facilitated the introduction of several innovative techniques for predicting share prices using these networks. These methods can be examined based on their prediction objectives, information sources or features, and the techniques employed.

Stock market prediction models are developed with different goals. The primary focus of stock market prediction has been on forecasting the price of a share for a specific future period \cite{1,2}. The price of a share is a numerical value, and its variation over time is often treated as a time series in various studies \cite{3,4}. The prediction timeframe can range from minutes to months. For instance, Di Persio et al. \cite{5} developed a model utilizing a multi-layer RNN, an LSTM, and a gated recurrent unit (GRU) to forecast Google stock price movements. Similarly, Althelaya et al. \cite{6} focused on evaluating and comparing LSTM deep learning architectures for the short- and long-term prediction of financial time series.

Another common aim of stock market prediction methods is to determine the trend of stock prices rather than predicting exact price values. This approach treats prediction as a binary classification task, thus classifying whether the price will increase or decrease \cite{7,8}. Some researchers have also explored the trends in market changes or market indexes, such as the Dow Jones index \cite{9,10}. With the growing use of social networks and their impact on financial markets, the sentiment analysis of messages \cite{11} and scoring the members \cite{12} of these networks have attracted the attention of researchers. In this paper, our focus is on predicting the daily price trend of certain shares in the DOW30 index.

Machine learning methods for predicting stock markets use different information sources or sets of features. One of the most crucial features shared among nearly all models is market data, which includes the open/high/low/close (OHLC) prices of shares during various time periods. Data from other markets, such as oil prices, are also used in many models \cite{13}. While classical methods were primarily limited to these numerical data, it has been shown that news and public opinions expressed in social network messages can provide valuable information for predicting market behavior. As a result, natural language processing has been employed to extract sentiment from textual data as news, blog posts, micro blogs, and messages \cite{14}. Social information about users, such as the number of followers or the accuracy of their predictions, is also utilized as a feature to weigh their opinions. Economic data, such as knowledge graphs illustrating relationships among companies and industries, macroeconomic data for a country, and analytics data from companies, can serve as potential features for share prediction. In this paper, we propose a comprehensive model that extracts and incorporates both market data and social media information for prediction.

Over the past few decades, a wide range of methods and techniques has been utilized for stock market prediction. Traditionally, human traders have relied on fundamental and technical analysis. Fundamental analysis involves estimating the intrinsic value of a stock, while technical analysis focuses on identifying recurring patterns in OHLC candlestick charts \cite{15}. However, automatic trading algorithms have adopted statistical and machine learning approaches for stock market prediction. Statistical methods treat the stock price over time as a time series and make assumptions of linearity, stationarity, and normality in the input data. The autoregressive integrated moving average (ARIMA) \cite{16,17} and generalized autoregressive conditional heteroscedasticity (GARCH) \cite{18,19} are well-known examples in this category. With the advancement of machine learning (ML) models, they have gained significant attention in stock market prediction. ML techniques can be categorized into shallow and deep learning methods. Shallow models, such as naive Bayes \cite{20}, logistic regression \cite{21}, support vector machines \cite{22,23}, decision trees \cite{2,24}, random forests \cite{25}, and early neural networks such as multilayer perceptron \cite{26–28}, have been extensively examined in this domain. However, following the emergence of deep learning methods, stock market prediction, as with many other data analysis tasks, has been dominated by deep neural networks \cite{29}. Section 2 explores some models from this group.
LSTM is a type of deep network used for various tasks, particularly in handling sequential data [30–32]. LSTM networks are well-suited for time-series data such as stock market prediction. In this context, the sequential nature of past stock prices forms a time series, where the order of the data is crucial. The ability of the LSTM network to keep and forget information over time enables it to capture the temporal dependencies in the data [33] effectively, which is essential for accurately predicting future stock prices. Compared to other advanced models such as transformers, LSTMs have a distinct advantage. Transformers, although powerful, are not inherently designed to handle sequential time-series data, as they process the entire sequence simultaneously [34]. Transformers have gained recognition for their exceptional performance in natural language processing tasks where the parallel processing of complete sentences or documents can be beneficial. However, for time-series data such as stock prices, this approach may not always be optimal [35].

LSTM models process data sequentially, and their stateful nature makes them well-suited for problems where data order matters, as with stock prices. The LSTM architecture is specifically designed to keep past information using a mechanism called a ‘gate’. These gates control the flow of information into and out of the memory cell, thereby determining which information should be kept or discarded [36]. This capability allows LSTM models to excel at capturing long-term dependencies in sequence data, which is crucial for stock market prediction. Furthermore, the sequential processing of LSTM models can be more computationally efficient when handling long sequences [37]. Unlike the transformer model, where each element in the sequence needs to be connected to every other element, LSTM models can process the elements individually, thereby making them a more practical choice for large datasets or long sequences that are commonly encountered in stock market prediction [8]. While transformers are complex models with various parameters, LSTM models are relatively simpler and more interpretable. In practical terms, this means that understanding the predictions made by an LSTM model and diagnosing and resolving any issues that may arise during model training or inference can be easier compared to transformers [34].

CNNs, which were originally designed for image processing tasks [38,39], have also found use in time-series data prediction such as stock market forecasting. A key feature of CNNs is their ability to extract local and global features [40], which can be useful in stock prediction. For instance, local features could represent short-term fluctuations in the stock market, while global features could capture long-term trends. Combining CNNs with LSTM models allows the resulting model to both abstract features effectively (through CNNs) and understand their sequential nature. Integrating LSTM models and CNNs into a hybrid model can leverage the strengths of both architectures. In such a model, the CNN layer can extract salient features from the input data, thereby reducing its dimensionality and highlighting crucial patterns. These abstracted features are then fed into the LSTM layer, which considers the sequential dependencies among these features to make final predictions. By employing this hybrid approach, the model benefits from the ability of CNNs to identify local and global trends and the capacity of LSTM models to understand temporal dynamics, thus leading to potentially more accurate and robust predictions. Considering these considerations, the LSTM model’s ability to handle sequential data gives it an edge over the transformer model for specific tasks such as stock market prediction [41].

In this paper, we propose a general method for predicting the close price’s trend (up/down) of a stock in the next day based on the market and social networks data of the current day. In this model, a novel representation, known as User2Vec, was devised for the opinion of each investor about a stock. User2Vec is a real-valued vector composed of the investor’s message in a social network in an embedded format, the sentiment of the message, the investor’s social and prediction scores, and the market data. Despite most of the existing approaches in which the predictions of different users about a specific stock are assumed to be of the same value, our method weights these opinions in two different ways: (1) the social information is used to estimate how influential the user is, and (2) the user’s predictions in the past are used to estimate the accuracy of his predictions. In the following,
a 1D-CNN stacked bidirectional LSTM is proposed to aggregate all opinions and predict the future of the stock. The proposed model was applied to several stocks in the Dow Jones market, and the results were analyzed. Through a thorough analysis of the results, we evaluated the effectiveness of our approach in enhancing the prediction of stock trends. The experimental findings show promising outcomes, thus indicating the potential value of incorporating social network data and User2Vec representation in stock market forecasting.

The main contributions of this research are as follows:

- We presented a hybrid model that leverages diverse sources of information for stock market prediction.
- We proposed a new representation called User2Vec, which captures each user’s tweet about a specific share.
- We employed two user scoring mechanisms based on the accuracy of their predictions and their credibility within the social network.
- We used 1D-CNN stacked recursive neural networks to forecast the market direction for the next day.

The rest of the paper is organized as follows: Section 2 provides a brief review of related works. The proposed model is presented in Section 3. Section 4 reports and examines the experimental results. Finally, Section 5 concludes the paper.

2. Literature Review

Each stock market prediction model comprises two primary components: the prediction method, and the features used by the model. While some researchers aim to enhance prediction accuracy by employing more advanced techniques, others focus on obtaining informative feature sets from various information sources. There are studies that report advancements in both aspects. In the following, we will first explore the machine learning methods that are commonly employed for stock market prediction and then review the features used in these models.

2.1. Prediction Methods

Regarding the first category of related works that focus on improving the prediction method, there is a wide range of algorithms and tools available for stock market prediction, such as neural networks, deep learning, support vector machines (SVMs), and random forests.

Machine learning methods have predominantly been used for technical analysis in this field, with various studies comparing various types of algorithms. Ensemble approaches such as random forests and AdaBoost, as well as single classifier models such as neural networks and logistic regression, were compared using data from 5767 businesses [42]. Sharma et al. [43] proposed using LSboost to aggregate the predictions of an ensemble of trees in a random forest (referred to as LS-RF). Each prediction model defines a set of technical indicators as inputs. The performance of the suggested model was compared to that of the well-known support vector regression model. Picasso et al. [44] incorporated technical and fundamental analysis parameters to evaluate the performance of several machine learning methods, including SVM and random forest. Various other supervised approaches, such as support vector regression (SVR) [45], multiple linear regression (LMLR) [46], and the j48 algorithm [47], have been investigated within the field of stock market prediction.

Artificial neural networks (ANN), based on several studies, have emerged as popular tools for financial prediction [48–50]. Most of these studies have utilized historical market data as input features. Among ANNs, the multilayer perceptron (MLP) network is widely employed for stock forecasts. MLP is a feed-forward network comprising one or more hidden layers: an input layer, and an output layer. Each layer incorporates non-linear learning capabilities. Previous studies [26–28] have proposed MLP networks for stock market prediction tasks. Deep ANNs are also extensively used in this field. To predict NASDAQ prices, ref. [51] tested ANNs with various structures using historical prices on
four- and nine-day timeframes. Their findings indicated that deep ANNs outperformed shallow networks. Arévalo et al. [52] applied a deep ANN with five hidden layers to forecast Apple’s stock in the NASDAQ exchange, and they achieved approximately 65% directional accuracy. Chong et al. [8] explored different data representation methods, including auto-encoder, RBM, and PCA, using raw data with 380 variables. These representations were employed as input for a deep ANN in stock prediction tasks. The results indicated no significant superiority of one method over the others. Hoseinzade et al. [53] employed combinations of historical prices, technical indicators, and macroeconomic data as features. They used a CNN model to train the model in 3D spaces. The proposed model was compared to a PCA+ANN technique, and the results showed the CNN outperformed the other methods. Gao et al. [54] and Wang et al. [55] integrated attention layers with CNNs to forecast the following day’s index price based on past data. The findings suggested that the attention-based approach yielded the best results among the tested models.

Recurrent neural networks (RNNs) incorporate an internal memory, thereby enabling them to capture historical information and generate predictions [56,57]. Among RNNs, LSTM is a widely used type that has also been applied to stock market prediction. Nelson et al. [58] fed technical indicators into an LSTM to forecast price trends in the Brazilian stock exchange. The results showed the superior performance of LSTM compared to MLP. In another work, [1] introduced a recursive network, the Echo State Network (ESN), to predict S&P 500 stocks. They used various stock market features, including price, volume, and the moving average. The ESN was applied to 50 stocks and achieved an error rate of 0.0027. Ding and Qin [59] proposed an LSTM-based network with multiple inputs and outputs—specifically, the opening price, lowest price, and highest price of a stock. Their investigations revealed that the suggested model outperformed the LSTM network model and other deep recurrent neural networks in predicting multiple values simultaneously, with a prediction accuracy exceeding 95%. Jin et al. [60] incorporated investors’ sentiment into stock prediction by utilizing empirical modal decomposition (EMD) to fail the complex sequence of stock prices. They also employed an LSTM network with attention mechanisms to focus on the most relevant data. The revised LSTM model not only improved prediction accuracy, but also reduced time delay according to the study’s findings. Liu et al. [61] proposed a two-component multi-element hierarchical attention capsule network. The first component, multi-element hierarchical attention, assigned weights to valuable information from various news and social media sources. The capsule network component captured additional context information from events. Their model enhanced prediction accuracy by quantifying the diverse influences of events.

2.2. Feature-Based Methods

Market data, which encompasses the open/high/low/close (OHLC) prices of a share over a specific period, stands as the most prevalent feature employed in nearly all prediction algorithms in this field [6,62–64]. The time of measurement (ranging from seconds to months) and the number of measurements used as inputs in the model may differ across various models.

In recent times, social networks have significantly influenced various aspects of human life, including financial markets. Social networks have a notable impact on financial markets through user interactions, opinion sharing, engaging in discussions, and following trusted individuals. Social trading is a specific form of this phenomenon, where investors observe and replicate the strategies of experts or peer traders. Within common social networks, two vital sources of information are the messages posted by users and the social information related to users themselves, such as following relationships. These aspects are further explored below. Concerning social network textual messages, sentiment analysis is a prevalent tool used to extract users’ opinions about shares, with the aim to classify the sentiment as positive, negative, or neutral [14]. Notably, the study conducted by Nelson et al. [65] represents one of the earliest attempts to forecast stock fluctuations using Twitter data.
To accurately assess stock market sentiment, the researchers evaluated a random subsample of tweets over six months and subsequently determined the correlation between this data and future stock market indicators. Baker et al. [66,67] developed a sentiment index that captures changes in investors’ sentiments. They showed that fluctuations in this index impacted investors and stimulated changes in the overall stock market. Gilbert et al. [68] suggested that an individual’s emotional state influences their decision making and confirmed that sentiment inferred from web content contained information that could forecast stock prices. While some studies have shown a correlation between emotional trends in internet comments and stock market movements, few have attempted to predict stock prices using sentiment analysis. For instance, Guo et al. [69] proposed a technique based on the hot optimization route, which examined the relationship between user mood and the stock market by analyzing user review data from a stock review website. Zhou et al. [70] achieved a stock market prediction accuracy of 64.15% using the SVM-ES model, wherein they incorporated social sentiments such as contempt, pleasure, melancholy, and fear. Picasso et al. [44] employed data science and machine learning tools to combine technical and fundamental assessments. The result was a predictive model capable of forecasting the trajectory of a portfolio comprising the twenty most-capitalized enterprises in the NASDAQ100 index. Bouktif et al. [71] employed improved sentiment analysis to assess the predictability of stock market directions. They delved deeper into stocks by examining various factors such as historical stock price, sentiment polarity, subjectivity, N-grams, custom text-based features, and feature delays. By employing advanced causality analysis, algorithmic feature selection, and machine learning techniques, including regularized model stacking, they collected and evaluated data from 10 major NASDAQ shares across diverse stock domains. Their method achieved a 60 percent accuracy rate, which surpassed existing sentiment-based stock market prediction algorithms, including deep learning. Alhamzeh et al. [72] analyzed innovative data sources, specifically StockTwits paired with financial news, and tackled the problem as a binary classification task. They adopted a hybrid approach that combines sentiment and event-based features. The findings indicated that StockTwits data outperformed price data in predicting the closing prices of eight NASDAQ100 companies. Another valuable information source from social networks is user-related data, including the user’s influence within the network and the accuracy of their predictions. Kamkarhaghighi et al. [12] examined the relationship between a Twitter user’s influential power in stock market prediction and their social network information, including details about their followers. They identified several active users in the stock exchange as valuable users and calculated a score for the accuracy of each user’s predictions. By setting a threshold to distinguish valuable and non-valuable users, they trained and reported the accuracy of a naive Bayes model using attributes such as the number of followers and the number of related followers of users. Ultimately, they concluded that users’ profile information could provide insights into their influence on the stock market. Bujari et al. [73] explored the relationship between various social features, including the number of each user’s followers and the volume of tweets related to each stock, in relation to that stock’s market data. The results revealed that predictive features for each stock differ, and there is no general model that applies to all stocks.

3. Materials and Methods

Figure 1 outlines the overall structure of our proposed model. The proposed model comprises three steps: (1) data collection, (2) feature extraction, and (3) prediction. Initially, input data is gathered from various sources, including text messages posted on social networks, social information related to the users, and market data containing the (open, high, low, close, Adj close, volume) prices of the target share at each time step. Twitter is a preferred data source for large-scale data analytics projects because of its appeal across the 5 Vs: Volume, Velocity, Variety, Value, and Veracity. It offers a vast volume of data generated daily, thus showcasing the rapid velocity of real-time data generation and sharing. Twitter shows variety by providing diverse data types and supporting multiple languages. The
data obtained from Twitter holds value because of its ability to provide timely insights into user behavior and trends. However, it is important to consider the veracity of the data, as its reliability and accuracy can vary because of the diverse user base. Yahoo Finance is utilized as another source for extracting daily market data. The second stage of the model focuses on extracting valuable features from the raw data obtained in the previous step. These features are represented by a numerical vector known as User2Vec, which encodes user-specific information into a vector format. Figure 2 illustrates the five distinct features comprising the User2Vec. Various components are proposed to calculate the values of these features based on the gathered data. The social network analysis module provides social information about the message authors, such as the number of followers and friends, thereby indicating their importance within the network. The text messages of users undergo preprocessing and are then encoded using a word-embedding network. Subsequently, they are processed by a sentiment analyzer that generates embedded text and sentiment features. The financial data is used to determine the actual label representing the market’s direction. The user’s score is computed based on the number of accurate and inaccurate predictions they have made. Finally, the market data is encoded into a set of features. Each message posted on the monitored social network generates a corresponding User2Vec vector. These vectors encapsulate the user’s opinion about the target share, their score, and their social information, along with the market data from the current day. Deep neural networks of various types are employed to integrate this information and predict the direction of the share for the following day. The subsequent sections provide further details on the different components of the model.

Figure 1. Overview of the proposed User2Vec model.
3.1. Data Collection

Subsequent sections provide details on data collection and utilized tools. Each set of features in our proposed model was obtained from a specific source. While there may be additional fields in the data collected at each stage, only those included in the User2Vec feature set were used in this research. The datasets generated and analyzed during the current study are available in [74].

3.1.1. Yahoo Finance App

Many free websites and software platforms, such as Yahoo and Google, offer online trading services and immediate stock market statistics for the New York and NASDAQ exchanges. However, these statistics are subject to a 20 min delay for public access and are only available without delay to brokerage clients. For this study, we gathered price information for each stock from the Yahoo website on specific dates. In order to ensure the practical applicability of our proposed model, we focused on predicting the direction of the next day’s price rather than the actual price value itself. To achieve this goal, we employed the approach of predicting the market direction as the target variable using the following equation:

\[
\begin{cases}
0, & \text{if } \frac{\text{"Next Day Close Price"} - \text{"Next Day Open Price"}}{\text{"Next Day Open Price"}} \leq 0 \\
1, & \text{if } \frac{\text{"Next Day Close Price"} - \text{"Next Day Open Price"}}{\text{"Next Day Open Price"}} > 0.
\end{cases}
\]  

(1)

This equation considers the relative increase or decrease in the price of each share and enables us to make informed investment decisions by considering market activity. The data collection process from the Yahoo website was facilitated through the utilization of the Yahoo Finance tool. In Table 1, we present the data samples extracted for Apple stock, including their corresponding starting prices. By accurately predicting the direction of price movements, our model aims to provide valuable insights and help make informed investment decisions.
Table 1. Apple stock market data.

<table>
<thead>
<tr>
<th>Date</th>
<th>Open</th>
<th>High</th>
<th>Low</th>
<th>Close</th>
<th>Adj Close</th>
<th>Volume</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/8/2018</td>
<td>174.35</td>
<td>175.61</td>
<td>173.93</td>
<td>174.3</td>
<td>171.03</td>
<td>20,567,800</td>
<td>0</td>
</tr>
<tr>
<td>1/9/2018</td>
<td>174.55</td>
<td>175.06</td>
<td>173.41</td>
<td>174.3</td>
<td>171.01</td>
<td>21,584,000</td>
<td>0</td>
</tr>
<tr>
<td>1/10/2018</td>
<td>173.16</td>
<td>174.3</td>
<td>173</td>
<td>174.2</td>
<td>170.97</td>
<td>23,959,900</td>
<td>1</td>
</tr>
<tr>
<td>1/11/2018</td>
<td>174.59</td>
<td>175.49</td>
<td>174.49</td>
<td>175.2</td>
<td>171.95</td>
<td>18,667,700</td>
<td>1</td>
</tr>
</tbody>
</table>

3.1.2. Twitter API

Various APIs have been developed to extract data from Twitter, which is widely recognized as a public platform. In this study, we used three tools, namely GetOldTweets3 (https://pypi.org/project/GetOldTweets3/, accessed on 2 December 2020), Tweepy (https://pypi.org/project/tweepy/, accessed on 2 December 2020), and Textblob, to collect data from Twitter. Each of these tools has its own capabilities and limitations for accessing and retrieving tweets. GetOldTweets3 is a free service that enables searching through older tweets using hybrid and word search techniques. It provides essential information such as tweet ID (str), permalink (str), username (str), recipient (str), text (str), date and time (DateTime) in UTC, number of retweets (int), number of favorites (int), number of mentions (str), number of hashtags (str), and geolocation (str). Although the data obtained from GetOldTweets3 is informative, it does not include crucial social information, such as the number of followers and followings. To retrieve additional relevant social information for each user, we utilized Tweepy. Last, Textblob, another program used in this study, can extract the emotional sentiment of a tweet.

3.1.3. Statistical Analysis of the Collected Messages

Data was collected from the beginning of 2018 until the end of 2019 for all shares, and, for AAPL, data collection extended until the end of 2020. The total number of tweets collected over the two-year period varied for each share because of differences in the number of tweets per day. Table 2 presents the number of samples collected per share, including information on the related hashtags, the count of positive and negative tweets, and the number of unique user IDs.

Table 2. The characteristics of the collected data from Twitter for the DOW 30 index shares.

<table>
<thead>
<tr>
<th>Hashtag</th>
<th>#Tweets</th>
<th>#Positive Tweets</th>
<th>#Negative Tweets</th>
<th>#Unique User-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAPL</td>
<td>417,334</td>
<td>366,080</td>
<td>51,254</td>
<td>63,355</td>
</tr>
<tr>
<td>AXP</td>
<td>9569</td>
<td>2873</td>
<td>1674</td>
<td>163</td>
</tr>
<tr>
<td>BA</td>
<td>66,637</td>
<td>24,481</td>
<td>10,484</td>
<td>641</td>
</tr>
<tr>
<td>CAT</td>
<td>41,543</td>
<td>13,193</td>
<td>7403</td>
<td>3380</td>
</tr>
<tr>
<td>CSCO</td>
<td>7093</td>
<td>2919</td>
<td>841</td>
<td>187</td>
</tr>
<tr>
<td>CVX</td>
<td>15,214</td>
<td>6322</td>
<td>1739</td>
<td>40</td>
</tr>
<tr>
<td>DIS</td>
<td>153,032</td>
<td>54,067</td>
<td>18,651</td>
<td>877</td>
</tr>
<tr>
<td>DOW</td>
<td>14,697</td>
<td>237</td>
<td>125</td>
<td>285</td>
</tr>
<tr>
<td>GS</td>
<td>59,401</td>
<td>26,514</td>
<td>9042</td>
<td>182</td>
</tr>
<tr>
<td>HD</td>
<td>62,497</td>
<td>21,553</td>
<td>7139</td>
<td>123</td>
</tr>
<tr>
<td>IBM</td>
<td>62,871</td>
<td>19,898</td>
<td>7394</td>
<td>10,440</td>
</tr>
<tr>
<td>INTC</td>
<td>36,816</td>
<td>14,394</td>
<td>5166</td>
<td>551</td>
</tr>
<tr>
<td>JNJ</td>
<td>42,045</td>
<td>14,656</td>
<td>5975</td>
<td>448</td>
</tr>
</tbody>
</table>
Table 2. Cont.

<table>
<thead>
<tr>
<th>Hashtag</th>
<th>#Tweets</th>
<th>#Positive Tweets</th>
<th>#Negative Tweets</th>
<th>#Unique User-ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPM</td>
<td>10,614</td>
<td>3817</td>
<td>1492</td>
<td>99</td>
</tr>
<tr>
<td>KO</td>
<td>16,444</td>
<td>5756</td>
<td>2273</td>
<td>17</td>
</tr>
<tr>
<td>MCD</td>
<td>68,810</td>
<td>23,163</td>
<td>8199</td>
<td>218</td>
</tr>
<tr>
<td>MMM</td>
<td>11,301</td>
<td>3500</td>
<td>2355</td>
<td>47</td>
</tr>
<tr>
<td>MRK</td>
<td>19,687</td>
<td>7610</td>
<td>2202</td>
<td>132</td>
</tr>
<tr>
<td>MSFT</td>
<td>156,643</td>
<td>81,402</td>
<td>13,081</td>
<td>979</td>
</tr>
<tr>
<td>NKE</td>
<td>50,659</td>
<td>17,551</td>
<td>6653</td>
<td>946</td>
</tr>
<tr>
<td>PFE</td>
<td>74,733</td>
<td>26,829</td>
<td>7601</td>
<td>151</td>
</tr>
<tr>
<td>PG</td>
<td>53,866</td>
<td>18,803</td>
<td>5635</td>
<td>193</td>
</tr>
<tr>
<td>TRV</td>
<td>4451</td>
<td>1344</td>
<td>505</td>
<td>318</td>
</tr>
<tr>
<td>UNH</td>
<td>15,576</td>
<td>5574</td>
<td>2280</td>
<td>80</td>
</tr>
<tr>
<td>UTX</td>
<td>9977</td>
<td>3085</td>
<td>1323</td>
<td>267</td>
</tr>
<tr>
<td>V</td>
<td>34,638</td>
<td>12,492</td>
<td>5144</td>
<td>163</td>
</tr>
<tr>
<td>VZ</td>
<td>18,913</td>
<td>7028</td>
<td>2513</td>
<td>116</td>
</tr>
<tr>
<td>WBA</td>
<td>29,835</td>
<td>8002</td>
<td>339</td>
<td>256</td>
</tr>
<tr>
<td>WMT</td>
<td>13,268</td>
<td>5275</td>
<td>1706</td>
<td>313</td>
</tr>
<tr>
<td>XOM</td>
<td>23,439</td>
<td>8153</td>
<td>3558</td>
<td>250</td>
</tr>
</tbody>
</table>

3.2. Feature Extraction

In this section, we delve into a more detailed examination of each of the features employed in the proposed User2Vec model.

3.2.1. Social Information

The social characteristics of a user or their tweets in a social network can significantly influence the impact of their comments on other users and future market changes. For instance, popular users are expected to have a higher level of effectiveness, since their messages reach larger communities compared to unknown users. Similarly, tweets that have been viewed or liked by a greater number of users have a wider reach among other users. Various metrics can measure a user’s activity or reach on Twitter. For instance, key features that can be used for this purpose include the number of followers a user has, the number of users they follow, and the frequency of retweets their tweets receive. Figure 2 illustrates different aspects of these features.

3.2.2. Embedded Text

The text of a user’s tweets on a social network can contain their opinions or news about a stock, which can influence the opinions of other users and market investors, thus ultimately impacting the future direction of the stock in the market. Therefore, this information can serve as an effective feature for market prediction. Incorporating the text of tweets, along with their emotional labels, can reduce the model’s reliance on the accuracy of the sentiment analyzer, thus enhancing the final accuracy of the model. To use the text of each tweet as a feature in the User2Vec model, we required a concise, effective, and consistent representation. This representation was generated by breaking the text of each tweet into its constituent words, thereby treating each tweet as a sequence of words. In the preprocessing step, all stop words and punctuation were removed from the tweets. This removal improves the overall accuracy of the model and enables the classifier to learn better features. The resulting sequence of words can be represented as \([x^1, x^2, \ldots, x^n]\), where \(x^i \in \mathbb{R}^k\) is a k-dimensional word vector corresponding to the \(i\)-th
word in the sequence. In this process, we utilized the pre-trained Word2vec model [75] to map each word to its respective embedded vector. Subsequently, the text of each tweet, being a sequence of fixed-length vectors, was transformed into a single vector. To achieve this transformation, various techniques such as the maximum (max), minimum (min), or average (ave) embeddings of all words in a tweet were widely used [76,77]. In our approach, we employed the average technique to convert the embedding vectors of the tweet’s words into a k-dimensional vector as follows:

\[ S = \text{ave}(x^1, x^2, \ldots, x^n), \]  

where \( n \) is the number of words in a tweet.

3.2.3. Sentiment

The emotional label of a tweet reflects the user’s opinion regarding the desired share at the time the tweet was sent. As a result, it can serve as an important feature in the User2Vec model. The distinction between the User2Vec model and previous approaches lies in how this feature is utilized. In the User2Vec model, the significance of each tweet is determined by employing a deep neural network to incorporate the other proposed user features. Conversely, most previous works rely on calculating the numerical sum of the emotional labels from all users as the final feature for predicting market trend changes. In this study, the TextBlob tool was used to extract the emotional tag of each tweet. This tool provides properties such as subjective, objective, and sentiment for each tweet. The subjective and objective properties are numerical values ranging from 0 to 1, while the sentiment categorizes tweets as negative, positive, or neutral.

3.2.4. User Score

The ability of a user to predict future market changes for a specific share can assess the importance of their comments regarding that share. A user who has showed accurate predictions in the past is likely to continue making correct predictions in the future. Therefore, this feature can be leveraged to estimate the significance of each user’s opinion. It is important to note that a user’s success rate may vary across different shares. Some users might excel at predicting changes in certain shares while not performing as well in others. It becomes necessary to calculate the successes and failures of a user for each individual share separately. To determine a user’s success rate for a particular share, we employed sentiment analysis methods to extract the user’s opinions concerning that share. Subsequently, by comparing these opinions with the actual market data, the accuracy of each prediction can be assessed, and the number of successes and failures can be tallied. Let \( S^d_i \) represent the prediction of tweet \( i \) posted at time \( t \) regarding a target share, and let \( R_t \) denote the actual direction of that share at the same time step. The score of the corresponding tweet can then be calculated as follows:

\[ \text{TweetScore}^d_i = \begin{cases} 
1 & \text{if } s^d_i = R_t \\
-1 & \text{if } s^d_i \neq R_t.
\end{cases} \]  

(3)

Based on the individual scores of the tweets mentioned above, the total scores for correct and incorrect predictions made by user \( u \) can be derived using the following equations:

\[ \text{SuccessUserScore}_u = \sum_{i, \text{user}(i) = u} I(\text{TweetScore}^d_i = 1) \]  

(4)

\[ \text{FailureUserScore}_u = \sum_{i, \text{user}(i) = u} I(\text{TweetScore}^d_i = -1), \]  

(5)

where user\((i)\) represents the user ID of the individual who posted tweet \( i \), and \( I \) is an indicator function that yields 1 when the specified condition is true and yields 0 if otherwise.
The users’ scores are updated at each time step, and, in the proposed model, the final scores of the users were used.

3.2.5. Market Data

Market features comprise six components: OHLC features (open, high, low, and close prices), Adj Close, and volume. The adjusted closing price considers various factors that can influence stock prices after the market closes, such as stock splits, dividends, and rights offers. Volume reflects the number of shares traded. These features are widely employed in many stock market prediction models, as they help expect the future changes in stock prices by capturing the price fluctuations during the current time period. In most instances, market data from preceding days have proven effective in predicting future changes.

3.3. Prediction

The goal of this section of our proposed model is to predict the direction of the market by leveraging the capabilities of deep recurrent neural networks. Figure 3 depicts the comprehensive architecture of the prediction model introduced in this study. The model undergoes an end-to-end training process using the User2Vec feature vectors outlined in the preceding section. To accomplish this, a fusion of CNNs and a stack of bidirectional LSTM networks are employed to process the User2Vec and generate a condensed, high-level feature vector for each day. Subsequently, this extracted feature vector is inputted into a two-layer fully connected network, thereby enabling the prediction of market direction as either bullish or bearish. In the following, we explain the structures of the CNN and LSTM model used in the proposed model.

- **CNNs** have shown their versatility and effectiveness in handling a wide range of data types, including sequences of embedded word vectors. Our proposed model uses a CNN network to process the feature matrix obtained from both numerical and textual data. The structure of the feature matrix is such that each row represents the feature vector of a specific tweet combined with the corresponding numerical market data, while each column captures a distinct feature containing relevant information. To extract meaningful patterns from the feature matrix, our model incorporates two convolutional layers within the CNN component. Each layer comprises fifty 1D filters, thereby enabling the model to capture and analyze local dependencies within the data effectively. Following the convolutional layers, we introduce a batch normalization layer to enhance the performance of the model and stability. This layer normalizes the intermediate outputs of the network, thereby mitigating the impact of internal covariate shifts, as well as improving overall training efficiency and the ability of the model to generalize. By harnessing the power of CNNs in our model architecture, we can efficiently process the combined numerical and textual data. This enables us to extract relevant features and facilitate subsequent analysis for market prediction purposes.

- In our model, we used stacked bidirectional LSTM layers to generate a concise representation of the input data for predicting stock market direction, thus building upon the output of the CNN networks. Stacked structures, comprising two or more LSTM layers, are employed to capture increasingly complex data patterns. In a stacked RNN, the output of the lower layers serves as input for higher layers, thereby enabling the model to achieve varying levels of abstraction across multiple network layers. According to theoretical evidence, a deep hierarchical model can represent some functions more efficiently than a shallow one [78]. Traditional RNNs have a limitation in that they only consider previous context and disregard future context. However, bidirectional LSTM models overcome this limitation by processing data in both forward and backward directions, thereby effectively incorporating both past and future context. Our model incorporates stacked bidirectional LSTM layers to leverage this advantage. The feature vectors obtained from the outputs of the CNNs are fed into the cells of the first bidirectional LSTM network. One LSTM layer processes the data in its original order, while the other LSTM layer performs the same task in reverse. Each input vector
generates two independent output vectors from these two networks: one considering the previous inputs and the other considering the future inputs. These two outputs are concatenated to form an intermediate representation for each input vector, which is then passed as input to the second bidirectional LSTM network. The second network operates similarly, thus producing the final representation of the input data for the last layer. The LSTM transition equations are as follows:

\[
\begin{align*}
    i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
    f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
    c_t &= f_t c_{t-1} + i_t \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
    o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
    h_t &= o_t \sigma_h(c_t),
\end{align*}
\]

where \(x_t\) represents the input at time step \(t\), \(i_t\) displays the input gate activation, \(f_t\) represents the forget gate activation, \(c_t\) serves as the memory cell state, \(o_t\) represents the output gate activation, and \(h_t\) is the hidden state at time step \(t\). \(W\) and \(U\) are weight matrices, and \(b\) denotes bias vectors. The activation functions used are the sigmoid function (\(\sigma_g\)) and the hyperbolic tangent function (\(\sigma_c\) and \(\sigma_h\)).

The output of the LSTM layers provides a bullish or bearish label, thereby representing the prediction of the network for the market direction of the following day.
Figure 3. The architecture of the prediction module of the proposed model.
4. Experimental Analysis

4.1. Evaluation Metrics

In this paper, the proposed model was evaluated using four measures: precision, recall, F-measure, and accuracy. These measures are calculated as follows:

\[ \text{Precision} = \frac{TP}{TP + FP} \]  
\[ \text{Recall} = \frac{TP}{TP + FN} \]  
\[ F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \]  
\[ \text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \]

where \( TP \) represents the number of positive samples correctly classified as positive, \( TN \) represents the number of negative samples correctly classified as negative, \( FP \) represents the number of negative samples incorrectly classified as positive, and \( FN \) represents the number of positive samples incorrectly classified as negative. In this context, the labels “bullish” and “bearish” correspond to positive and negative labels, respectively.

4.2. Experiment Setup

Table 3 presents the optimal network structure and parameter settings for our proposed model, which includes the 1D-CNN, stacked bidirectional LSTM layers, and dense layers. As shown in the table, our model’s CNN module comprises two convolutional layers accompanied by their respective batch normalization layers. To minimize computational requirements, only 50 tweets per day were utilized in the model. The input tweets were selected from the messages of the most popular accounts. For all experiments, 80% of the data was allocated for training the model, while the remaining 20% was reserved for testing. The train and test data were chosen from distinct time periods to ensure that the test set was entirely independent from the training data.

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>filter size</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>kernel size</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>activation function</td>
<td>relu</td>
</tr>
<tr>
<td>LSTM</td>
<td>unit size</td>
<td>20</td>
</tr>
<tr>
<td>Proposed</td>
<td>batch size</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>learning rate</td>
<td>0.01</td>
</tr>
<tr>
<td></td>
<td>optimizer</td>
<td>Adam</td>
</tr>
<tr>
<td></td>
<td>loss function</td>
<td>mean absolute error</td>
</tr>
<tr>
<td></td>
<td>epoch</td>
<td>700</td>
</tr>
</tbody>
</table>

4.3. Pre-Processing

The User2Vec feature set was utilized as the input of the prediction model. However, these features could not be directly used and required pre-processing. In the following, we describe the pre-processing steps undertaken.

- Weekend data exclusion: To preserve the integrity and accuracy of our dataset, we performed a meticulous data pre-processing procedure to eliminate information pertaining to weekends or holidays. This aspect holds significant importance in financial data analysis, since markets are commonly closed during these time periods, thus introducing potential irregularities into the data. By excluding weekends and holidays from our dataset, we concentrated our analysis only on trading days when the market
was active. This approach ensures consistency within our data and prevents any distortions or biases that may arise from incorporating non-trading days.

- Time zone alignment: Achieving the proper alignment of time zones between financial data and text data is of paramount importance to ensure accurate analysis and the seamless integration of information. By establishing consistent time zones, we effectively eliminate discrepancies and allow for a seamless merging of these two essential data sources. In this study, we adopted the United States Eastern Time Zone as our reference time zone for data processing.

- Normalization: In order to mitigate multivariate value divergence, the numerical characteristics were standardized using min–max normalization, thereby ensuring that their values fell within the range of \([0, 1]\). This was achieved by linearly transforming the data values. The equation below describes the min–max normalization for feature vector \(x\).

The min–max normalization formula for normalizing a value \(x^i\) to a target interval \([n_{\text{min}}^i, n_{\text{max}}^i]\) with original boundaries \([\text{min}_x, \text{max}_x]\) is represented as:

\[
\text{Norm}(x^i) = \frac{x^i - \text{min}_x}{\text{max}_x - \text{min}_x},
\]

where \(x^i\) is the \(i\)-th element in \(x\). \(\text{min}_x\) and \(\text{max}_x\) denote the minimum and maximum values of \(x\).

- Tweet embedding layer: Word-embedding networks assign each word in a text to a \(D\)-dimensional vector space, which can subsequently serve as input for a neural network. Here, we utilized the word2vec word-embedding network, which was trained on a corpus of 100 billion words sourced from Google News. This network assigns a 300-dimensional vector to each of the 3 million words within its vocabulary. Each word in a tweet’s text is converted to a 300-dimensional vector. Then, by averaging these vectors, a 300-dimensional representation is produced for each tweet.

4.4. Model Performance

The aim of this section is to assess and analyze the accuracy of the predictions made by our proposed model. In order to achieve this, we compared the accuracy of our model with various baselines and state-of-the-art models in the field. We examined the impact of different components of the proposed model on the prediction accuracy.

4.4.1. Ablation Study

In this section, we investigated the impact of various components of our proposed User2Vec representation on the predictive accuracy of our model. We compared the outcomes of various variants of our model with several baselines. In the following section, we begin by introducing the baseline and the different versions of our model, and we then subsequently review their respective results.

Baselines: We compared the outcomes of our proposed model with those of several methods, commonly known as baseline methods, in previous works within this field for predicting the direction of the stock market one day in advance. The baseline models used for comparison in this study include the following:

- Rand: A naive Bayes predictor that disregards all tweets and generates random predictions for market directions, with a 50% probability of success for both upward and downward trends.
- Arima(M) [17]: A traditional prediction method that solely relies on market information.
- SBiLSTM-Day2Vec(M,Se) [6]: A state-of-the-art deep neural network baseline that predicts stock trends by incorporating a sequence of market data (M) and sentiments of the messages (Se) for each day. Many previous works in the field use this model with variations, such as using daily aggregated sentiment or incorporating additional layers such as attention [26], ref. [55], to predict the stock’s direction. Unlike our proposed
model, which uses the sentiment of each tweet separately, this model aggregates the sentiment for an entire day.

User2Vec variants: We investigated several variations in our proposed model, with each utilizing a distinct subset of the proposed feature set. The prediction network structure remained consistent across all models. The objective was to assess the influence of each feature and combinations thereof on enhancing the accuracy. The models are outlined as follows:

- **User2Vec (M,Se):** This model incorporates market data and sentiment analysis as its feature set, which is a commonly employed approach to stock market prediction. Many prior studies have utilized these features in their models. Comparing the results with the outcomes of the baseline models is limited to the market data and sentiment tag (SBiLSTM-Day2Vec(M,Se)), which allows us to observe the impact of the primary contribution of this paper: converting each tweet of a user into a vector instead of converting all messages of a day into a vector.
- **User2Vec(Txt):** In this model, solely the text section of the tweet is utilized to construct the input vector. This approach aims to investigate whether utilizing the text of the tweets directly can be effective or if it is preferable to employ the extracted sentiment from them.
- **User2Vec(Txt,M,Se):** By comparing the results with a model that only uses the market and sentiment data, we can examine the effect of using the embedded text representations of tweets to enhance the accuracy of the model’s predictions.
- **User2Vec(Txt,M,Se,So):** By comparing the results with the previous model, we can determine whether incorporating a social analysis of users can enhance the prediction accuracy or not.
- **User2Vec(Txt,M,Se,Sc):** Similarly to the previous model, this model explores the impact of utilizing users’ scores on the model’s success rate.
- **User2Vec(M,Se,So):** The only missing feature in this model is the text of the tweets. As the length of the embedded text is significantly larger than the numerical features, excluding this feature reduces the input size and the model’s parameters. Compared to the full model, the objective of this model is to examine the contribution of the textual component of the input data.
- **User2Vec(Txt,M,Se,So,Sc):** This represents the complete version of our proposed model, wherein it incorporates the entire set of features.

Results: All the aforementioned models were trained and tested using data from Apple’s stock, and the results are presented in Table 4. Based on these results, all the User2Vec models outperformed the baseline models in terms of accuracy. This reveals the effectiveness of the concept of converting each user’s opinion into a vector and utilizing a specific number of these vectors daily to predict the market trend for the following day. Specifically, the accuracy of our proposed User2Vec(M,Se) model significantly surpassed that of the SBiLSTM-Day2Vec(M,Se) model, which employed the same feature set but aggregated the entire message content of a day into a single vector.

Another significant finding from this experiment pertains to the impact of text features. Sentiment analysis is the most prevalent application of tweets or news text in stock market forecasting. In most studies, the market’s direction is subsequently predicted based on the sentiment analysis of tweets for each day. The text of the tweets is not directly utilized in these models. Constraining the models to interpret only three distinct sentiments (positive, negative, and neutral) from each tweet leads to a loss of valuable information inherent in the tweet’s text. Furthermore, the performance of the model is also constrained by the accuracy of the sentiment analysis method. In our proposed model, we directly incorporated the text of each tweet, along with its sentiment, as part of the feature set to address these limitations. Based on the results presented in Table 4, including the textual content in addition to the sentiment score enhanced the accuracy from 71% in
User2Vec(M,Se) to 74% in User2Vec(Txt,M,Se), and from 72% in User2Vec(M,Se,So) to 78% in User2Vec(Txt,M,Se,So).

Table 4. Results obtained using different models on Apple stock.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy ± 95% CI</th>
<th>Precision ± 95% CI</th>
<th>Recall ± 95% CI</th>
<th>F-Measure ± 95% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand</td>
<td>0.4623 ± 0.0055</td>
<td>0.5645 ± 0.0047</td>
<td>0.6412 ± 0.0061</td>
<td>0.5012 ± 0.0070</td>
</tr>
<tr>
<td>Arima(M)</td>
<td>0.5105 ± 0.0035</td>
<td>0.5955 ± 0.0033</td>
<td>0.7022 ± 0.0050</td>
<td>0.5345 ± 0.0063</td>
</tr>
<tr>
<td>SBiLSTM-Day2Vec(M,Se)</td>
<td>0.6255 ± 0.0042</td>
<td>0.6245 ± 0.0039</td>
<td>0.9602 ± 0.0022</td>
<td>0.7523 ± 0.0028</td>
</tr>
<tr>
<td>User2Vec(M,Se)</td>
<td>0.7140 ± 0.0030</td>
<td>0.6902 ± 0.0031</td>
<td>0.9015 ± 0.0023</td>
<td>0.7812 ± 0.0025</td>
</tr>
<tr>
<td>User2Vec(Txt)</td>
<td>0.6815 ± 0.0037</td>
<td>0.7241 ± 0.0033</td>
<td>0.7255 ± 0.0034</td>
<td>0.7210 ± 0.0035</td>
</tr>
<tr>
<td>User2Vec(Txt,M,Se)</td>
<td>0.7444 ± 0.0027</td>
<td>0.7212 ± 0.0032</td>
<td>0.9045 ± 0.0021</td>
<td>0.8019 ± 0.0024</td>
</tr>
<tr>
<td>User2Vec(Txt,M,Se,So)</td>
<td>0.7810 ± 0.0025</td>
<td>0.7845 ± 0.0024</td>
<td>0.8615 ± 0.0026</td>
<td>0.8245 ± 0.0023</td>
</tr>
<tr>
<td>User2Vec(M,Se,So)</td>
<td>0.7242 ± 0.0031</td>
<td>0.7200 ± 0.0032</td>
<td>0.8242 ± 0.0028</td>
<td>0.7712 ± 0.0027</td>
</tr>
<tr>
<td>User2Vec(Txt,M,Se,Sc)</td>
<td>0.7344 ± 0.0030</td>
<td>0.7314 ± 0.0032</td>
<td>0.8300 ± 0.0027</td>
<td>0.7842 ± 0.0026</td>
</tr>
<tr>
<td>User2Vec(Txt,M,Se,So,Sc)</td>
<td>0.7514 ± 0.0029</td>
<td>0.8100 ± 0.0023</td>
<td>0.7311 ± 0.0031</td>
<td>0.7710 ± 0.0027</td>
</tr>
</tbody>
</table>

The next feature examined in this experiment is the social status information of Twitter users. This feature is seldom used in stock market prediction. In previous studies, relevant social features have been identified first, and then tweets were selected based on the values of these features. However, in our approach, we selected tweets from users with the highest number of followers each day and incorporated all the social information from those tweets in our proposed feature set. It is also obvious that individual messages from a particular user can exert differing impacts on the prediction. For instance, a message that garners more attention and is retweeted multiple times holds greater significance in predicting the future of a stock. Therefore, we also considered the social information conveyed by the messages. As the results indicate, incorporating social information improved the model’s accuracy from 74% in User2Vec(Txt,M,Se) to 78% in User2Vec(Txt,M,Se,So), from 73% in User2Vec(Txt,M,Se,Sc) to 75% in User2Vec(Txt,M,Se,So,Sc), and from 71% in User2Vec(M,Se) to 72% in User2Vec(M,Se,So). The results indicate that incorporating user scores not only cannot increase the accuracy, but also decreases it, wherein it was reduced from 74% in the User2Vec(Txt,M,Se,So) model to 73% in the User2Vec(Txt,M,Se,So,Sc) model.

4.4.2. Analysis and Comparison with the State of the Art

Based on the results got from the previous experiment, the User2Vec(Txt,M,Se,So) model was chosen as the final configuration of our proposed model. This model incorporates market data, tweet sentiment, the social information of users and tweets, and the text of the tweets. The model was applied to various companies within the Dow 30 stocks, and the results for predicting the next day’s closing price are presented in Table 5. The findings reveal that the AAPL achieved the highest accuracy, while the mean accuracy for the other stocks exceeded 60%. Upon closer examination of the stocks with lower prediction accuracies, such as GS and IBM, it becomes apparent that these companies generate less buzz on social media, resulting in a considerably smaller number of messages about them compared to companies with higher prediction accuracies like AAPL. This discrepancy is also clear in the number of unique users (IDs) who have tweeted about these stocks. For instance, while 63,355 users expressed their opinions about AAPL, the corresponding number for GS was only 182. Once again, this highlights the significance of expert opinions within our proposed model.
Table 5. Results obtained using the proposed model for various companies on the Dow 30 stocks.

<table>
<thead>
<tr>
<th>Stock</th>
<th>Accuracy ±</th>
<th>Precision ±</th>
<th>Recall ±</th>
<th>F-Measure ±</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>0.7410 ± 0.0015</td>
<td>0.7443 ± 0.0023</td>
<td>0.7546 ± 0.0021</td>
<td>0.7494 ± 0.0018</td>
</tr>
<tr>
<td>CAT</td>
<td>0.6358 ± 0.0034</td>
<td>0.6614 ± 0.0036</td>
<td>0.6728 ± 0.0031</td>
<td>0.6670 ± 0.0033</td>
</tr>
<tr>
<td>CSCO</td>
<td>0.7143 ± 0.0027</td>
<td>0.7059 ± 0.0025</td>
<td>0.7301 ± 0.0028</td>
<td>0.7177 ± 0.0026</td>
</tr>
<tr>
<td>CVX</td>
<td>0.6351 ± 0.0033</td>
<td>0.6406 ± 0.0037</td>
<td>0.6601 ± 0.0035</td>
<td>0.6502 ± 0.0036</td>
</tr>
<tr>
<td>GS</td>
<td>0.5682 ± 0.0042</td>
<td>0.5816 ± 0.0043</td>
<td>0.6012 ± 0.0041</td>
<td>0.5911 ± 0.0042</td>
</tr>
<tr>
<td>IBM</td>
<td>0.5987 ± 0.0038</td>
<td>0.6027 ± 0.0039</td>
<td>0.6128 ± 0.0037</td>
<td>0.6077 ± 0.0038</td>
</tr>
<tr>
<td>JNJ</td>
<td>0.6214 ± 0.0035</td>
<td>0.6137 ± 0.0037</td>
<td>0.6208 ± 0.0036</td>
<td>0.6170 ± 0.0037</td>
</tr>
<tr>
<td>JPM</td>
<td>0.7273 ± 0.0022</td>
<td>0.7143 ± 0.0024</td>
<td>0.8333 ± 0.0023</td>
<td>0.7692 ± 0.0025</td>
</tr>
<tr>
<td>MCD</td>
<td>0.6182 ± 0.0036</td>
<td>0.6363 ± 0.0037</td>
<td>0.6423 ± 0.0038</td>
<td>0.6392 ± 0.0037</td>
</tr>
<tr>
<td>MSFT</td>
<td>0.6605 ± 0.0033</td>
<td>0.7046 ± 0.0034</td>
<td>0.7911 ± 0.0032</td>
<td>0.7452 ± 0.0035</td>
</tr>
<tr>
<td>KO</td>
<td>0.6061 ± 0.0039</td>
<td>0.6230 ± 0.0041</td>
<td>0.6545 ± 0.0040</td>
<td>0.6381 ± 0.0042</td>
</tr>
</tbody>
</table>

Our proposed algorithm was compared with eight state-of-the-art models, specifically those of Sharma et al. [43], Jin et al. [60], Liu et al. [61], Bouktif et al. [71], Mehta et al. [79], Wang C et al. [80], Wang Z et al. [81], and Wang J et al. [82]. Proposed with Transformer is a model that uses a transformer instead of LSTM. This comparative study was meticulously conducted to reveal the superiority and effectiveness of our model in the domain of stock market prediction. The comparative results are summarized in Table 6, which showcase the performance of all the models on Dow 30 stocks. In every key performance metric, our proposed model outperformed the others. Notably, it superseded the model by Wang J et al. [82], which was our stiffest competition across all criteria. The enhanced performance of our model is striking when considering the reduction in error. In two primary evaluation metrics, namely, accuracy and F-measure, our model successfully diminished the error rate by over 12% and 9%, respectively. This substantial decrease in error is a testament to the superior predictive capabilities of our algorithm. A comparison of our model with Proposed with Transformer helped to highlight the importance of LSTM over the transformer model. This comparison was insightful, as it revealed a significant reduction in the error rate by approximately 8.49% when using LSTM. This result further underscores the effectiveness of LSTM in handling the intricacies of time-series data, such as stock market prices, and its role in bolstering the accuracy of our proposed model.

Table 6. Results obtained using the proposed model and other state-of-the-art methods on the Dow 30 stocks.

<table>
<thead>
<tr>
<th>Work</th>
<th>Accuracy ±</th>
<th>Precision ±</th>
<th>Recall ±</th>
<th>F-Measure ±</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sharma et al. [43]</td>
<td>0.5201 ± 0.0035</td>
<td>0.5404 ± 0.0037</td>
<td>0.5302 ± 0.0039</td>
<td>0.5335 ± 0.0038</td>
</tr>
<tr>
<td>Jin et al. [60]</td>
<td>0.5523 ± 0.0031</td>
<td>0.5644 ± 0.0030</td>
<td>0.5801 ± 0.0032</td>
<td>0.5721 ± 0.0031</td>
</tr>
<tr>
<td>Liu et al. [61]</td>
<td>0.5804 ± 0.0029</td>
<td>0.5909 ± 0.0028</td>
<td>0.6050 ± 0.0030</td>
<td>0.5979 ± 0.0029</td>
</tr>
<tr>
<td>Bouktif et al. [71]</td>
<td>0.6003 ± 0.0027</td>
<td>0.6110 ± 0.0026</td>
<td>0.6251 ± 0.0028</td>
<td>0.6180 ± 0.0027</td>
</tr>
<tr>
<td>Mehta et al. [79]</td>
<td>0.6250 ± 0.0024</td>
<td>0.6355 ± 0.0023</td>
<td>0.6485 ± 0.0025</td>
<td>0.6359 ± 0.0024</td>
</tr>
<tr>
<td>Wang C et al. [80]</td>
<td>0.6455 ± 0.0022</td>
<td>0.6563 ± 0.0021</td>
<td>0.6688 ± 0.0023</td>
<td>0.6525 ± 0.0022</td>
</tr>
<tr>
<td>Wang Z et al. [81]</td>
<td>0.6650 ± 0.0020</td>
<td>0.6661 ± 0.0019</td>
<td>0.6880 ± 0.0021</td>
<td>0.6720 ± 0.0020</td>
</tr>
<tr>
<td>Wang J et al. [82]</td>
<td>0.6800 ± 0.0019</td>
<td>0.6913 ± 0.0018</td>
<td>0.7132 ± 0.0020</td>
<td>0.7072 ± 0.0019</td>
</tr>
<tr>
<td>Proposed with Transformer</td>
<td>0.7051 ± 0.0017</td>
<td>0.7269 ± 0.0012</td>
<td>0.7483 ± 0.0018</td>
<td>0.7324 ± 0.0017</td>
</tr>
<tr>
<td>Proposed</td>
<td>0.7661 ± 0.0016</td>
<td>0.7667 ± 0.0014</td>
<td>0.7878 ± 0.0018</td>
<td>0.7722 ± 0.0018</td>
</tr>
</tbody>
</table>
Figure 4 presents the receiver operating characteristic (ROC) curves for the methodologies highlighted in Table 6. The area under the curve (AUC), a scalar measure of a classifier’s performance, was used to supplement our analysis. An AUC score of 1 signifies impeccable discrimination, while a score of 0.5 reflects an absolute lack of discrimination akin to random guessing. Notably, our proposed method surpassed all the other methods in achieving a remarkable AUC of 0.64. This superior performance underscores its substantial ability to accurately differentiate between positive and negative outcomes, thus bolstering the standing of our proposed method as an effective tool for stock market prediction. The Proposed with Transformer method secured an AUC of 0.60, thus highlighting its reliable ability to discern between positive and negative instances. These findings further emphasize the resilience and effectiveness of our proposed approach, whether it is implemented independently or in tandem with a transformer architecture. In contrast, the models presented by Wang J et al., Wang Z et al., and Wang C et al. [80–82], achieved only modest AUC values of 0.57, 0.53, and 0.53, respectively. While their discriminatory capabilities were satisfactory, they fell short when compared to our proposed methods. Further down the performance spectrum, Mehta et al., Bouktif et al., Liu et al., Jin et al., and Sharma et al. [43,60,61,71,79], demonstrated lesser performance, with AUC values spanning from 0.51 to 0.44. Among these, Sharma et al. found itself at the bottom with an AUC of a mere 0.44, which is only slightly better than pure chance. Our ROC analysis provides a clear demarcation of the varying performances of the evaluated methods. The unmatched predictive capacity of our proposed method, both independently and with a transformer, exemplifies the effectiveness of our approach. This analysis not only paves the way for potential improvements, but also unveils promising applications in the realm of stock prediction.

![ROC diagram for the proposed model and state-of-the-art methods on the Dow 30 stocks [43,60,61,71,79–82]. Bule dashed line represents the ROC curve for a random guess.](image)

Figure 4. ROC diagram for the proposed model and state-of-the-art methods on the Dow 30 stocks [43,60,61,71,79–82]. Bule dashed line represents the ROC curve for a random guess.

Finally, we conducted statistical tests to evaluate the significance of the superiority of our proposed model. To achieve this, we extracted 10 train/test datasets from the original dataset and calculated the accuracy of both models for each dataset. Out of all
the models studied, only the implementation of SBiLSTM-Day2Vec(M,Se) was available for comparison. By performing a t-test between our proposed model and the SBiLSTM-Day2Vec(M,Se) model, we obtained a p-value of 0.0002 and a t-value of 6.2006. These results clearly indicate that there was a significant difference between the outcomes of the two models. The average accuracy of our model across these ten runs was 74, with a standard deviation of 4. In contrast, the SBiLSTM-Day2Vec model had a mean accuracy of 62 and a standard deviation of 0.5.

4.4.3. Discussion

Our research introduced the novel User2Vec model for predicting stock market trends. A unique feature of User2Vec is its capacity to assign weights to different user opinions based on their respective social metrics, thus recognizing that each opinion carries varying levels of influence on specific stocks. This model effectively integrates and processes various sources of information, including numerical market data, sentiment from social network messages, the textual content of messages, and associated social information of users. Empirical studies conducted on the Dow Jones 30 stock market show a clear enhancement in prediction accuracy, thereby demonstrating the superiority of User2Vec over existing state-of-the-art models. Most existing models in this domain predominantly focus on numerical market data and the sentiment analysis of social network messages. However, our User2Vec model expands on these traditional approaches by assimilating not only the sentiment, but also the social information associated with the users and the textual content of the messages. Previous models often fall short in effectively integrating these disparate sources of information, which presents a significant gap in the existing research. Our approach addresses this gap by offering a holistic approach to data assimilation that significantly improves prediction accuracy. The enhanced accuracy of User2Vec is likely because of its unique approach in considering the varying influence of different user opinions. Traditional models often treat all user opinions equally and neglect the reality that certain users or messages may have a larger impact on stock performance. By encoding each message as a vector and feeding it into a convolutional neural network, our model effectively processes a vast array of information from different sources. It subsequently uses a stacked bidirectional LSTM model to provide a comprehensive representation of the input data over a period. Our findings support the value of considering the social information associated with users and messages. This inclusion recognizes that the context in which an opinion is formed and shared can have significant bearing on its weight and impact, thus offering more nuanced predictions.

While User2Vec offers an innovative framework for predicting stock market trends, it is crucial to recognize its limitations. One potential drawback lies in its dependence on the precise social metrics used to assign weight to user opinions. If these benchmarks are inaccurate or skewed, it could negatively affect the model’s predictive performance. User2Vec’s evaluation was primarily focused on the Dow Jones 30 stock market, which leaves the model’s broad applicability to other markets or sectors unproven. Future research needs to address these limitations by enhancing both the model’s applicability and robustness. Another potential pitfall is the noise and misinformation within social network textual messages. Platforms such as these are prone to fake news, spam, and manipulation, which potentially could lead to imprecise sentiment analysis and skewed user opinion perceptions. Such inaccuracies could compromise the reliability of user opinion weights and subsequently impair the model’s predictive precision. It is imperative for future research to explore methods to counteract such noise and misinformation, such as implementing robust filtering algorithms or integrating credibility assessments of user-generated content. The model’s dependence on historical data for training and evaluation raises questions about its adaptability to changing market dynamics. Given the stock market’s susceptibility to evolving trends, patterns, and shifts in investor sentiment, the capacity of the model to adjust to new market conditions and change its predictions accordingly remains unclear. Future studies could investigate approaches to incorporating adaptive learning techniques
or integrating real-time data streams to improve the model’s adaptability. The computational complexity of the User2Vec model may also present scalability issues. The processes of encoding messages, training deep learning architectures, and conducting comprehensive evaluations require substantial computational resources and time. With the continuously growing volume of social network data, optimizing the model’s efficiency and scalability is crucial. Future research should investigate techniques to optimize computations and improve User2Vec’s scalability, thereby facilitating its applicability to larger datasets and supporting real-time prediction systems.

Integrating multicriteria decision-making methods, such as the Analytic Hierarchy Process (AHP), the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), or Multi-Attribute Utility Theory (MAUT), might offer a more nuanced approach to understanding and quantifying the impact of user opinions on stock market predictions [83,84]. These multicriteria methods are adept at handling complex decision-making scenarios involving multiple factors. Their implementation could help to capture the multifaceted nature of user opinions and the diverse ways these opinions can influence stock market trends. The exploration of hybrid methodologies, which would combine multicriteria methods with complementary techniques such as machine learning, artificial neural networks, or clustering algorithms, could be valuable. Such an approach would provide a more comprehensive perspective to leverage both quantitative and qualitative information to weight user assessments more accurately and reliably. A promising research direction might be the introduction of a dynamic weighting system, where the weights of user opinions are continuously updated based on their predictive accuracy. This could aid in more precisely and adaptively identifying influential users. Future work could investigate User2Vec’s robustness by testing it against other financial markets and comparing its performance with other state-of-the-art models. This could not only validate the model’s generalizability but also offer further insights into how the integration of social metrics can enhance stock market prediction across different contexts. Last, the potential for real-time prediction and analysis should be explored by developing strategies for the model to react to rapidly evolving market conditions and adapt its predictions accordingly. This advancement could transform User2Vec into a practical tool for real-time financial decision making.

5. Conclusions

The influence of investor sentiment on stocks, coupled with market prices, has been a cornerstone in a multitude of studies focusing on predicting future stock price trends. These studies consistently affirm that including textual opinions can boost the accuracy of predictions. Considering this concept, we have introduced a groundbreaking representation for stock market prediction in this paper named User2Vec. In User2Vec, each message undergoes encoding as a vector and is subsequently processed through a convolutional network. The output from this network is then fed into a bidirectional LSTM network, which furnishes the final representation deployed in the classification network. Our model markedly deviates from previous efforts in this field by integrating accessible social information pertaining to the users and messages, as well as the actual text of the messages. This integration is realized while also considering the sentiment of each message, with each message undergoing separate encoding. Our experimental results decisively showed that including tweet text supplied valuable information for predicting stock direction, even when sentiment from the message was already available. Integrating social information from users and their tweets further improved the results. When juxtaposed with results from state-of-the-art models, the superior performance of our proposed model is apparent. However, it’s essential to note that the effectiveness of our model hinges on the presence of informative messages about the target stock. For stocks associated with a sparse number of messages, the model performance diminishes and reflects the level of performance exhibited by previous models.
This paves the way for future improvements in our User2Vec model. To counteract sparse messages, future versions of the model might incorporate alternative data sources or leverage techniques to extrapolate information from the limited available data. Further enhancements can also be introduced, such as refining the method of encoding messages, enabling the extraction of more nuanced information from the text of the tweets, or augmenting the system’s capacity to handle the growing volume of data in social networks. Lastly, one can also explore the prospects of integrating more sophisticated machine learning techniques to improve the accuracy and robustness of predictions, thereby making User2Vec an even more powerful tool for stock market prediction.

**Author Contributions:** Conceptualization, P.E.; methodology, M.S. and A.N.; software, M.S.; validation, P.E.; formal analysis, P.E.; investigation, P.E.; resources, M.S.; data curation, A.N.; writing—original draft preparation, P.E.; writing—review and editing, P.E., M.S. and A.N.; visualization, M.S. and A.N.; supervision, A.N.; project administration, A.N. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Data will be made available on request.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**


27. Qiu, M.; Song, Y. Predicting the direction of stock market index movement using an optimized artificial neural network model. PloS ONE 2016, 11, e0155133. [CrossRef] [PubMed]


65. Zhang, X.; Fuehres, H.; Gloo, P.A. Predicting stock market indicators through twitter “I hope it is not as bad as I fear”. *Procedia-Soc. Behav. Sci.* 2015, 21, 323–340. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.