# Evaluation of Stock Closing Prices using Transformer Learning

**Tariq Saeed Mian**
Department of IS, College of Computer Science & Engineering, Taibah University, Saudi Arabia
tmian@taibahu.edu.sa (corresponding author)

*Received: 7 May 2023 | Revised: 2 June 2023, 12 July 2023, and 18 July 2023 | Accepted: 19 July 2023*

 *DOI: https://doi.org/10.48084/etasr.6017*

## ABSTRACT

**Predicting stock markets remains a critical and challenging task due to many factors, such as the enormous volume of generated price data, instant price data changes, and sensitivity to human sentiments, wars, and natural disasters. Since the previous three years of the COVID-19 pandemic, forecasting stock markets is more difficult, complex, and problematic for stock market analysts. However, technical analysts of the stock market and academic researchers are continuously trying to develop innovative and modern methods for forecasting stock market prices, using statistical techniques, machine learning, and deep learning-based algorithms. This study investigated a Transformer sequential-based approach to forecast the closing price for the next day. Ten sliding window timesteps were used to forecast next-day stock closing prices. This study aimed to investigate reliable techniques based on stock input features. The proposed Transformer-based method was compared with ARIMA, Long-Short Term Memory (LSTM), and Random Forest (RF) algorithms, showing its outstanding results on Yahoo Finance data, Facebook Intra data, and JPMorgan's Intra data. Each model was evaluated using Mean Absolute Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE).**

***Keywords-stock prediction; ARIMA; SARIMA; LSTM; transformer; stock volatility; stock market; stock market prediction; machine learning; deep learning***

## I. INTRODUCTION

The stock market has great potential for profitmaking [1]. Different techniques and studies have been presented to predict future prices, maximize the returns of investments, and minimize the corresponding risks [2]. However, there are still risks due to unpredictable fluctuations in stock market prices, complicated trading circumstances in practice [3], and the difficulty in interpreting price movement with domain knowledge. Various statistical techniques with good interpretation, such as auto-regressive [4], tree [5], and hidden Markov [6] models, have been introduced for this purpose. Powerful prediction models [7] are expensive to train but provide good results. The importance of deep learning models cannot be ignored in powerful prediction models for sequential data. Deep learning models are receiving more research interest due to the increasing number of available datasets, growing computational power, and long-term dependency capability [8-9]. Recurrent Neural Network (RNN)-based deep learning models such as RNN, LSTM, and Transformer have shown state-of-the-art results in time-series tasks. Analyzing historical stock market patterns can provide valuable insight in predicting stock values, optimizing gain, and minimizing losses [10-11]. Intercorrelation among different stocks is a crucial factor in estimating future stock prices. All tradable stocks in today's markets are interconnected through various attributes. Although correlation does not imply causation, strong Pearson's correlation scores suggest a potential relationship between correlated stocks. Therefore, investigating stock price trends and forecasting future changes remains a popular research field.

This study aims to develop an advanced model for short-term price forecasting, considering factors such as the global economy, international politics, financial performance, stakeholder expectations, and financial reports. Traditional investors face difficulties in predicting market behavior and selecting effective forecasting techniques to maximize profits and minimize losses. Predicting future stock prices is a challenging task for investors, companies, and financial organizations [12]. Variables such as the national political environment, economic situation, and investor psychology are crucial to accurate predictions. Analysts in various fields face numerous challenges in the forecasting of financial markets [13]. Stock market predictions are heavily based on factors such as intrinsic value, financial performance, regulatory requirements, GDP, natural disasters, and other variables [14].

Machine learning algorithms have transformed stock market forecasting by leveraging large and nonlinear datasets. These methods have shown significant improvements over traditional approaches, ranging from 60% to 86% [15]. Long-term investment in established stocks is considered more profitable and requires less effort and time compared to other investments like bonds. However, short-term trading aims to capitalize on small fluctuations in stock values and requires substantial time and effort from traders, particularly when done manually. In this case, staying up-to-date with the latest stock news is essential. Time-series stock market prediction models using statistical and machine learning methods help traders and investors make informed decisions based on historical data and market trends. However, these predictions are not always accurate, and investing in the stock market carries inherent risks. Traders and investors must employ well-informed strategies, risk management plans, and a deep understanding of the market to mitigate these risks. It is crucial to acknowledge that relying solely on stock market predictions does not guarantee success in trading or investing. By incorporating these considerations, investors can increase the likelihood of positive returns while minimizing the risk of loss. In light of the contributions of statistical, machine learning, and deep learning methods, this study aims to investigate two specific research questions:

RQ1: Do machine learning techniques outperform the statistical techniques when applied to time series data?

RQ2: Do deep learning models effectively address sequential data challenges and yield superior outcomes when compared to machine learning algorithms?

Previous studies used statistical time series methods such as moving averages and autoregressive techniques for stock market price forecasting. This study examined the efficacy of the Transformer deep learning model in predicting closing stock prices, leveraging its ability to handle long-term dependencies and address the vanishing gradient problem. This study aims to make the following contributions:

1. Present a novel Transformer-based framework for predicting stock prices of real indices, such as Yahoo, JPMorgan, and Facebook, using their intraday data.
2. Present a comparative analysis of random forest, LSTM, and Transformer-based models on three benchmark datasets.
3. Assess the prevalent machine and deep learning models that can benefit financial analysis.
4. Evaluate the effectiveness of proposed techniques using Root Mean Squared Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Percentage Error (MAPE).

Machine learning and deep learning algorithms revolutionized stock forecasting using historical data. In [16], the influence between different machine learning algorithms and multi-feature methods in time series was studied. In [17], an algorithm was presented that combined artificial neural networks and genetic algorithms. In [18], a look-back period was used for accurate forecasting. In [19], an RNN was used with news data to achieve better performance than ARIMA. In [20], LSTM was used for growth calculation. In [21], the SARIMA and BPNN models were compared on the Korean stock exchange. Previous studies have shown that the SARIMA model outperforms the BPNN and KOSPI models. The KOSPI model excels at predicting volatile and nonlinear data. Prediction accuracy depends on the evolution of the model. In [22], regression techniques were used with ARIMA for stock market prediction. In [23], accurate stock price prediction was achieved in banking-related markets using ARIMA. In [24], support vector machines, MLP, and logistic regression with technical indicators were used for NIFTY50 index prediction. In [25], a rigorous selection process was used to identify dynamic stocks on the Dar es Salaam Stock Exchange, using the LSTM and GRU deep learning models to forecast closing prices. The LSTM model outperformed GRU with a lower RMSE of 4.7524 and an MAE of 2.4377. In [26], a novel model was introduced combining LBL and RNN to capture short- and long-term sentiment patterns.

Deep learning algorithms are used in computer vision, video games, large data, and multimedia [27-28]. RNNs are commonly used for time series prediction due to their ability to capture past data [29-30]. In [31], Bayesian RNNs were implemented using variational inference and measuring the ambiguity of the prediction. CNNs are effective in forecasting time sequences, such as wind power [32] and precipitation [33] predictions. CNN and LSTM were combined in an ensemble approach for air pollution quality forecasting [34-35]. LSTM networks were used to predict river discharge levels [36], and forecast stock market trends [37-38]. CNNs were also suggested for market price forecasting [39]. The MFNN model [40] combined convolution and recursive neurons for feature extraction in fiscal time-series prediction. LSTM was also used to extract data and forecast stock values with improved performance [41]. Deep learning models have gained attention in various domains, including estimating future values. For example, in [42], a deep learning method was used to forecast short-term electricity demand.

## II. METHODOLOGY

To achieve accurate and timely results in stock market forecasting, it is necessary to synchronize historical data with streaming data and train machine learning models accordingly. Numerous techniques have been proposed using machine learning for this purpose. This study used a regression-based model and an LSTM-based network model. Such models follow a specific structural design, as shown in Figure 1, which includes the following processes:

- Data Collection Stage: Stock data are collected to acquire the necessary dataset for analysis.
- Data Preprocessing Stage: The collected data are subjected to preprocessing techniques to be transformed into a suitable format for further analysis. This stage involves tasks such as data cleaning, handling missing values, and normalizing.
- Modeling Stage: At this stage, a stock forecast model is developed using the preprocessed data as input. Data analysis techniques are used to accurately predict the closing values of stocks.
- Performance Evaluation Stage: The model's results are evaluated by comparing them with the actual results to assess its validity. If the model fails to produce the desired results, further refinement of the model or data preprocessing techniques may be required. Otherwise, the results are presented to the user.

- Results Presentation Stage: In the final stage, the model and its predictions are visually presented to the user, allowing for a comprehensive understanding of the results.

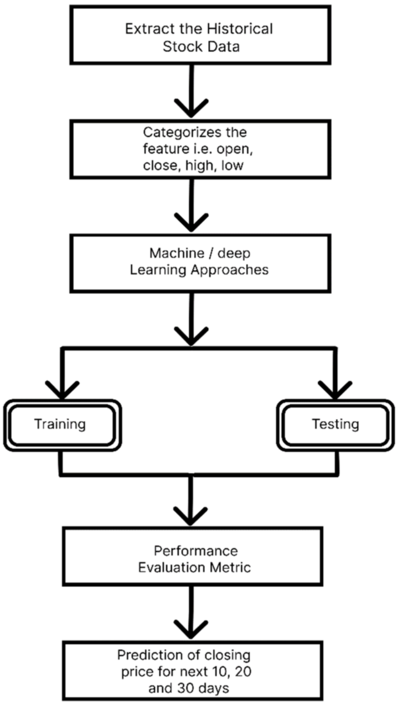


Fig. 1. Time series data analysis flow.

## A. Dataset

This study used three benchmark datasets: Yahoo's [43], Facebook's [44], and JPMorgan's [45] finance data, from January 1, 2017, to September 17. There were no identical or missing values in the dataset. Six features, common in all datasets, were used: High, Low, Open, Close, Noise, and Volume. The primary objective was to forecast the closing price based on past data. The dataset features are described as follows:

- Open price represents the initial price of a stock at the beginning of the market trading session.
- Close price signifies the final price of a stock when the market closes.
- High price indicates the highest value attained by a stock during a trading day.
- Low price denotes the lowest value reached by a stock during a trading day.
- Volume reflects the number of shares or trades executed for a particular stock within a given trading day.

Before conducting a comprehensive analysis and prediction of the closing price, the trends in the closing prices were examined, as shown in Figure 2. It was evident that during the initial phase of the COVID-19 pandemic, the closing prices of the stocks experienced a significant decline, reaching their lowest values.

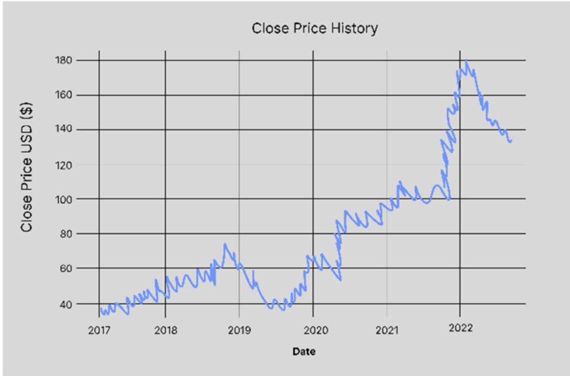


Fig. 2. Closing price history.

## B. Problem Statement

The multivariate time series $X_t$ data at timestamp $t$ were passed to the time series model. This was defined as follows:

$$X_t = \{X_{t-m+1}, X_{t-m+2}, X_{t-m+3}, \dots\dots X_t | X_i \in R^n\} \quad (1)$$

where $m$ is the length of input multivariate time series data and $X_t$ is the multivariate vector at timestamp $t$. A multivariate vector $X_t$ contains $n$ variables for a single time period. The $n$ value depends on the attribute values of the dataset and can be expressed as follows:

$$X_t = \{X_{t(1)}, X_{t(2)}, X_{t(3)}, \dots\dots X_{t(n)} | X_i \in R \quad (2)$$

The multivariate time series input $X_i \in R^{(m\times n)}$ represents the historical multivariate data for a part-time period from $t$-$m$+1 to $t$. In this study, the dataset had multiple input variables to forecast the close prices based on past data. The approach was to predict the multivariate vector

$$X_{t+r} = \{x_{(t+r)(1)}, x_{(t+r)(2)}, \dots\dots x_{(t+r)(n)}\}$$

of timestamp $t$+$r$ with past historical data

$$X_t = \{ X_{t-m+1}, X_{t-m+2}, \dots\dots\dots, X_t \}$$

as input.

## C. Learning Based Approaches

### 1) Transformer

Although the Transformer architecture was originally designed for Natural Language Processing (NLP), this model has also found applications in time-series analysis and computer vision tasks. One of the key elements of Transformer is the attention mechanism, which allows for a targeted focus on specific information. This study used a variant called the sparse-attention Transformer to predict the closing prices of stocks. This variant offers reduced memory requirements, enabling effective handling of longer historical data for more accurate forecasting. Figure 3 illustrates the architecture of the proposed approach.

The Transformer distinguishes itself from previous approaches in time series data and forecasting by employing an attention mechanism [41]. This mechanism enables the Transformer to selectively focus on relevant information in historical data, disregarding irrelevant features. On the contrary, LSTM encounters difficulties in processing long sequences of data due to short-term memory issues. LSTM updates a hidden state with each new input token, resulting in a prolonged sequence of input. However, LSTM encounters challenges in propagating the entire set of long input sequences due to the vanishing gradient problem. As a consequence, earlier tokens in the long input sequence are gradually forgotten. In contrast, Transformer retains indirect connections to all preceding timestamps, facilitating information propagation across significantly longer sequences.

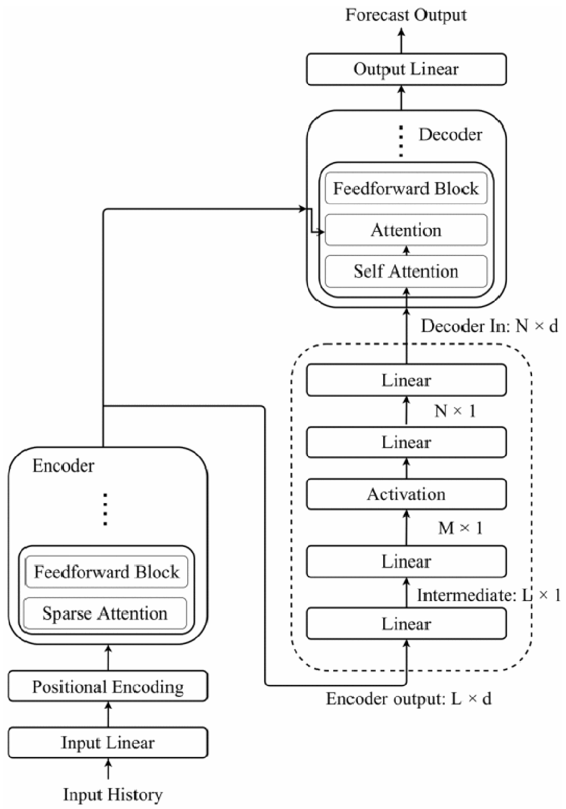


Fig. 3. The Transformer architecture.

The Transformer model consists of two primary components, the encoder and the decoder. The encoder applies dot-product attention iteratively within the input sequence, preserving the sequence length. On the other hand, the decoder generates the output by using dot-product attention between the encoded input and output sequences, with the initial decoder layer containing a placeholder sequence. The proposed method used the full encoder-decoder structure, allowing the decoder to directly generate a forecast sequence of the desired length. The primary components of the Transformer model include the input, encoder, attention mechanism, and decoder.

- Input: The nonsequential processing of historical information poses a significant challenge for the Transformer network. To address this problem, positional encoding [47] is used to provide context on the relative distance between each token and the current timestamp. This information is crucial for determining relevance in the self-attention mechanism. Positional encoding assigns a unique real number to each location, ensuring that it is not duplicated. The positional encoding function *PE*( ) for a given multivariate time series data $X_t$ is illustrated in the following equations:

$$PE(x_i, 2k) = x_i + sin\left(\frac{x_i}{10000^{\frac{2k}{d}}}\right) \quad (3)$$

$$PE(x_i, 2k+1) = x_i + cos\left(\frac{x_i}{10000^{\frac{2k}{d}}}\right) \quad (4)$$

The attention block calculates attention weights by taking the dot product between vectors. The size of the vectors determines the behavior of the attention mechanism. In this scenario, as the stock closing price is represented by a one-dimensional vector, this is addressed by expanding the input to a higher embedding dimension using a linear layer. The output of the linear layer transforms the decoder information into a one-dimensional sequence.

- Encoder: This is a layer that encompasses multiread self-attention along with a two-layer feedforward section.
- Attention: Dot product attention is a commonly used fusion mechanism within transformer models. It derives its name from the fact that attention weights are calculated through dot products of queries and key vectors. Self-attention is used in the encoder, expressed by the following equation [5]:

$$Attention(x) = \sigma\left(\frac{Q(X)(K(X)))^T}{\sqrt{d}}\right)V(X) \quad (5)$$

This equation pertains to self-attention, wherein the query and key vectors are extracted from the input sequences. However, in the decoder attention block, the query vectors are obtained from the output sequence, while the key vectors are derived from the encoder output.

- Decoder: The decoder consists of multiple decoding layers, where each layer incorporates self-attention on the decoder output, and subsequently, attention is applied between the decoder output and the processed sequence obtained from the encoder.

### 2) *Long Short-Term Memory Network (LSTM)*

RNN is a variant of neural networks that incorporates feedback, allowing the modeling of sequential data. The LSTM model is a widely used RNN that excels in various problem domains [45-47]. LSTM units have a memory capability, enabling them to retain information from previous computations, making them suitable for tasks that involve sequential data. Within the LSTM model, gated cells play a pivotal role in regulating information flow within the network [48]. The LSTM architecture consists of input, forget, and output gates. The input gate (fi) controls the information to be updated based on the incoming signal, while the forget gate (ft) determines which states should be retained or forgotten. The output gate (Ot) decides whether the cell state should influence other neurons or not. Activation functions, such as logistic layers, produce values between 0 and 1 in each layer. The LSTM layer generates a new vector that is added to the state, facilitating memory retention and information flow [49]. LSTM networks are useful for examining the impact of one stock's price change on multiple other stocks over time. They can provide insight into the duration for which past stock price patterns should be considered and help predict future trends in stock price variations. LSTM networks offer a powerful approach for analyzing time-series data and extracting meaningful insights for forecasting and trend analysis. The following equations provide a mathematical representation of the flow of information within LSTM:

$$f_t = \sigma(W_{xf}X_t + W_{yf}Y_{t-1} + b_f) \quad (6)$$

$$i_t = \sigma(W_{xi}X_t + W_{yi}Y_{t-1} + b_i) \quad (7)$$

$$g_t = tanh(W_{xg}X_t + W_{yg}Y_{t-1} + b_g) \quad (8)$$

$$s_t = f_t * s_{t-1} + i_t * g_t \quad (9)$$

$$o_t = \sigma(W_{xo}X_t + W_{yo}Y_{t-1} + b_0) \quad (10)$$

$$y_t = o_t * tanh_{(st)} \quad (11)$$

The proposed LSTM model consists of an input layer, two hidden layers, and an output layer. The input layer consists of 50 neurons, while the first hidden layer consists of 50 neurons, and the second hidden layer contains 25 neurons. The final layer contains a single cell. The model was trained using two different group sizes, namely 1 and 15 epochs. This study used the Adam optimizer and MSE as the loss function. To determine the optimal parameters, a series of tests were conducted and the parameters that yielded the best results were selected.

### 3) *Random Forest*

This model is an ensemble approach that can assist in categorization and regression problems. While learning, a random selection of features is chosen using a modified tree-learning method. This approach regulates only a random subset of factors to determine the optimal split at each node. The input vector is given to each tree in the Random Forest Model (RFM) for the classification assignment, and each tree casts a vote for a class. Then, the class that receives the most scores is selected. Compared to other models, RFM can handle large input datasets and eliminates the overfitting problem by aggregating or combining the results of different decision trees. RFM is a suitable choice for time series tasks, due to its ensemble nature. This study used the random forest regressor with 100 estimators to predict the closing price. The RFM aggregates the outputs of individual decision trees to determine the mean value for regression tasks and uses consensus voting among decision trees for classification tasks. Several hyperparameters can be tuned to enhance the RFM performance. Notable hyperparameters include maximal features, minimal sample split, and the number of estimators. The number of estimators determines the number of decision trees constructed, and more trees generally improve performance at the cost of increasing computation time. The minimal sample split determines the minimum number of samples required to split an internal node, which should be adjusted based on the dataset's size. Furthermore, RFM is used to assess the relative importance of features, allowing the selection of the most significant characteristics for model construction. Overall, the RFM provides a robust framework for tackling categorization and regression problems, particularly in handling large datasets and capturing temporal patterns.

## III. RESULTS

Three benchmarks were used to train and testing the proposed models. These datasets were divided into training and test datasets in a 70:30 ratio, respectively. A PC with a 2.60 GHz Intel i7 CPU and 16 GB of RAM and Python Pytorch 4 on Widows 10 was used for all tests. The time (s) required to finish one round of processing is noted for each type. The following metrics were used, with their respective mathematical formulas:

- RMSE: Measures the deviation between the actual and predicted values.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y' - y)^2} \quad (12)$$

- MAE: Represents the average of the absolute errors in prediction across all samples in the test dataset.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y' - y| \quad (13)$$

- MAPE: Quantifies the accuracy of the predictions in percentage terms.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y' - y|}{y} \quad (14)$$

The investigation of RQ1 showed that machine learning algorithms yield superior outcomes compared to the statistical ARIMA model. This study used a range of models including ARIMA, Random Forest, LSTM, and Transformer. Tables I, II, and III display the comprehensive findings that correspond to the datasets collected from Yahoo Finance, Facebook, and JPMorgan, respectively. In particular, Tables I-III illustrate that the ARIMA model performed worse than the learning-based techniques. The investigation of RQ2 showed that deep learning algorithms delivered superior results compared to conventional machine learning models. This validation is evident in Tables I-III across all benchmark datasets. The RFM exhibited inferior performance when compared to LSTM and Transformer-based models. The Transformer-based approach demonstrated significantly improved results compared to both machine learning and statistical methods.

TABLE I. YAHOO FINANCE DATASET

| Models | MAE | RMSE | MAPE |
|---|---|---|---|
| ARIMA | 2.351 | 3.099 | 1.503 |
| Random Forest | 2.250 | 3.154 | 1.500 |
| LSTM | 1.758 | 2.157 | 1.425 |
| Transformer | 1.254 | 2.056 | 1.256 |

TABLE II. FACEBOOK STOCK FORECASTING RESULTS

| Models | MAE | RMSE | MAPE |
|---|---|---|---|
| ARIMA | 2.423 | 3.256 | 1.423 |
| Random Forest | 2.265 | 3.164 | 1.240 |
| LSTM | 1.756 | 2.124 | 1.435 |
| Transformer | 1.095 | 1.985 | 1.125 |

TABLE III. JPMORGAN FORECASTING RESULTS

| Models | MAE | RMSE | MAPE |
|---|---|---|---|
| ARIMA | 2.389 | 3.453 | 1.564 |
| Random Forest | 2.145 | 3.256 | 1.356 |
| LSTM | 1.0995 | 2.125 | 1.235 |
| Transformer | 1.088 | 1.865 | 1.116 |

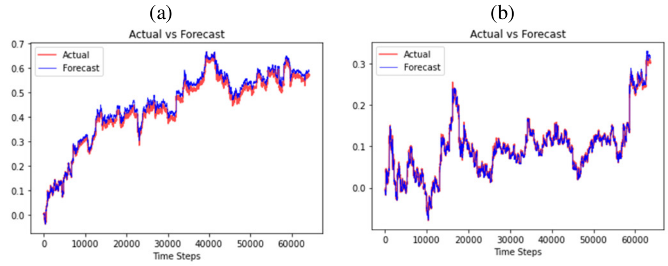


Fig. 4. Transformer forecasting on (a) FB and (b) JP Morgan.

## IV. DISCUSSION

Due to continuously fluctuating stock values that depend on numerous factors, predicting the return of the stock market is both important and difficult to do. This complexity and dependency on various factors make it difficult to accurately forecast the returns and closing prices of multiple stocks in the stock market. Although each company's website lists a small portion of the previous dataset, such as High, Low, Open, Close, and Volume, this information is inadequate to make a reliable forecast. This information only projects a narrow picture of the prospects and performance of the stock in the future. These variables can be used to generate new variables to increase the accuracy of the prediction. An increased number of variables responsible for the movement of the stock market can make the predictions more accurate. The most widely used method for making stock forecasts is a trend-based strategy that extrapolates a company's future worth from its historical stock prices. This method assumes that the particular company maintains its previous position and keeps going on the track, therefore, maintaining the projectile based on its past record. This is regarded as a tried-and-trusted strategy. Traders or investors have the greatest impact on a company's worth, and if everyone follows the same strategy, it will produce the same results. Since people tend to follow similar approaches, the result was expected. As a result, the primary driver of a company's ultimate value is its investors. Predicting broad trends is simple, but predicting a sudden shift in a company's worth is challenging and has long confounded academics and researchers, as it typically occurs as a result of various factors, such as significant company news or changes to the general stock market.

This study used an LSTM and a Transformer model to forecast stock market prices. These models were evaluated on three benchmark datasets, using three distinct situations, including daily predictions of using timestamps from 30, 60, and 90 days in the past. The results showed that a 90-day timeframe yielded superior results as the models advanced in their learning of more data distribution. The suggested techniques performed well, which can help achieve the desired outcomes. The combination of LSTM and regression significantly increased the precision of the process and generated better outcomes. Newly developed deep learning techniques for market forecasting produced promising outcomes, as the proposed Transformed-based method maximized the accuracy of the predicted stock prices.

Due to the complications in examining the stock data and the fluctuating closing price of the stocks, it is difficult to carefully analyze the stock data using conventional machine learning methods. To decide which company's stock to purchase or sell, an investor must first understand how the stock market fluctuates. Future stock prices are heavily influenced by a variety of factors, including company traits, the stock's historical price, and recent financial news about that particular stock. This study was designed to assist investors in making informed decisions regarding the buy or sell process and protect them from potential financial losses.

## V. CONCLUSION AND FUTURE DIRECTION

The stock market is a complex and ever-changing environment that is difficult to predict accurately. Traditional methods of analyzing stock data may not be enough to provide investors with reliable insights into the stock market. Machine learning and deep learning algorithms offer promising solutions to this problem. By analyzing historical stock data and using advanced techniques, these algorithms can generate predictions with greater accuracy, enabling investors to make more informed decisions about buying or selling stocks. In addition, the Transformed-based framework proposed in this study could help investors protect themselves from financial loss by allowing them to predict the price of a stock and make more informed decisions. The proposed Transformer-based model has some limitations, such as input structure, temporal dependencies, interpretability, and limited training data. Stock exchange datasets have some time missing values and unstructured information that make the function and the results of the proposed model harder. Another problem is that time series are limited, so effective training cannot be possible. In the future, the ensemble approach of machine learning will be investigated with a bio-inspired technique to overcome these issues.

## REFERENCES

[1] A. Thakkar and K. Chaudhari, "Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions," *Information Fusion*, vol. 65, pp. 95–107, Jan. 2021, https://doi.org/10.1016/j.inffus.2020.08.019.

[2] S. M. Idrees, M. A. Alam, and P. Agarwal, "A Prediction Approach for Stock Market Volatility Based on Time Series Data," *IEEE Access*, vol. 7, pp. 17287–17298, 2019, https://doi.org/10.1109/ACCESS.2019.2895252.

[3] K. C. Rasekhschaffe and R. C. Jones, "Machine Learning for Stock Selection," *Financial Analysts Journal*, vol. 75, no. 3, pp. 70–88, Jul. 2019, https://doi.org/10.1080/0015198X.2019.1596678.

[4] C. S. Wong and W. K. Li, "On a Mixture Autoregressive Model," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 62, no. 1, pp. 95–115, Jan. 2000, https://doi.org/10.1111/1467-9868.00222.

[5] M. R. Hassan and B. Nath, "Stock market forecasting using hidden Markov model: a new approach," in *5th International Conference on Intelligent Systems Design and Applications (ISDA'05)*, Warsaw, Poland, Sep. 2005, pp. 192–196, https://doi.org/10.1109/ISDA.2005.85.

[6] E. K. Ampomah, Z. Qin, and G. Nyame, "Evaluation of Tree-Based Ensemble Machine Learning Models in Predicting Stock Price Direction

of Movement," *Information*, vol. 11, no. 6, Jun. 2020, Art. no. 332, https://doi.org/10.3390/info11060332.

[7] X. Yu and D. Li, "Important Trading Point Prediction Using a Hybrid Convolutional Recurrent Neural Network," *Applied Sciences*, vol. 11, no. 9, Jan. 2021, Art. no. 3984, https://doi.org/10.3390/app11093984.

[8] A. P. Ruiz, A. A. Gila, U. Irusta, and J. E. Huguet, "Why Deep Learning Performs Better than Classical Learning?," *DYNA Ingeneria e Industria*, vol. 95, no. 2, pp. 119–122, Mar. 2020, https://doi.org/10.6036/9574.

[9] J. Liu, X. Guo, B. Li, and Y. Yuan, "COINet: Adaptive Segmentation with Co-Interactive Network for Autonomous Driving," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Prague, Czech Republic, Sep. 2021, pp. 4800–4806, https://doi.org/10.1109/IROS51168.2021.9636111.

[10] N. M. H. Masoud, "The Impact of Stock Market Performance upon Economic Growth," *International Journal of Economics and Financial Issues*, vol. 3, no. 4, pp. 788–798, Dec. 2013.

[11] A. Murkute and T. Sarode, "Forecasting Market Price of Stock using Artificial Neural Network," *International Journal of Computer Applications*, vol. 124, no. 12, pp. 11–15, Aug. 2015, https://doi.org/10.5120/ijca2015905681.

[12] L. J. Kao, C. C. Chiu, C. J. Lu, and J. L. Yang, "Integration of nonlinear independent component analysis and support vector regression for stock price forecasting," *Neurocomputing*, vol. 99, pp. 534–542, Jan. 2013, https://doi.org/10.1016/j.neucom.2012.06.037.

[13] W. Khan, M. ali Ghazanfar, M. Assam, S. Ahmad, and J. Khan, "Predicting Trend in Stock Market Exchange Using Machine Learning Classifiers," *Science International*, vol. 28, no. 2, pp. 1363–1367, May 2016.

[14] R. Gupta, N. Garg, and S. Singh, "Stock Market Prediction Accuracy Analysis Using Kappa Measure," in *2013 International Conference on Communication Systems and Network Technologies*, Gwalior, India, Apr. 2013, pp. 635–639, https://doi.org/10.1109/CSNT.2013.136.

[15] A. E. Ezugwu *et al.*, "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects," *Engineering Applications of Artificial Intelligence*, vol. 110, Apr. 2022, Art. no. 104743, https://doi.org/10.1016/j.engappai.2022.104743.

[16] L. Li, Y. Wu, Y. Ou, Q. Li, Y. Zhou, and D. Chen, "Research on machine learning algorithms and feature extraction for time series," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Montreal, QC, Canada, Jul. 2017, pp. 1–5, https://doi.org/10.1109/PIMRC.2017.8292668.

[17] J. Zhao, N. Sun, and W. Cheng, "Logistics forum based prediction on stock index using intelligent data analysis and processing of online web posts," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 9, pp. 3575–3584, Sep. 2020, https://doi.org/10.1007/s12652-019-01520-x.

[18] A. S. Saud and S. Shakya, "Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE," *Procedia Computer Science*, vol. 167, pp. 788–798, Jan. 2020, https://doi.org/10.1016/j.procs.2020.03.419.

[19] S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia, and D. C. Anastasiu, "Stock Price Prediction Using News Sentiment Analysis," presented at the 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService), Newark, CA, USA, Apr. 2019, pp. 205–208, https://doi.org/10.1109/BigDataService.2019.00035.

[20] A. Ghosh, S. Bose, G. Maji, N. Debnath, and S. Sen, "Stock Price Prediction Using LSTM on Indian Share Market," in *Proceedings of 32nd International Conference on Computer Applications in Industry and Engineering*, 2019, pp. 101–110, https://doi.org/10.29007/qgcz.

[21] "Neural Network Model vs. SARIMA Model In Forecasting Korean Stock Price Index (KOSPI)," *Issues In Information Systems*, vol. VIII, no. 2, 2007, https://doi.org/10.48009/2_iis_2007_372-378.

[22] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," in *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, Cambridge, UK, Mar. 2014, pp. 106–112, https://doi.org/10.1109/UKSim.2014.67.

[23] M. Almasarweh and S. A. Wadi, "ARIMA Model in Predicting Banking Stock Market Data," *Modern Applied Science*, vol. 12, no. 11, 2018.

[24] I. R. Parray, S. S. Khurana, M. Kumar, and A. A. Altalbe, "Time series data analysis of stock price movement using machine learning techniques," *Soft Computing*, vol. 24, no. 21, pp. 16509–16517, Nov. 2020, https://doi.org/10.1007/s00500-020-04957-x.

[25] S. Joseph, N. Mduma, and D. Nyambo, "A Deep Learning Model for Predicting Stock Prices in Tanzania," *Engineering, Technology & Applied Science Research*, vol. 13, no. 2, pp. 10517–10522, Apr. 2023, https://doi.org/10.48084/etasr.5710.

[26] U. P. Gurav and S. Kotrappa, "Sentiment Aware Stock Price Forecasting using an SA-RNN-LBL Learning Model," *Engineering, Technology & Applied Science Research*, vol. 10, no. 5, pp. 6356–6361, Oct. 2020, https://doi.org/10.48084/etasr.3805.

[27] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, https://doi.org/10.1038/nature14539.

[28] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, Jul. 2020, https://doi.org/10.1016/j.ijforecast.2019.07.001.

[29] D. T. Mirikitani and N. Nikolaev, "Recursive Bayesian Recurrent Neural Networks for Time-Series Modeling," *IEEE Transactions on Neural Networks*, vol. 21, no. 2, pp. 262–274, Oct. 2010, https://doi.org/10.1109/TNN.2009.2036174.

[30] H. Wang, G. Li, G. Wang, J. Peng, H. Jiang, and Y. Liu, "Deep learning based ensemble approach for probabilistic wind power forecasting," *Applied Energy*, vol. 188, pp. 56–70, Feb. 2017, https://doi.org/10.1016/j.apenergy.2016.11.111.

[31] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, and W. Woo, "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting," in *Advances in Neural Information Processing Systems*, 2015, vol. 28.

[32] K. Amarasinghe, D. L. Marino, and M. Manic, "Deep neural networks for energy load forecasting," in *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, Edinburgh, UK, Jun. 2017, pp. 1483–1488, https://doi.org/10.1109/ISIE.2017.8001465.

[33] C. J. Huang and P. H. Kuo, "A Deep CNN-LSTM Model for Particulate Matter (PM2.5) Forecasting in Smart Cities," *Sensors*, vol. 18, no. 7, 2018, https://doi.org/10.3390/s18072220.

[34] Y. Sudriani, I. Ridwansyah, and H. A. Rustini, "Long short term memory (LSTM) recurrent neural network (RNN) for discharge level prediction and forecast in Cimandiri river, Indonesia," *IOP Conference Series: Earth and Environmental Science*, vol. 299, no. 1, Apr. 2019, Art. no. 012037, https://doi.org/10.1088/1755-1315/299/1/012037.

[35] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Deep learning for event-driven stock prediction," in *Proceedings of the 24th International Conference on Artificial Intelligence*, Buenos Aires, Argentina, Apr. 2015, pp. 2327–2333.

[36] D. M. Q. Nelson, A. C. M. Pereira, and R. A. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in *2017 International Joint Conference on Neural Networks (IJCNN)*, Anchorage, AK, USA, Feb. 2017, pp. 1419–1426, https://doi.org/10.1109/IJCNN.2017.7966019.

[37] S. Al-Janabi, A. Alkaim, E. Al-Janabi, A. Aljeboree, and M. Mustafa, "Intelligent forecaster of concentrations (PM2.5, PM10, NO2, CO, O3, SO2) caused air pollution (IFCsAP)," *Neural Computing and Applications*, vol. 33, no. 21, pp. 14199–14229, Nov. 2021, https://doi.org/10.1007/s00521-021-06067-7.

[38] Y. Hu, "Stock market timing model based on convolutional neural network–a case study of Shanghai composite index," *Finance& Economy*, vol. 4, pp. 71–74, 2018.

[39] W. Long, Z. Lu, and L. Cui, "Deep learning-based feature engineering for stock price movement prediction," *Knowledge-Based Systems*, vol. 164, pp. 163–173, Jan. 2019, https://doi.org/10.1016/j.knosys.2018.10.034.

[40] X. Pang, Y. Zhou, P. Wang, W. Lin, and V. Chang, "An innovative neural network approach for stock market prediction," *The Journal of*

*Supercomputing*, vol. 76, no. 3, pp. 2098–2118, Mar. 2020, https://doi.org/10.1007/s11227-017-2228-y.

[41] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into Deep Learning." arXiv, Feb. 10, 2023, https://doi.org/10.48550/arXiv.2106.11342.

[42] S. K. Filipova-Petrakieva and V. Dochev, "Short-Term Forecasting of Hourly Electricity Power Demand: Reggresion and Cluster Methods for Short-Term Prognosis," *Engineering, Technology & Applied Science Research*, vol. 12, no. 2, pp. 8374–8381, Apr. 2022, https://doi.org/10.48084/etasr.4787.

[43] "Yahoo Finance - Stock Market Live, Quotes, Business & Finance News." https://finance.yahoo.com/.

[44] "Meta - Financials." https://investor.fb.com/financials/default.aspx.

[45] "J.P. Morgan Data and Analytics." https://www.jpmorgan.com/securities-services/data-analytics.

[46] E. Haugsdal, E. Aune, and M. Ruocco, "Persistence Initialization: A novel adaptation of the Transformer architecture for Time Series Forecasting." arXiv, Aug. 30, 2022, https://doi.org/10.48550/arXiv.2208.14236.

[47] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, https://doi.org/10.1162/neco.1997.9.8.1735.

[48] S. Fernández, A. Graves, and J. Schmidhuber, "An Application of Recurrent Neural Networks to Discriminative Keyword Spotting," in *Artificial Neural Networks – ICANN 2007*, 2007, pp. 220–229, https://doi.org/10.1007/978-3-540-74695-9_23.

[49] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, Jan. 2015, https://doi.org/10.1016/j.neunet.2014.09.003.