Stock Prediction Based on Twitter Sentiment Extraction Using BiLSTM-Attention

Dhomas Hatta Fudholi[1], Royan Abida N. Nayoan[2], Septia Rani[3]
[1,2,3] Department of Informatics, Universitas Islam Indonesia, Indonesia

Article Info

Article history:

Received Feb 13, 2021
Revised Mar 15, 2022
Accepted Mar 31, 2022

Keyword:

Attention
Bidirectional LSTM (BiLSTM)
Sentiment Analysis
Stock Prediction
Twitter

ABSTRACT

A profitable stock price prediction will yield a large profit. According to behavioural economics, other people's emotions and viewpoints have a significant impact on business. One of them is the rise and fall of stock prices. Previous studies have shown that public sentiments retrieved from online information can be very valuable on market trading. In this paper, we propose a model that works well in predicting future stock prices by using public sentiments from social media. The online information used in this research is financial tweets collected from Twitter and the stock prices values retrieved from Yahoo! Finance. We collected tweets related to Netflix Company stocks and the stock prices for the same period which is 5 years from 2015 to 2020 as the dataset. We extracted the sentiment value using VADER algorithm. In this paper, we apply a Bidirectional Long Short-Term Memory (BiLSTM) architecture to achieve our goal. Moreover, we created seven different experiments with different stock price parameters and selected sentiment values combinations and investigated the model by adding an attention layer. We experimented with two different sentiment values, tweet's compound value and tweet's compound value multiplied by favorites count. We considered the favorites count as one representation of public sentiments. From the seven experiments, the experiment with Bidirectional Long Short-Term Memory (BiLSTM) - attention model combined with our selected stock price parameters namely close price, open price, and using Twitter sentiment values that are multiplied with the tweet's favorites count yields a better RMSE result of 2.482e-02 in train set and 2.981e-02 in the test set.

Corresponding Author:

Dhomas Hatta Fudholi,
Department of Informatics, Universitas Islam Indonesia,
Jl. Kaliurang km 14.5, Sleman, Yogyakarta, Indonesia 55584
Email: hatta.fudholi@uii.ac.id

1. INTRODUCTION

Stock price forecasting is a very challenging and important task as it can affect the economy. A successful stock price prediction will directly result in a significant profit, while an unsuccessful one will result in financial loss. Hence, many researchers in various disciplines such as economics, financial engineering, statistics, operations research, computer science, and machine learning are attracted to stock price prediction [1]–[3]. As stated by behavioral economics, business is strongly influenced by the emotions and opinions of others [4]–[6]. When the public sentiment of a company is positive, there are possibilities that the stock price will rise and if the public sentiment is negative, then the stock price may fall. Based on the premise of the behavioural economics, we created a model by using tweets sentiment as a variable in predicting the future stock price.

A lot of parameters need to be considered in financial data, hence predicting stock price is a complex task [7]. Opening price, closing price, public opinions, political issues, and many other parameters are deemed important in the business domain. Stock price prediction is a task which assumed that price movements have a certain pattern to follow such as the public sentiments. Many previous studies on stock price prediction considered the opinion of publics and their models yielded a good result [8], [2].

Nowadays, due to the popularity of social media, a large amount of data is available. Therefore, data can easily be collected from social media, such as Twitter. Twitter is an excellent platform which allows users to send short texts known as tweet to express their thoughts and opinions publicly about any subject. As a public platform, it has a significant impact on public opinion. In recent year, not only Twitter has become very popular, but also becoming very popular for researchers in stock prediction task [1], [9]. According to statistics, a total number of monthly active Twitter users are approximately 330 million and around 500 million tweets are sent per day. These statistics shows a massive data in Twitter which contain various information and topics that are ready to be utilized. In this research, we use the financial topic which will be beneficial in helping future stock price prediction.

Previous researches have used a variety of data that are able to represents the public sentiments. [8] utilized the news articles from New York Times to perform the sentiment analysis and Yahoo! Finance stock price as the datasets. With the help of two open-source libraries, SentiMo and VADER algorithm are used in extracting sentiments. Study [2] also used the news articles from a news provider in India, moneycontrol.com and scraped the data by using BeautifulSoup. Instead of using open-source library, Shah created a dictionary to perform sentiment analysis and then classified them into positive, negative, or neutral. Another study on stock prediction carried out by [10] used a Twitter data to predict public mood, and later on, aggregated the mood result with previous DJIA to predict the stock market movements. His research obtained 87% accuracy in predicting stock market. Later on, a study carried out by [11] investigated if twitter feeds related to a company has a connection in the company stock market prediction. His research which also uses VADER algorithm in performing sentiment analysis obtained a positive correlation of 0.7815. This result shows that there is a strong correlation between tweets and company's stock prices. Another studies also stated that positive news or tweets regarding a company will encourage people to invest in stock and the company's stock price will definitely also rise [11]–[14].

In light of the usage of deep learning techniques in modeling time series data, [15] investigated a BiLSTM in predicting financial time series. Their paper reported the analysis by comparing BiLSTM and LSTM models and concluded that BiLSTM has a better prediction than using LSTM and ARIMA technique. Another study combined an LSTM architecture and attention layer in predicting the stock price was carried out by [16]. Their paper reported that LSTM and attention layer has better performance in extracting information and with addition of attention layer, their model doesn't only focus on important factors but also is able to capture the relevant time steps. [17] also reported that LSTM model was superior compared to the other model studied in [19].

This research is an attempt to build a model to predict future stock prices with an additional data of public sentiments on Twitter. In this research, five-year span of historical data and financial tweets of Netflix Company are used for the data. Netflix data is used due to few reasons. First, we found that Netflix stock price is quite fluctuated. It can be a good dataset which can show the ability of the developed model in predicting a fluctuated price rather than the stable one. Second, we can find pretty much tweets regarding to Netflix, since many people use its services. Few studies have shown the use of different models to predict Netflix stock price. Previous model applied ARIMA (Auto Regressive Integrated Moving Average model) to get the accurate result in forecasting Netflix stock price [18]. This study compares ARIMA and its customized methodology five years Netflix dataset. Study [19] predicts the stock markets using three stocks exchange of Apple Inc. (AAPL), Amazon.com Inc. (AMZN), and Netflix Inc. (NFLX) and the financial news from well-known news publishers to get the sentiment feature. The research predicted the stock price using a binary classification of uptrend and downtrend. Three models of Logistic Regression, Random Forest, and Gradient Boosting Machine are compared that resulted in Random Forest having the highest accuracy of 63.58%. There are notable differences of these research with our proposed research which become our contribution. We use Twitter sentiment as one factor of the prediction and we use BiLSTM with attention mechanism as the base architecture. In extracting the polarity of the sentiment of each tweet, VADER algorithm is used. We created seven different experiments with different stock price parameters and sentiment values and experimenting by adding attention layer to improve the model. Stock price parameters that are considered in this research are namely close price, open price, high value, and low value, while in sentiment parameters, we considered the sentiment values of each tweet and the sentiment values of each tweet multiplied by tweet's favorites count (the amount of likes the tweet received).

The rest of the paper is organized as follows. The methodology used to predict future stock prices is presented in Section 2. Results and discussion of the model is presented in Section 3. Section 4 concludes the paper.

2. METHOD

In this section, we elaborate the method used in our study. Our work consists of several stages, including collecting Netflix stock price data and public sentiments on Twitter, data preparation, sentiment extraction, stock prediction, and evaluation

2.1. Data Collection

The datasets used to predict stock prediction are Netflix stock price retrieved from Yahoo! Finance and tweets related to Netflix stocks scraped from Twitter. Twitter dataset is used as the representation of public opinion. Stock price data contains many parameters, to name a few 'Close', 'Open', 'Low', 'High'. This research experimented by making use of these four columns, namely open price, close price, high value, and low value to obtain the best result in predicting stock price. Open price is the first or opening price of a security upon an exchange on a trading day. Close price is the last price at the end of a trading day. Low value shows the lowest price on a trading day, while high value shows the highest price on a trading day. On the tweet data part, we collect the tweet, date, and favourite count of the tweet. We spesified the date range of the tweet collected.

2.2. Data Preparation

Both stock price and tweets data need to be preprocessed before the data is finally ready to use in the modelling stage. The date on tweet data is reformatted to match the date format of the retrieved stock price, since later, both values of stock price and the sentiment value will be used alongside. The next step is to perform addition of 1 on each tweet favourite count. The favourite count shows that many people agreed with the tweet posted which can be considered as the representation of public sentiments. The favourite count cannot be zero since we will perform multiplication on the sentiment value. In tweet data preprocessing we performed: (i) tweet-preprocessing by removing number, URLs, mention, and reserved words, (ii) casefolding, (iii) removing non-ascii character, (iv) replacing contraction words to its original form (e.g., "won't" to "will not"), (v) removing punctuation, (vi) removing stopwords and (vii) removing words that are less than 2 characters.

After the stock price data and Tweets data are prepared, stock price and Tweets data are joined and grouped by the date. Since several stock price data collected still has missing values, the next step is to fill the missing data by averaging the next and previous close price, open price, low price, and high price from the current missing data. The last step is to normalize the data since all the variables need to be normalized to the same range using the number -1 and +1 as the predefined boundaries. Normalization is performed on data to prevent features with large ranges affecting the metric [20]. Normalization is carried out so that data revolves around a certain value. This step helps in making the data flexible and helps in lowering ambiguity. The Min-Max scaler is used as a technique to specifically fit all the variables in pre-defined boundaries [21]. The formula for Min-Max can be seen in Equation (1), where A is the range of the original data, A' is the Min-Max normalized data, C and D as the predefined boundaries.

$$A' = \left(\frac{A - \min Value\ of\ A}{max\ Value\ of\ A - \min Value\ of\ A}\right) * (D - C) + C \quad (1)$$

2.3. Sentiment Extraction

We use valence aware dictionary for sentiment reasoning (VADER) algorithm for sentiment extraction. VADER algorithm extracts four different values, namely positive value, negative value, neutral value, and compound value. VADER will extract the sentimental word and its valence score which is ranged from +4 (most positive) and -4 (most negative). In this research, we only utilized the compound value. Compound value is obtained by computing the valence scores of each word in the lexicon which is adjusted to the rules and then normalized between -1 as the most extreme negative and +1 as the most extreme positive. If the text has more than one sentiment, the sentiment value will be added and normalized to obtain the compound value [22]. An example is as follows, "The food is nice and good". VADER recognizes "nice" as a sentimental word that has a polarity score of 1.9 and "good" that has a polarity of 1.8. The text now has a compound score of 3.7 and proceed to be normalized and gives a compound score of 0.6907. The normalization formula can be seen in Equation (2), where x is the sum of valence scores of all words and α is the normalization constant which is set to default value of 15.

$$x = \frac{x}{\sqrt{x^2 + \alpha}} \quad (2)$$

2.4. BiLSTM

RNN works well in time-series classification since RNN has a recursive formulation that allows Recurrent Neural Networks (RNN) to handle variable-length sequences [23], [24]. However, RNNs are known to be difficult to train due to the vanishing gradient and exploding gradient [25], [26] proposed a new method of an extended RNN named as Long Short-Term Memory, which solves the problem of RNN. LSTM has three types of gates, namely input gate, output gate, and forget gate. The formulas of LSTM are expressed in Equations (3-7) [27], where i_t, f_t, o_t are respectively the input gate, forget gate, output gate. C_t, h_t calculates the cell state and cell output. σ is the non-linier sigmoid function and tanh is the activation. $W_f,\ W_i,\ W_o,\ W_c$ denotes the weight matrices, and $b_i,\ b_f,\ b_o,\ b_C$ denotes the bias vectors.

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \quad (3)$$

$$\mathrm{f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f)} \quad (4)$$

$$\mathrm{o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o)} \quad (5)$$

$$\mathrm{C_t = f_t * c_{t-1} + i_t * \tanh(W_{Cx}x_t + W_{Ch}h_{t-1} + b_C)} \quad (6)$$

$$\mathrm{h_t = o_t * \tanh\ (c_t)} \quad (7)$$

BiLSTM is consisted of two LSTMs, the forward LSTM and backward LSTM. In forecasting task, it is beneficial to not only have the input features, but also the subsequent inputs, and BiLSTM is able to perform this task. The previous feature will be captured by the forward LSTM and the subsequent feature will be extracted by backward LSTM. This way, BiLSTM will elevate the forecasting accuracy. The BiLSTM structure is illustrated in Figure 1.

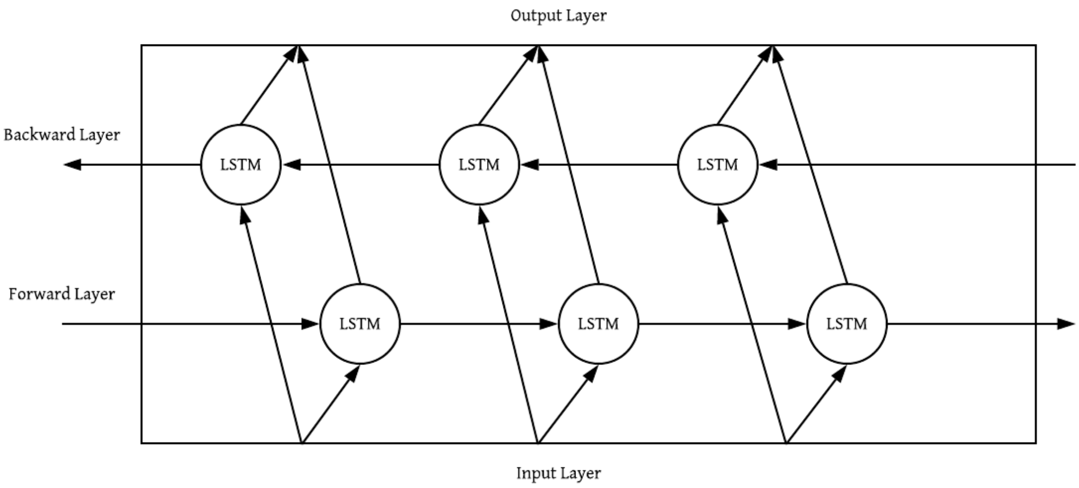

Figure 1. Illustration of how BiLSTM works

2.5. Attention Layer

Attention layer has become more popular in deep learning. Attention layer has also become successfully applied in text classification and helped in improving accuracy and enhancing BiLSTM model [28], [29]. The main idea of attention layer is this layer will only use parts of given input with the most relevant information rather than using an entire input. In this research, we also applied attention layer to our stock prediction model. The equations of Attention layer are shown in Equations (8-9) [30]. w_a denotes as the weight matrix of attention mechanism. e_t denotes the result of the weight calculation and a_t is the final weight. b is the deviation of attention and $[x_1, x_2, \dots, x_T]$ denotes the input of attention.

$$e_t = \tanh\ (w_a[x_1, x_2, \dots, x_T] + b) \quad (8)$$

$$a_t = \frac{\exp\ (e_t)}{\sum_{k=1}^{T} \exp\ (e_k)} \quad (9)$$

2.6. Evaluation

The regression problem usually only has one output which is the predicting value because this regression method is used to predict an arbitrary real number [31]. One of loss function that is commonly used in regression problems is Root Mean Square Error (RMSE). RMSE measures the performance of a model by comparing predicted values and the actual values. When the RMSE value is getting smaller, that shows that the model has better performance. The RMSE formulation can be seen on Equation (10), where y_i is the actual values and y'_i is the predicted values.

$$RMSE_{(y,y')} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y'_i)^2}{n}} \quad (10)$$

3. RESULT AND DISCUSSION

3.1. Collected Data

On the Twitter dataset, we collected tweet date, tweet's favourite counts, and the tweets. The keywords used to retrieve all the tweets related to Netflix stocks are $NFLX and #NFLX. In this research, we set the range date to 5 years, ranging from August 21 2015, to August 21 2020. Each day, we take 50 latest related tweets of the day as the representation of the public sentiment. Table 1 shows an example of collected tweets.

Table 1. Example of Collected Tweets

TWEETS	DATE	FAVORITE
Biggest mistakes of the century: From before, Apple $AAPL is only a PC maker Amazon $AMZN is only a bookstore Google $GOOG $GOOGL is only a search engine Netflix $NFLX is only a DVD rental.	2020-04-11 17:30:07+00:00	2005
Keep in mind, over the past 20 yrs, @PeterSchiff never recommended $AAPl $AMZN $MSFT $GOOG $FB $NFLX or any other top performer over the past two decades. He missed #Bitcoin at $1 $10 $100 when I repeatedly told him to buy it Why does anyone listen to him?	2020-06-21 22:12:36+00:00	895

Netflix daily stock price data was downloaded from Yahoo! Finance. The downloaded data contains 'Date', 'Open', 'High', 'Low', 'Close', 'Adj Close' or adjusted close, and 'Volume'. Netflix stock price is also collected for the same 5 years time frame. Figure 2 show the stock price data on the open price value, where y-axis refletcs values in dollar unit. Table 2 shows the example of daily stock price data that we collected.

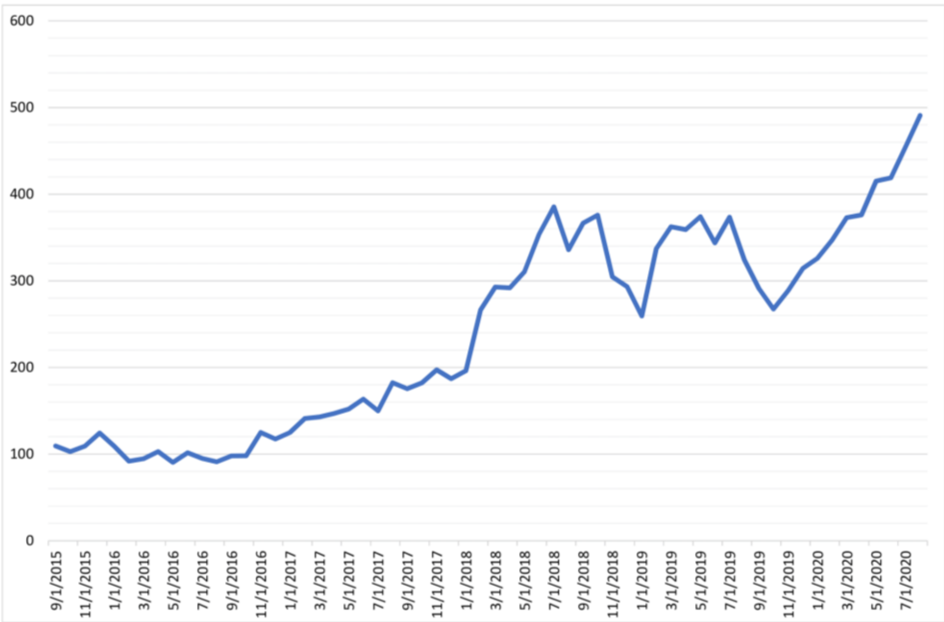

Figure 2. Netflix stock open price from 21 August 2015 to 21 August 2020

Table 2. Example of Daily Stock Price Data

Date	Open	High	Low	Close	Adj Close	Volume
8/21/2015	106.2	110	102.75	103.96	103.96	33228300
8/24/2015	88.75	109.63	85.5	96.88	96.88	59951900
8/25/2015	107.7	107.88	101.5	101.52	101.52	37620700

3.2. Prepared Data

The data preparation process is done following the steps in Section 2.2. Especially for the favourite count addition, we would like to give one example. Addition of 1 is created to avoid the zero multiplication and can be included as the representation of the tweet sender likes on their own tweet. A tweet of "Game of Thrones mania impacts Netflix views" has 2033 favourite count. So, we add 1 value on the favourite count, and the sentiment compound value is multiplied by 2034.

We combined tweets data and stock price data by grouping them by date. Our combined data have missing values since stock price does not record stock prices every day. To fill the missing values in open, close, high, and low price, we use average on the previous data value and next value from the current missing value. For example, when the previous day of close price is 102 and the next day close price is 104, we get 103 to replace the missing value. The example of combined tweets and stock price data can be seen in Table 3.

Table 3. Example of Combined Tweets and Stock Price Data

Date	Compound	$Compound \times Favorite$	Open	Close	High	Low
08/21/2015	6.5792	11.0231	106.199	103.959	110.000	102.750
08/22/2015	11.0457	14.5550	97.474	100.419	109.814	94.125
08/23/2015	6.8788	9.5826	97.474	100.419	109.814	94.125
08/24/2015	8.4287	19.4569	88.750	96.879	109.629	85.500

Using 20-30% of the dataset as test set and 70%-80% of the dataset as train set can give the best result [32], [33]. In our experiment, we divided the data into 3 sets, namely test set, train set, and validation set. Test set used 20% of total data, train set used 80% of the total data, and 20% of the train set is used as validation set. We compare different time window size to see which one has the best prediction result. The time window size used are 5 years, 4 years, 3 years, 2 years and 1 year with each size divided into 3 sets. To note that the window size is retracted from the most recent data. The 5-year dataset has the train set ranged from 21st August 2015 to 21st August 2019, while the test set is ranged from 22nd August 2019 to 21st August 2020. The 4-year dataset has train set ranged from 21st September 2016 to 4th August 2020, and the test set ranged from 5th August 2020 to 21st August 2020. The 3-year dataset has the train set ranged from 21st September 2017 to 21st January 2020, and the test set ranged from 22nd January 2020 to 21st August 2020. The 2-year dataset has the train set ranged from 21st September 2018 to 3rd April 2020, and the test set ranged from 4th April 2020 to 21st August 2020. The 1-year datset has the train set ranged from 21st September 2019 to 11th June 2020 and the test set ranged from 12th June 2020 to 21st August 2020. Our stock prediction experiment results are discussed in Section 3.6.

3.3. Sentiment Extraction Result

To perform sentiment extraction, we used VADER algorithm. VADER returned 4 values on each tweet, namely the positive value, negative value, neutral value, and compound value. After all the tweets have their own sentiment value, the compound value of each tweet will be multiplied by the number of favourites count. The results of sentiment extraction will then be used along the selected stock price parameters as a vector and fed to the BiLSTM architecture to predict the future stock market. The example of sentiment extraction results after we summed the values on each date can be seen in Table 4.

Table 4. The Example of Sentiment Extraction Results Grouped by Date

Date	Compound	Positive	Negative	Neutral	$Compound \times Favorite$
08/21/2015	6.5792	5.677	1.824	42.500	11.0231
08/22/2015	11.0457	8.054	0.748	41.198	14.5550
08/23/2015	6.8788	7.371	2.766	36.863	9.5826
08/24/2015	8.4287	5.929	1.011	43.060	19.4569
08/25/2015	5.0953	4.436	1.967	41.594	6.0593

3.4. Stock Prediction Model Architecture

To obtain the best model, different model architectures are initially studied. We investigated BiLSTM and attention layer architectures. Three architecture model of BiLSTM are initially experimented to see which one works better. The first architecture model uses 2 stacked BiLSTM layer, attention layer, and a dropout layer. The second architecture model uses only 1 layer of BiLSTM, attention layer, and a dropout layer. The third architecture model uses 2 stacked BiLSTM layer and a dropout layer. Figure 3 shows that the best architecture model that reached the lowest RMSE of 0.02930 is the first architecture model (BiLSTM Model 1) when using the same batch size: 32, epoch: 180, neuron: 32, and dropout: 0.1.

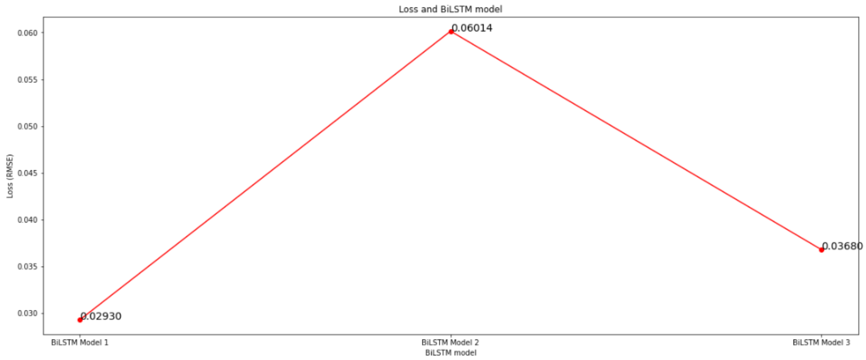

Figure 3. Comparing Loss (RMSE) and BiLSTM Model

The combined BiLSTM + attention layer architecture of BiLSTM Model 1 is illustrated in Figure 4. The model architecture started with an input layer which will pass all the information to the hidden nodes and followed by the first BiLSTM layer to process all the information and forwarded to attention layer which will focus on extracting most valuable information. The result is then passed to the second BiLSTM and accompanied by a dropout layer to reduce overfitting and passed to a dense layer with tanh activation in the last layer. This model is compiled using RMSprop as optimizers and RMSE to measure the loss.

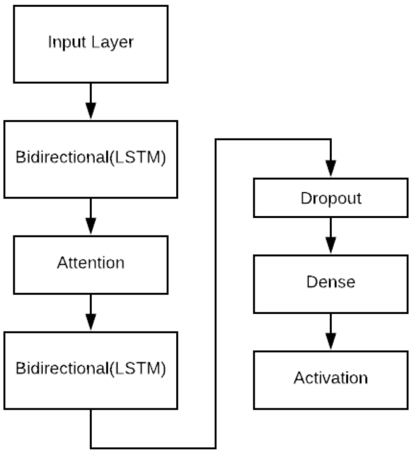

Figure 4. Bidirectional LSTM – Attention Layer

3.5. Hyperparameters Setting

We tune the hyperparameters such as number of epochs, batch size, neurons, and dropout to achieve the best result. By tuning the hyperparameters, it will affect the model and help us find the best model to predict the stock price. We did an initial experiment to select the best hyperparameter. The base hyperparameter used in the initial experiments are batch size: 32, epoch: 180, neuron: 32, and dropout: 0.1. Table 5 gives us the final chosen hyperparameter. The breakdown of the hyperparameter selection process is as follows.

Table 5. Hyperparameters

Parameter	Value	Description
batch_size	16	Number of data points used in one mini-batch
nb_epoch	170	Number of epochs used in training the model
neurons	64	Number of neurons used in one LSTM layer
dropout	0.1	Dropouts to prevent overfitting

Figure 5 shows how the number of epochs affect the RMSE value. To note that the RMSE value shown is produced with normalizing the dataset value (as explained in Section 2). Choosing the appropriate number of epochs can make or break deep learning model. A small number of epochs can lead to under-fitting and using many epochs will cause overfitting. From Figure 5, the RMSE reached the lowest value of 0.03892 when the epoch is 170. Batch size allows training deep learning models using fewer samples. Samples in the batch are introduced to model to recognize the features and patterns of data. The size of batch strongly impacts the accuracy, or the time taken for the model to reach convergence. From Figure 6, the batch size of 16 has the lowest RMSE of 0.03253. Neurons are used to process complex and challenging problems. The number of neurons used will affect the performance of the model. Neuron will receive the input signal and forward it to the next neuron. Figure 7 shows a neuron of 64 has RMSE of 0.03096. Figure 8 shows dropout of 0.1 has lowest RMSE of 0.03093. Dropout is a regularization used to reduce over-fitting. Dropout layer is usually placed between existing layers; it will ignore random neurons when training deep learning model. Using dropout will decrease the model loss. Dropout values are between 0 to 1.

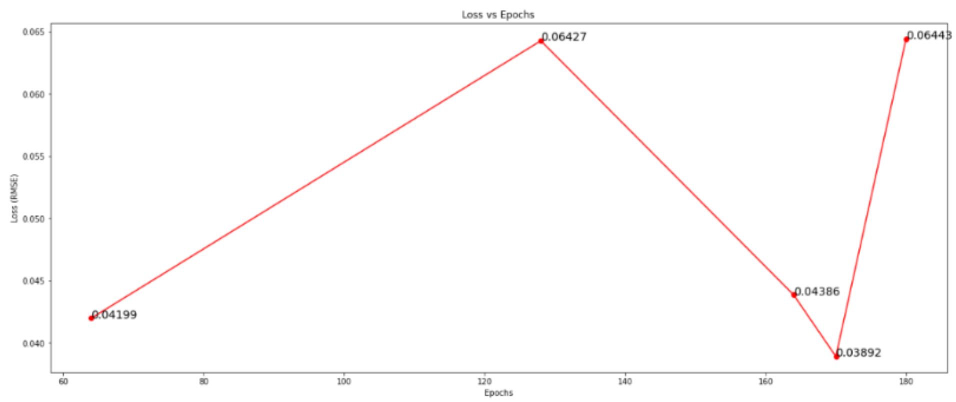

Figure 5. Comparing Loss (RMSE) and Epochs

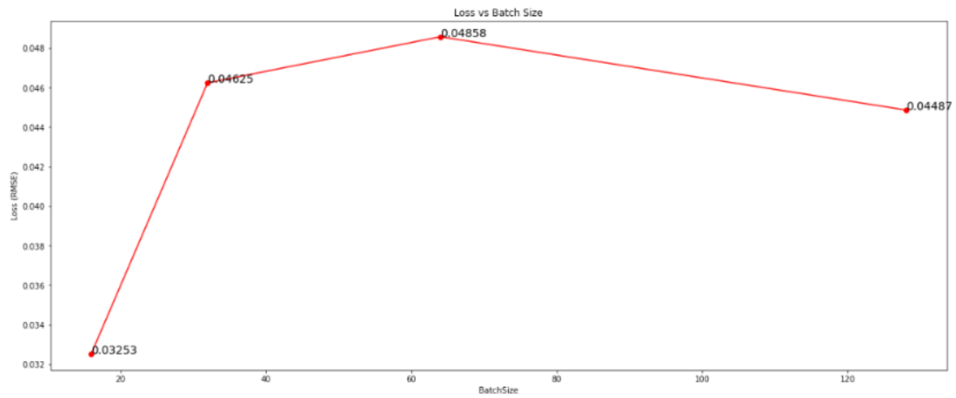

Figure 6. Comparing Loss (RMSE) and Batch Size

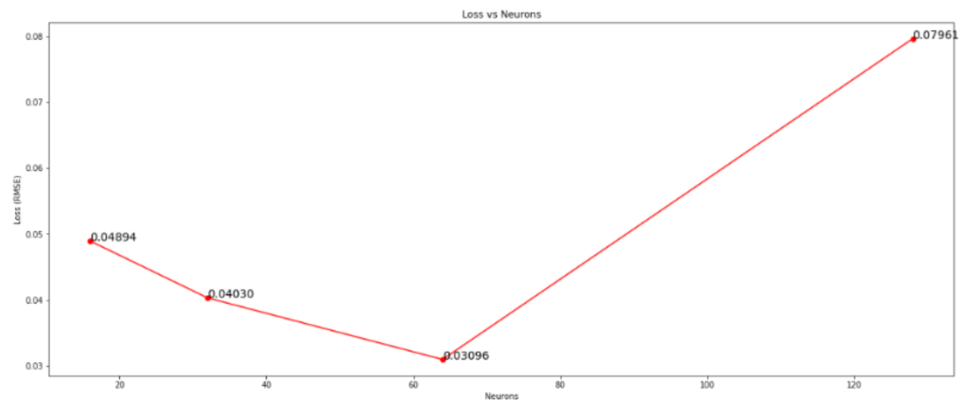

Figure 7. Comparing Loss (RMSE) and Neurons

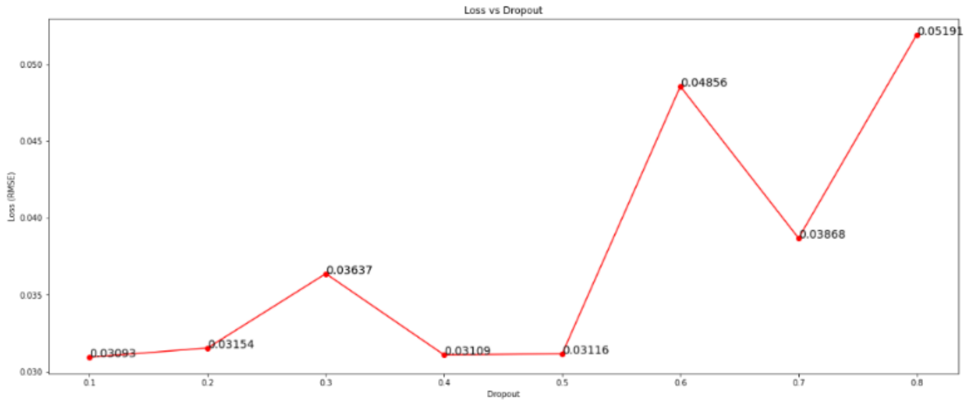

Figure 8. Comparing Loss (RMSE) and Dropout

3.6. Stock Prediction Result

We divide our experiment into two parts. The first part is to find the best model based on different data parameters and model architectures using the same dataser. The second part is to use the best setup in the first part into different dataset based on the time window frame.

The first part of the experiment used the 5-year dataset (see Section 3.2) and has seven different experiments with different data parameter and model architecture. As seen in Table 6, the best result (the lowest RMSE) is obtained by the model that use close price, open price, and multiplied compound value as parameters. By using the said parameters using BiLSTM + attention layer architecture gives better result againts BiLSTM layer only architecture. The combination of the parameters and the architecture (Exp 3 in Table 6) resulted in RMSE score 2.981e-02 on the test set (To note once again that the RMSE value shown in the paper is produced after normalizing the dataset as explained in Section 2). The value reflects the original RMSE value (without normalizing the actual stock and predicted stock data) of the Exp 3 model, which is 9.43 on the test set. Figure 9 depicted the visualization of the stock price prediction comparison in detail.

With model from Exp 3 being the best model that reaches the lowest RMSE using 5-year dataset, we experimented by comparing different kind of dataset from 1-year to 5-years dataset using model in Exp 3. Table 7 shows the RMSE obtained from different dataset with the different time window size. As shown in Table 7, the lowest RMSE is achieved by using the 5-years dataset.

Table 6. Stock Prediction Experiment Results

Model	**Exp**	**Stock Price Parameters**				**Sentiment Parameter**	**RMSE**	
		Close	**Open**	**High**	**Low**		**Train**	**Test**
BiLSTM + Attention	*1*	√	√	√	√	*Compound × Favorite*	4.126e-02	4.877e-02
	2	√	√	√	√	Compound	2.552e-02	7.556e-02
	3	√	√	-	-	***Compound × Favorite***	**2.482e-02**	**2.981e-02**
	4	√	√	-	-	Compound	2.867e-02	3.785e-02
	5	√	√	-	-	-	3.033e-02	4.701e-02
	6	√	√	√	√	-	2.867e-02	5.214e-02
BiLSTM	*7*	√	√	-	-	*Compound × Favorite*	3.018e-02	6.462e-02

Table 7. RMSE Value of Model in Exp 3 on Different Datasets

1-year		**2-years**		**3-years**		**4-years**		**5-years**	
Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
5.840e-02	3.079e-01	3.370e-02	1.557e-01	2.720e-02	8.873e-02	2.605e-02	5.908e-02	**2.482e-02**	**2.981e-02**

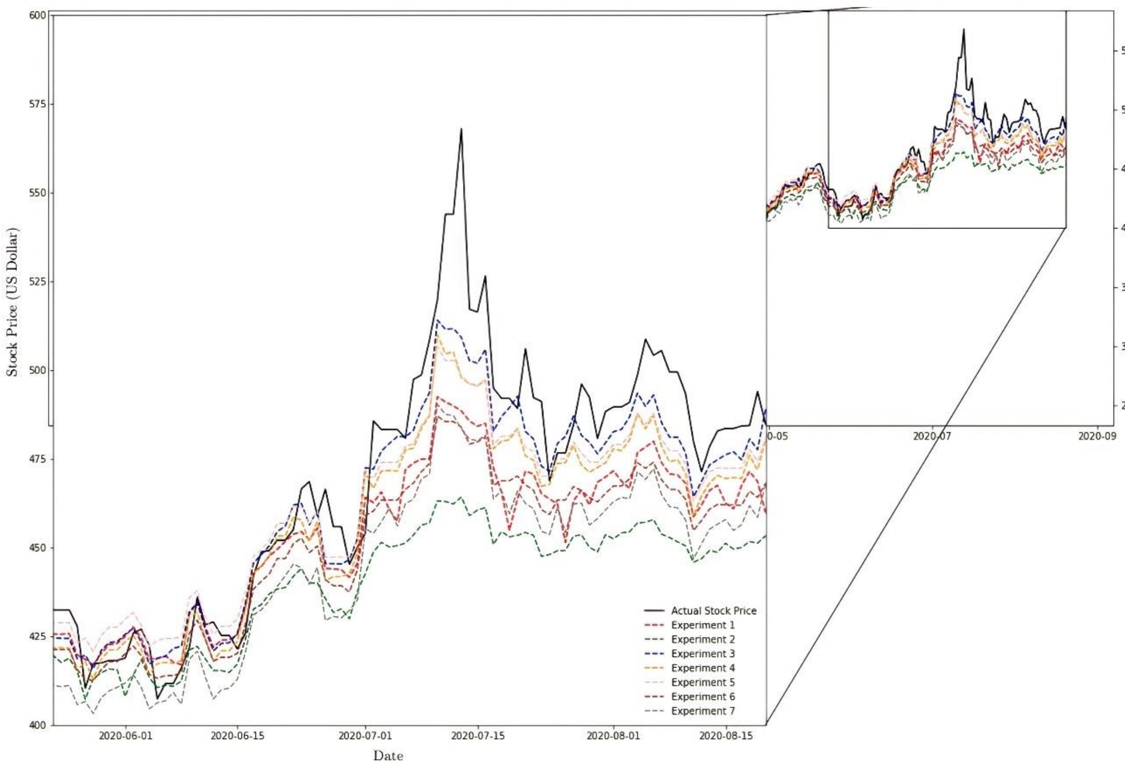

Figure 9. Detailed predicted stock price comparison

To highlight the performance of out model, we compared our model performace with other stock price prediction studies. On Table 8, we show a comparison of RMSE evaluation of various prediction models on different datasets and studies. The compared studies has different approaches and datasets. [30] implemented a Wavelet LSTM and attention model without utilizing sentiment on various dataset such as DJIA, S&P 500 dataset, and HIS dataset. [34] uses an LSTM model and applied dimensionality reduction PCA on various dataset such as Apple Inc, Amerisource Bergen Corporation, and Cardinal Health datasets. [35] proposed a hybrid LSTM and Artificial Bee Colony (ABC) algorithm on various datasets and incorporate tweets sentiment polarity using lexicon-based methods. This research shows that by incorporating sentiment polarity helps in improving the model to predict future stock price. [36] also uses LSTM that incorporated twitter coronavirus sentiment as feature to predict stock price. In this paper, we implemented BiLSTM-Attention on Netflix dataset using VADER to extract tweet sentiment scores. From Table 8, we can see that our model's performance is comparable yet achieves better result compared to other models.

Table 8. Comparison of RMSE Evaluation in Predicting Stock Prices

Model	Sentiment	Dataset	RMSE
Wavelet LSTM – Attention [30]	-	DJIA Dataset	1.97e-01
LSTM-PCA [34]	-	Apple Inc.	6.43e-02
ABC-LSTM [35]	√	Apple Inc.	7.35e-00
LSTM [36]	√	Cisco Inc.	6.54e-02
Ours (BiLSTM-Attention)	√	Netflix Inc.	**2.98e-02**

4. CONCLUSION

Improving economic development by building a good stock price prediction model is one of the backgrounds of this research. Public opinion has an influence on the rise and fall of stock prices. Therefore, we build a predictive model by including public sentiment as one of the factors. We use Netflix company stock as the case study. We used tweets that are scraped from Twitter to perform our sentiment analysis. VADER algorithm is used to determine the degree of the sentiment. Through our experiment, we aim to select the best configuration and parameter to get the most accurate model. There are seven different experiments with different parameters of stock price data, sentiment values, and learning architectures. Different combination of BiLSTM and attention layers with a different window size is found in the experiment. We found that the combination of close price, open price in stock price parameters, and multiplied compound value in sentiment parameters are the best parameter for stock prediction. In this research, we also provide the evidence that by combining Bidirectional Long Short-Term Memory (BiLSTM) and attention layer with a window size of 5 years data in the training process, the prediction model can get the smallest RMSE score of 2.482e-02 on train set and 2.981e-02 on test set. Furthermore, in our evaluation, our RMSE score is comparable and even lower, compared to another research.

ACKNOWLEDGEMENTS

This research was funded by the DIKTI Research Grant no 081/SP2H/AMD/LT/DRPM/2020.

REFERENCES

[1] J. Zhang, S. Cui, Y. Xu, Q. Li, and T. Li, "A novel data-driven stock price trend prediction system," *Expert Syst. Appl.*, vol. 97, pp. 60–69, 2018, doi: 10.1016/j.eswa.2017.12.026.

[2] D. Shah, H. Isah, and F. Zulkernine, "Predicting the Effects of News Sentiments on the Stock Market," *Proc. - 2018 IEEE Int. Conf. Big Data, Big Data 2018*, pp. 4705–4708, 2019, doi: 10.1109/BigData.2018.8621884.

[3] F. Rundo, F. Trenta, A. L. Di Stallo, and S. Battiato, "Advanced Markov-based machine learning framework for making adaptive trading system," *Computation*, vol. 7, no. 1, 2019, doi: 10.3390/computation7010004.

[4] S. Rick and G. Loewenstein, "The role of emotion in economic behavior Consequentialist models of decision making," *Handb. Emot.*, pp. 138–156, 2008, doi: 10.2139/ssrn.954862.

[5] J. Griffith, M. Najand, and J. Shen, "Emotions in the Stock Market," *J. Behav. Financ.*, vol. 21, no. 1, pp. 42–56, 2020, doi: 10.1080/15427560.2019.1588275.

[6] J. Shen, M. Najand, F. Dong, and W. He, "News and social media emotions in the commodity market," *Rev. Behav. Financ.*, vol. 9, no. 2, pp. 148–168, 2017, doi: 10.1108/RBF-09-2016-0060.

[7] Z. Wang, S. B. Ho, and Z. Lin, "Stock market prediction analysis by incorporating social and news opinion and sentiment," *IEEE Int. Conf. Data Min. Work. ICDMW*, vol. 2018-Novem, pp. 1375–1380, 2019, doi: 10.1109/ICDMW.2018.00195.

[8] J. Gupta, A. Jain, and Y. Bohra, "SENTIMENTAL ANALYSIS ON NEWS DATA FOR STOCK MARKET," no. 6, pp. 84–86, 2018.

[9] W. Budiharto and M. Meiliana, "Prediction and analysis of Indonesia Presidential election from Twitter using sentiment analysis," *J. Big Data*, vol. 5, no. 1, pp. 1–10, 2018, doi: 10.1186/s40537-018-0164-1.

[10] A. Goel and A. Mittal, "Stock prediction using twitter sentiment analysis. Standford University, CS229," *Cs229.Stanford.Edu*, no. December, pp. 1–5, 2012, [Online]. Available: http://cs229.stanford.edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis.pdf.

[11] A. Kirlić, Z. Orhan, A. Hasovic, and M. Kevser-Gokgol, "Stock prediction using twitter sentiment analysis," *Int. J. Psychosoc. Rehabil.*, vol. 24, no. 8, pp. 1031–1035, 2018, doi: 10.37200/IJPR/V24I8/PR280113.

[12] K. Joshi, B. H. N., and P. J. Rao, "S TOCK TREND PREDICTION USING NEWS SENTIMENT ANALYSIS," 2016.

[13] V. S. Pagolu, K. N. Reddy, G. Panda, and B. Majhi, "Sentiment analysis of Twitter data for predicting stock market movements," *Int. Conf. Signal Process. Commun. Power Embed. Syst. SCOPES 2016 - Proc.*, pp. 1345–1350, 2017, doi: 10.1109/SCOPES.2016.7955659.

[14] J. Smailović, M. Grčar, N. Lavrač, and M. Žnidaršič, "Predictive sentiment analysis of tweets: A stock market application," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7947 LNCS, pp. 77–88, 2013, doi: 10.1007/978-3-642-39146-0_8.

[15] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A Comparative Analysis of Forecasting Financial Time Series Using ARIMA, LSTM, and BiLSTM," 2019, [Online]. Available: http://arxiv.org/abs/1911.09512.

[16] H. Li, Y. Shen, and Y. Zhu, "Stock Price Prediction Using Attention-based Multi-Input LSTM," *Proc. Mach. Learn. Res.*, vol. 95, no. 2017, pp. 454–469, 2018.

[17] S. Mehtab, J. Sen, and A. Dutta, "Stock Price Prediction Using Machine Learning and LSTM-Based Deep Learning Models," no. December, Sep. 2020, [Online]. Available: https://arxiv.org/abs/2009.10819.

[18] S. Khan and H. Alghulaiakh, "ARIMA model for accurate time series stocks forecasting," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 7, pp. 524–528, 2020, doi: 10.14569/IJACSA.2020.0110765.

[19] T. Kabbani and F. E. Usta, "Predicting The Stock Trend Using News Sentiment Analysis and Technical Indicators in Spark," *arXiv Prepr. arXiv2201.12283*, pp. 1–4, 2022, [Online]. Available: https://arxiv.org/abs/2201.12283%0Ahttps://arxiv.org/pdf/2201.12283.

[20] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification," *Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020*, no. Icssit, pp. 729–735, 2020, doi: 10.1109/ICSSIT48917.2020.9214160.

[21] S. G. K. Patro and K. K. sahu, "Normalization: A Preprocessing Stage," *Iarjset*, no. March, pp. 20–22, 2015, doi: 10.17148/iarjset.2015.2305.

[22] R. Adarsh, A. Patil, S. Rayar, and K. M. Veena, "Comparison of VADER and LSTM for sentiment analysis," *Int. J. Recent Technol. Eng.*, vol. 7, no. 6, pp. 540–543, 2019.

[23] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent Neural Networks for Multivariate Time Series with Missing Values," *Sci. Rep.*, vol. 8, no. 1, pp. 1–12, 2018, doi: 10.1038/s41598-018-24271-9.

[24] A. Murad and J. Y. Pyun, "Deep recurrent neural networks for human activity recognition," *Sensors (Switzerland)*, vol. 17, no. 11, 2017, doi: 10.3390/s17112556.

[25] Y. Bengio, N. Boulanger-Lewandowski, and R. Pascanu, “Advances in optimizing recurrent networks,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 8624–8628, 2012, doi: 10.1109/ICASSP.2013.6639349.
[26] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
[27] Q. Chen, W. Zhang, and Y. Lou, “Forecasting Stock Prices Using a Hybrid Deep Learning Model Integrating Attention Mechanism, Multi-Layer Perceptron, and Bidirectional Long-Short Term Memory Neural Network,” *IEEE Access*, vol. 8, pp. 117365–117376, 2020, doi: 10.1109/ACCESS.2020.3004284.
[28] G. Liu and J. Guo, “Bidirectional LSTM with attention mechanism and convolutional layer for text classification,” *Neurocomputing*, vol. 337, pp. 325–338, 2019, doi: 10.1016/j.neucom.2019.01.078.
[29] G. Ramet, P. N. Garner, M. Baeriswyl, A. Lazaridis, E. Polytechnique, and F. De Lausanne, “CONTEXT-AWARE ATTENTION MECHANISM FOR SPEECH EMOTION RECOGNITION Idiap Research Institute , Martigny , Switzerland Artificial Intelligence and Machine Learning Group , Swisscom,” *2018 IEEE Spok. Lang. Technol. Work.*, no. 3, pp. 126–131, 2018, [Online]. Available: http://publications.idiap.ch/downloads/papers/2018/Ramet_SLT_2018.pdf.
[30] J. Qiu, B. Wang, and C. Zhou, “Forecasting stock prices with long-short term memory neural network based on attention mechanism,” *PLoS One*, vol. 15, no. 1, pp. 1–15, 2020, doi: 10.1371/journal.pone.0227222.
[31] G. Ding and L. Qin, “Study on the prediction of stock price based on the associated network model of LSTM,” *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 6, pp. 1307–1317, 2019, doi: 10.1007/s13042-019-01041-1.
[32] A. Gholamy, V. Kreinovich, and O. Kosheleva, “Why 70 / 30 or 80 / 20 Relation Between Training and Testing Sets : A Pedagogical Explanation,” pp. 1–6, 2018.
[33] S. Raschka, “Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning,” 2018.
[34] J. Rasheed, A. Jamil, A. Ali Hameed, M. Ilyas, A. Ozyavas, and N. Ajlouni, “Improving Stock Prediction Accuracy Using CNN and LSTM,” *2020 Int. Conf. Data Anal. Bus. Ind. W. Towar. a Sustain. Econ. ICDABI 2020*, 2020, doi: 10.1109/ICDABI51230.2020.9325597.
[35] R. Kumar, P. Kumar, and Y. Kumar, “Integrating big data driven sentiments polarity and ABC-optimized LSTM for time series forecasting,” *Multimed. Tools Appl.*, 2021, doi: 10.1007/s11042-021-11029-1.
[36] A. Jabeen *et al.*, “An LSTM based forecasting for major stock sectors using COVID sentiment,” *Comput. Mater. Contin.*, vol. 67, no. 1, 2021, doi: 10.32604/cmc.2021.014598.