ISSN: 2754-6659

Journal of Artificial Intelligence & Cloud Computing

SCIENTIFIC Research and Community

Review Article

Open Access

Forecasting Stock Price Movements With Deep Learning Models for time Series Data Analysis

Varun Bitkuri[1], Raghuvaran Kendyala[2], Jagan Kurma[3], Jaya Vardhani Mamidala[4], Sunil Jacob Enokkaren[5] and Avinash Attipalli[6]

[1]Stratford University ,Software Engineer, USA

[2]University of Illinois at Springfield, Department of Computer Science, USA

[3]Christian Brothers University, Computer Information Systems, USA

[4]University of Central Missouri, Department of Computer Science, USA

[5]ADP, Solution Architect, USA

[6]University of Bridgeport, Department of Computer Science, USA

ABSTRACT

It is challenging to anticipate stock price fluctuations with any degree of accuracy since they are nonlinear, non-stationary, and volatile. Such challenges are addressed by developing a deep learning-based forecasting model that blends advanced preprocessing with sequential modelling. The empirical foundation with the use of S&P 500 index data, including 16,706 records (16,706 daily records) of data, is used, 1950- 2016. Data was prepared by cleaning, feature engineering, normalizing the data, and Empirical Mode Decomposition (EMD) to obtain intrinsic oscillatory modes. The nonlinear dynamics and temporal links of the time series were identified using a Gated Recurrent Unit (GRU) model. Robust error metrics, including R2, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE), were used to assess the model's prediction accuracy. The findings indicate that GRU had better performance and the R2 was 97.41%, RMSE was 0.0528%, MAE was 0.0416%, and MAPE was 0.2909%. Comparative analysis also showed that GRU was better than ANN and LSTM models. These results demonstrate the ability of deep learning (DL) algorithms to accurately capture the complexity of the market, providing valuable insights that benefit forecasts for investors, portfolio managers, and policymakers. Future developments may consider high-frequency data and hybrid DL architectures to be extended to cover a wider range of applications.

***Corresponding author**

Varun Bitkuri, Stratford University ,Software Engineer, USA.

Received: December 15, 2023; **Accepted:** December 21, 2023; **Published:** December 30, 2023

Keywords: Stock Price Forecasting, Deep Learning, Time Series Analysis, Financial Time Series, Market Volatility, Financial Prediction

Introduction

The financial markets might be regarded as one of the most captivating inventions of the contemporary world, which has had a profound effect on business, education, employment, technological development, and the economy in general. Researchers and investors have been particularly interested in predicting changes in stock prices because they potentially affect the process of making investment decisions, risk management, and portfolio optimisation [1,2]. However, because of its nonlinear, volatile, and dynamic character, analysing stock market activity is a very difficult process. Economic variables are among the general elements that influence the stock prices in question, along with politics, investors' psychology, and company-specific factors. All these complications guarantee that the future projections process is highly uncertain yet profitable, as even the most minor advantages of the forecast process can be translated into substantial gains, which will be experienced on the financial scale [3].

Traditional stock price prediction techniques have mostly relied on econometric and statistical techniques such as autoregressive conditional heteroskedasticity (ARCH/GARCH), autoregressive integrated moving average (ARIMA), and autoregressive moving average (ARMA). Such models have proved helpful in summarizing some of the statistical properties of financial time series, particularly volatility clustering and short-term autocorrelation. However, they are limited in a major way [4]. Assumptions of linearity and stationarity tend to limit their performance and are not characteristic of the financial markets' unpredictability and chaos. Moreover, a time series model developed with a specific time series of only one company often fails to scale or fit another company, hence its applicability is limited [5]. They are therefore unable to deal with large-dimensional, nonlinear, and noisy stock market data, which requires these traditional models as a foundation. To counter these issues, some machine learning (ML) methods have gained popularity as an alternative solution [6]. ML models are better at uncovering the underlying pattern of the complex datasets without necessarily having to make rigid statistic assumptions. In further detail, deep learning (DL) models have recently become promising predictors of time series [7]. Recurrent neural networks (RNNs), long short-term memory (LSTM) networks, gated recurrent units (GRUs), and early transformer-based models have shown promise in tasks requiring sequence modelling. Their ability to learn complicated time dependencies to extract knowledge from vast amounts of data, and to model nonlinear behaviours is due to their particular adaptation to the task of financial forecasting, which often involves trends that occur across multiple time cycles [8].

The stock market information when viewed as a time series demonstrates more issues. The prices do not remain in place but fluctuate and are always influenced by the internal and external activities in the market, such as the macroeconomic factors, events in the news, and investor psychology. It is a multidimensional problem that requires models capable of calculating sequential dependence, noise, and long dependency [9]. The best way to address these needs is through the use of DL models, which effectively learn detailed representations based on data, offering a stronger framework for predicting the stock market [10]. DL technology' growing application in financial time series analysis is thus a potentially promising development to overcome the disadvantages of the traditional forecasting models, and it opens new possibilities to forecast the stock prices with greater accuracy and more flexibility.

Motivation and Contribution of the Paper

The increased complexity and volatility inform this paper of financial markets in which adequate forecasting of price movements of stocks is a requisite for investors, portfolio managers and policymakers. Nonlinear patterns and long-run correlation, which constitute foundation of stock exchange time series data, have a high probability of failing to be detected by the conventional statistical and econometric models. Additionally, predictive activities become even more challenging since financial indicators are dynamic and multidimensional. The advanced capabilities of DL models enable them to learn more complex temporal correlations and nonlinear dynamics, making stock behaviour predictions with greater confidence. The current research utilises advanced deep learning (DL) algorithms, as well as empirical mode decomposition, to develop a powerful forecasting model capable of modelling both long-term market cycles and short-term fluctuations.

The Following are this Study's Primary Contributions

- Used a total of 16706 daily S&P500 index records (closing prices and daily returns) between the years 1950 and 2016 to obtain both the volatility and long-term market behaviour.
- Conducted a lot of pre-processing and feature engineering work, including addressing outliers and missing values, creating lag variables, and using rolling statistics to more accurately depict time.
- Applied Empirical Mode Decomposition (EMD) to obtain intrinsic oscillatory modes of the time series to enhance the model with multi-scale financial patterns.
- Used normalization methods to normalize feature values, which resulted in a quicker and more consistent model training.
- Trained and developed a GRU model that can recognise nonlinearities and sequential relationships in time series data related to finance.
- Measured predictive performance based on powerful measures of MAPE, MAE, RMSE, and R2 and offered a comprehensive measure of forecasting errors and extrapolation.
-

Justification and Novelty of the Paper

This study is justified by the drawbacks of other econometric models (ARIMA and GARCH), which are unable to capture nonlinearities, non-stationarity, and long-term relationships in time series related to finance. The markets are becoming increasingly volatile and complex, thus there is an urgent need for models that are adaptable to volatile and high-dimensional data. This paper is novel because it integrates Empirical Mode Decomposition (EMD) with GRU to construct a predictive model that is capable of modelling intrinsic oscillatory trends and sequential relationships. This is a hybrid method that improves prediction accuracy while maintaining computational efficiency, compared to current techniques.

Structure of Paper

The paper is structured in the following manner: Section II provides a literature review of related research done in stock price forecasting. Section III describes the methodology, which comprises data pre-processing, model implementation, and evaluation. The performance evaluation and discussion on it are found in Section IV. Lastly, Section V ends with a conclusion of some of the most important findings and future research recommendations.

Literature Review

The reviewed literature examines various approaches to stock price forecasting, demonstrating encouraging results while revealing persistent challenges in handling noise, volatility, and accuracy in high-frequency financial time series. previously looked into predicting the stock market using ML techniques, especially for financial time series like the S&P 500. They proposed a new process that reduces the amount of noisy financial data by reconstructing sequences with motifs (recurring patterns) and utilises a convolutional neural network (CNN) to identify the spatial structure of the time series. Their experimental findings revealed that their method is more efficient in satisfying the stock market's expectations, which creates non-stationary patterns and high volatility, by increasing the accuracy of frequency trading pattern modelling and classical signal processing approaches in stock trend prediction by 4–7% [11]. discuss the problem of training efficient stock selection models with high-frequency stock data. They postulate two architectures: CNN and LSTM, using high-frequency price information. The CNN model also has an advantage over the low-frequency feature prediction model as it does not entail the loss of original information and eliminates high-level elements. The LSTM model is based on the potential RNN to work with time series. With the transaction costs, CNN model yields a net rate of return of 62.27% per year and the LSTM yields 50.31, respectively [12]. created a deep neural network (DNN) with a discrete wavelet transform (DWT) to improve the use of financial time series. The model outperformed other DL topologies and models, such as ARIMA. Since the DWT separates data into frequency and time, it offers a more accurate depiction of the sequential behaviour of high-frequency data, which was used for these studies. Among the 27 input variables utilised were the last three one-minute pseudo-log-returns and the last three one-minute compressed tick-by-tick wavelet vectors. The expected one-minute average price was then predicted by the DNN using the following one-minute pseudo-log-return prediction. Directional Accuracy (DA) of the DNN made a fantastic forecast rate of 64% to 72% [13]. responded to ANN (and specifically LSTM networks) in financial time series analysis. They trained an LSTM model with six layers and ensemble-learned the network using the Bagging method on the Chinese Stock Market, involving eight LSTM networks. Indexes like the Shenzhen and Shanghai Composite Indexes were studied in the project, This lasted till December 29, 2017, and from January 4, 2012. The model of ensemble learning achieved 58.5% accuracy, 58.33% precision, 73.5% recall, 64.5% F1 score, and 57.67% AUC, which is higher than the traditional multilayer models of LSTM and other models with high predictive abilities [14]. introduce a novel prediction model that uses the Bidirectional Gated Recurrent Unit (BGRU) to forecast potential market moves by considering internet financial news as well as past stock price data. According to experimental results, the model describes both individual stocks (with an accuracy of over 65%) and the S&P 500 index (about 60% accurate). This demonstrates the value of combining various data sources [15]. discuss the application of the LSTM networks as a predictor of future tendencies of the stock market using past price history and technical indicators of its analysis. The authors have built a prediction model and tested it to evaluate the performance against alternative approaches of ML and investment strategies. The findings were encouraging as the overall accuracy was 55.9% in determining if there was a recent gain or reduction in a stock's price. This is especially effective in dynamic, complex and chaotic environments, where the results of the ML algorithms have demonstrated reasonable performance in predicting changes in stock values [16]. Table I below is a review of literature focusing on methodologies, datasets, findings, limitations, and gaps in the research on using DL to forecast frequent changes in stock

Table: Literature Review on Deep Learning Models for time Series Stock Price Movements Forecasting

Author(s)	Methodology	Dataset	Key Findings	Limitations	Future Direction
Wen et al. (2019)	CNN with motif-based sequence reconstruction	General stock market time series	4–7% accuracy improvement over traditional methods	Focus on noise reduction and spatial patterns; not designed for high-frequency intraday data	Apply motif-based sequence reconstruction directly on high-frequency tick/second data, then use hybrid CNN-LSTM/Transformers for intraday prediction
Yang et al. (2019)	CNN and LSTM for return prediction	High-frequency price data (U.S. market, several days)	CNN achieved 62.27% annualized net return; LSTM 50.31%	High transaction costs due to daily position changes; limited dataset scope	Develop cost-aware and risk-adjusted models; expand to multi-market high-frequency datasets
Arévalo et al. (2018)	DNN + Discrete Wavelet Transform (DWT)	Tick-by-tick data of 19 DJIA companies (2015–2017)	Directional Accuracy of 64–72%	Requires wavelet preprocessing (computationally heavy); limited generalizability	Explore deep feature extraction without heavy transforms (e.g., CNN/attention mechanisms) for scalability in high-frequency environments
Xie et al. (2018)	Six-layer LSTM neural network; Ensemble of 8 LSTMs with Bagging	Chinese Stock Market indices: Shanghai Composite, Shenzhen Composite, SSE 50, CSI 300, Medium & Small Cap, GEM (2012–2017)	Ensemble LSTM: Accuracy 58.5%, Precision 58.33%, Recall 73.5%, F1 64.5%, AUC 57.67%	Moderate accuracy; only applied to daily/weekly indices; limited to Chinese markets	Extend ensemble LSTM approach to intraday data, multi-market datasets, or hybrid deep learning models
Huynh, Dang & Duong (2017)	Bidirectional GRU (BGRU) with news + historical prices	S&P 500 index + individual stock data	60% accuracy (S&P 500), >65% (individual stocks)	Not high-frequency; accuracy moderate; limited dataset	Adapt BGRU/Transformers to high-frequency multimodal data (tick prices + real-time news sentiment)
Nelson, Pereira & de Oliveira (2017)	LSTM + technical analysis indicators	Historical daily stock data + technical indicators	55.9% accuracy in price movement prediction	Low-frequency focus; excluded news/macro factors; limited markets	Upgrade to LSTM/Transformer hybrids for intraday high-frequency technical indicators

Methodology

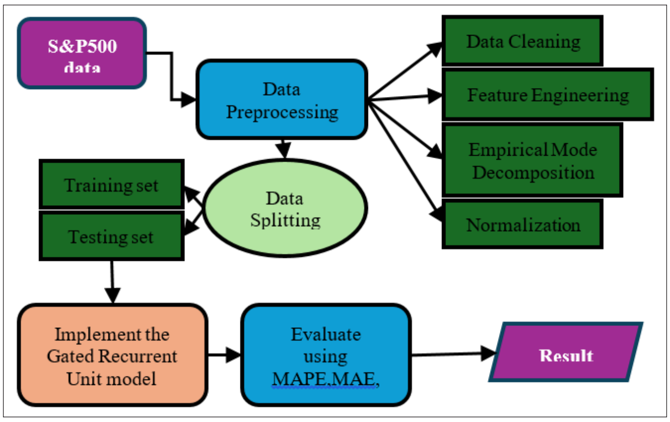

Figure 1: Methodology Framework for Stock Movement Forecasting

The methodology used in creating a trustworthy financial time series analysis forecasting model is the aim of this study, as shown in Figure 1. The data is 16,706 daily price records of the S&P 500 index between 1950-2016 including both the closing prices and daily returns that are used to estimate the market volatility and long-term trends. It starts with preprocessing, in which missing values and outliers are resolved by data cleaning, and feature engineering, in which lag variables and rolling statistics are added to improve temporal representation. Then use Empirical Mode Decomposition (EMD) to extract intrinsic oscillatory modes and normalize all features to a normalized range to guarantee good convergence of the model. The dataset is prepared and then divided into 70:30 training and testing groups, allowing for a reliable evaluation of the model on unseen data. The GRU model is applied to the training set to capture sequential relationships within the time series. At the same time, evaluation measures such as MAPE, MAE, RMSE, and R^2 are used to assess the predictive performance.

Data Collection

The data is historical, covering the S&P 500 index from 1950 to 2016, comprising 16,706 daily data entries. It contains both the closing values of the index and the calculated daily returns, which are used as measures of how the market has performed over time. These values not only give the variable's long-term trend as well as its short-term fluctuation, but also are appropriate in financial forecasting studies. The data is a true-life representation of real market dynamics whereby the market moves up and down, in a volatile manner, thus offering a stable platform to predictive modelling. The time series graphs are provided:

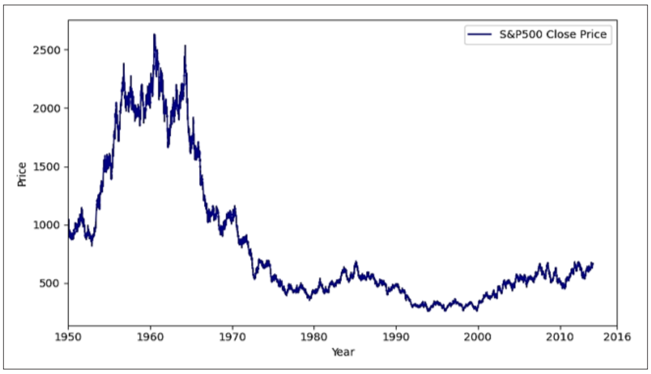

Figure 2: Closing Prices of the S&P 500 Index

In Figure 2, a line graph of the S&P 500 Index from 1950 to 2016 is shown. The year and the index's price are represented on the X and Y axes, respectively. Following the closing price of the S&P 500 over this period is the deep navy-blue chart. The graph exhibits significant volatility. The index shows a steep rise from the 1950s to the early 1960s, reaching a peak. It then enters a prolonged decline, bottoming out around the early 1980s. From there, the index gradually recovers and trends upward toward the end of the period, though it does not fully reach the highs observed in the 1960s. Overall, the graph highlights periods of substantial growth and decline, reflecting a dynamic and turbulent market history.

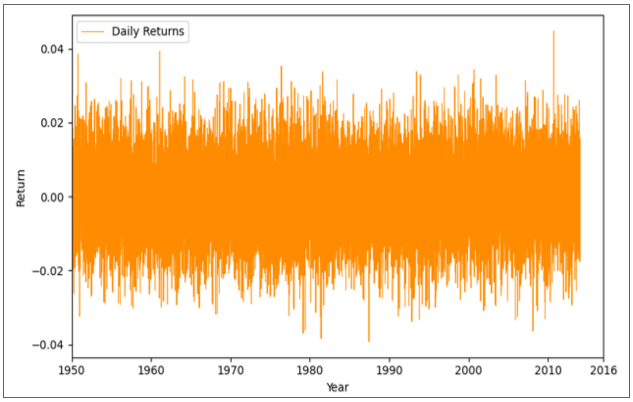

Figure 3: S&P 500 Daily Returns

A line graph of the S&P 500 daily returns is displayed in Figure 3. The X-axis shows the year, while the Y-axis shows the daily returns, which range from -0.04 to 0.04. The orange line displays the daily fluctuations in the S&P 500 Index throughout these 66 years. Unlike a chart of the index's price, this graph shows day-to-day percentage changes, resulting in a highly volatile and seemingly random pattern. The line oscillates around the zero-return mark, with frequent spikes both above and below, illustrating the constant and unpredictable nature of daily returns rather than a long-term trend.

Data Preparation

The data preparation phase involves converting raw email datasets into an organized, uniform manner that may be used for analysis. This step ensures uniformity, removes inconsistencies, and organizes the data with proper labels to support accurate training and evaluation. The pre-processing steps carried out include:

Data Cleaning

An essential step in getting financial time series ready for DL models is data cleansing, as the presence of missing or inconsistent values can significantly affect model training and prediction accuracy. The following actions are taken to enhance the dependability and quality of the data:

Identify missing values in prices and returns, which can occur due to non-trading days, data reporting issues, or technical errors.

Fill in missing values either forward-filled or backwards-filled to continue the time series while preserving time-dependent data.

Identify outliers through statistical tests like z-score analysis, interquartile Range (IQR), which are used to determine extreme deviations of usual market behaviour.

Correct or delete extreme cases either by capping, smoothing or deleting them, so that learning does not get distorted by large spikes or drops.

Feature Engineering
In model engineering, feature engineering is employed to enhance the performance of models by generating informative features from the data. Lag features represent the dependence on time, utilising past values of returns or prices. The rolling statistics capture trends and volatility, i.e. moving averages and rolling standard deviations through different windows. These designed features also enable the model to incorporate both transient variations and long-term trends, making it more effective at learning complex time variability and producing high-quality predictions.

Empirical Mode Decomposition
A nonlinear and non-stationary time series x(t) can be empirically separated into a residual trend and a finite number of IMFs using the EMD approach. Each IMF is a simple oscillation of the signal, and the rest of the signal is an ascending or descending trend. It is a dynamic decomposition that does not require any prior assumption on the signal [17]. The EMD procedure begins with the identification of the first IMF through an iterative sifting process. Let h_0 (t)=x(t) denotes the original signal. The detail in Equation (1) is obtained by subtracting the mean of h_(i-1) (t) from the upper and lower envelopes at each iteration, which is represented by m_(i-1) (t).

$$\mathbf{h_i(t) = h_{i-1}(t) - m_{i-1}(t)} \qquad (1)$$

This procedure is carried out repeatedly until h_i (t)Meets the requirements of the IMF. The first IMF is then defined as given in Equation (2):

$$\mathbf{IMF_1(t) = h_i(t)} \qquad (2)$$

The remaining signal is calculated by deducting the IMF from the unique signal after an IMF has been extracted. This residual is treated as a new signal, and the original signal can finally be reconstructed as defined in Equation (3):

$$\mathbf{x(t) = \sum_{k=1}^{n} IMF_k(t) + r_n(t)} \qquad (3)$$

Here, r_n (t)Is the residual component that cannot be decomposed further.

Data Normalization
Starting the training process for every feature within the same scale is another way that data normalisation might save training time. This method maps the incoming data into one of two predetermined ranges: [0, 1] or [-1, 1] [18]. A data set's properties are normalised using the min-max method according to their lowest and highest values. In the interval [0, 1], Equation (4) calculates a value x of the property X to x^'.

$$\mathbf{x' = \frac{x - x_{min}}{x_{max} - x_{min}}} \qquad (4)$$

Where the lowest and highest values of the data being considered are denoted by$\mathbf{x_{min}}$ and $\mathbf{x_{max}}$.

Data Splitting
The training and testing subsets of the dataset are divided 70:30. The model can learn temporal patterns by using the first 70% of the sequential data for training, with the remaining 30% set aside for testing, ensuring reliable evaluation on unseen data and preventing information leakage.

Implementation of Gated Recurrent Unit (GRU) model
The GRU model is capable of learning some dependencies that are introduced by Cho [19]. It is a specific kind of RNN created to overcome its drawbacks, including its inability to identify long-range connections due to the vanishing gradient issue. GRUs are extensively adaptable and have been effectively used to several issues, such as speech recognition, NLP, and time series forecasting. Structurally, GRUs employ a gating mechanism similar to that of LSTMs, but with a more streamlined design. In particular, the cell state and hidden state are effectively integrated, and the input and forget gates of an LSTM are merged into a single update gate [20]. Because fewer parameters are needed, the model is lighter, takes less processing resources, can be trained more quickly, and occasionally performs better on particular tasks. The end effect is a lighter model that should train more quickly and perform marginally better on some tasks. Figure 4 illustrates the usual GRU cell diagram.

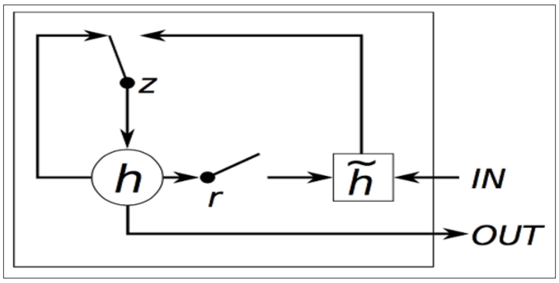

Figure 4: Gated Recurrent Unit Network [21].

The following are the forward propagation equations for common GRU gates: Equation (5-7):

$$\mathbf{z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)} \qquad (5)$$

$$\mathbf{r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)} \qquad (6)$$

$$\mathbf{h_t = (1 - z_t).\, h_{t-1} + z_t.\, tanh(W_h x_t + U_h(r_t.\, h_{t-1}) + b_h)} \quad (7)$$

Where W and U stand for the parameter matrices, b_z, b_r, and b_h Are the biases, and x_t, h_t, z_t, and r_t Are the input, output, update gate, and reset gate vectors, respectively.

Evaluation Parameters
The evaluation parameters are essential tools for evaluating regression models' effectiveness by measuring the variations between expected and actual values:

Mean Absolute Error (MAE)
An important statistic for assessing regression models is the MAE. If y_iis the corresponding real value and y ̂_i is the projected value of the ith sample, then the MAE may be calculated using Equation. (8):

$$\mathbf{MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|} \qquad (8)$$

Mean Absolute Percentage Error (MAPE)
A statistic of the extent to which a dependent series departs from the level anticipated by the model [22]. It may be used to compare series with various units, as it is not affected by the units used Equation.(9) computes MAPE as follows:

$$\mathbf{MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100} \qquad (9)$$

Root Mean Squared Error (RMSE)
It is applied to the differences between the values that a model forecasts and the values that are actually observed [23]. Stated differently, these individual differences are referred to as the

residuals, and the RMSE consolidates them into a single metric of predictive capability as presented in Equation. (10).

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad (10)$$

R2 score

The R² score is one of the most popular and generally recognised measures for assessing regression models. Given y_i as the iThe sample's anticipated value and y_iOnce its actual value, the following equations may be derived. The Equation.(11) used to compute the R2 score value are as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y}_i)^2} \qquad (11)$$

In this case, the actual value is represented by y, the predicted value by (y)̂, and the mean of all the actual values by (y)̅.

The metrics give a complete understanding of model performance to make objective comparisons and decisions in predictive analysis

Results Analysis and Discussions

The proposed research aims at predicting changes in stock prices based on DL models to analyze time series data. TensorFlow was used in Python to implement the experiment, and Keras to construct the models, and NumPy, pandas, and scikit-learn to pre-process and evaluate data. The PC used for the tests features an NVIDIA GPU, 16GB of RAM, and an Intel Core i7 CPU. The model was trained using the Adam optimizer and MSE loss. Table II summarizes the performance of the proposed GRU model. The model had an R-squared value of 97.41%, indicating a high capacity to forecast changes in stock prices. The error measures, include MAE (0.0416), RMSE (0.0528), and MAPE (0.2909%), indicate that the predictive accuracy is very high, and the error rate is very low. In general, these findings confirm the usefulness of the GRU model to represent the time-dependent variations in financial time series [22-30].

Table: Performance Evaluation of the GRU Model for Stock Price Forecasting

Parameter	GRU
R2	97.41
RMSE	0.0528
MAE	0.0416
MAPE	0.2909

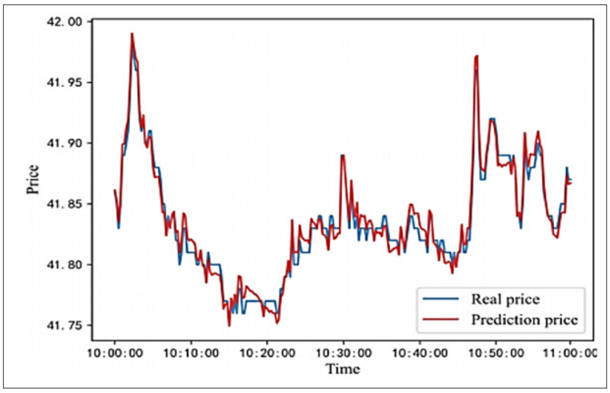

Figure 4: Real vs. Predicted Price Movements with GRU

A line graph illustrating the movement of actual and anticipated prices over one hour is displayed in Figure 5. The blue line, which depicts the actual price, depicts an upward movement that is sharp at the onset, and thereafter it starts falling, then it starts to move up and down before recording another major upward shoot. These moves are closely followed by the red line which is the prediction price. It indicates a steep increase, followed by a decrease and fluctuation, with an upward trend. Although the predicted price shows some deviations and lags, overall direction and volatility of the real price is well reflected by the predicted price, which shows high correlation with the actual market movement [31-40].

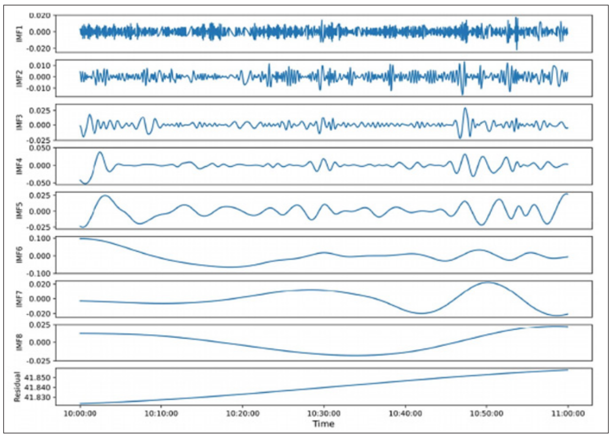

Figure 5: Time Series Feature Movements Using GRU Model

Figure 6 shows the time series prediction outcomes with the help of GRU model. The subplots model various statistical and predictive properties over time, and focus on the motions of signals within the dataset. The upper plots show quick and chaotic changes in raw input data, and the lower plots show more consistent and consistent changes as learned by the GRU. This modification demonstrates that the model is capable of taking into consideration both short-term and long-term variations. All in all, the GRU is a strong model of temporal relationships, as it eliminates noise and highlights the significant movement of serial data.

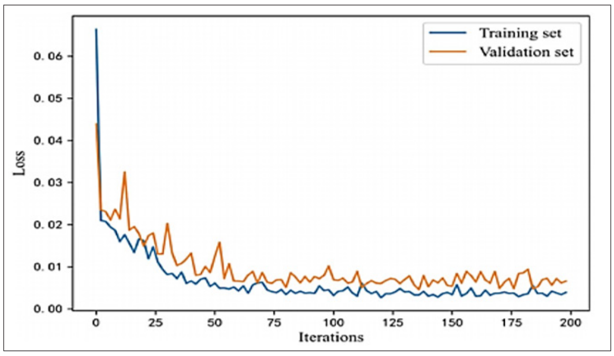

Figure 6: Training and Validation Loss Over Iterations of GRU Model

Figure 7 is a line graph illustrating how variables Epochs and Loss are related to identifying spam in emails, using an ANN model. The appropriate loss is displayed on the y-axis of a line graph that displays the ML model's training and validation loss across 200 training epochs. The x-axis displays the number of training epochs, which can range from 50 to 300. Whereas the x-axis shows the loss, the y-axis shows the number of iterations. The line corresponding to the training set, the blue line, depicts a steep reduction in loss at the beginning, followed by a gradual reduction and stabilisation. The line depicting the validation set (orange line) also declines albeit with a larger number of fluctuations. Once this happens (around 100 iterations), both converging lines stabilise and oscillate around a small loss value, indicating that the model has converged and is performing well on both training data and unseen data, with minimal severe overfitting.

Comparative Analysis

The comparison between the models to predict stock price movement as in Table III indicates the variations in the predictive power of ANN, LSTM, and GRU. Based on the results, ANN had a higher R2 value of 92.53% than LSTM which had a better value of 94.86% and GRU with a better value of 97.41%. These results indicate that ANN can offer a reasonable baseline, but its inability to memorize information hinders its capability to memorize sequential patterns. LSTM eliminates this shortcoming by using memory cells and some gating capabilities, which allow better management of long-term connections. GRU with its simplified efficient gating architecture provides better results with computational efficiency. In general, the table shows clearly that GRU performs best in stock price prediction [41-50].

Table: Comparative performance of deep learning models for stock price forecasting

Models	R2
ANN[24]	92.53
LSTM[25]	94.86
GRU	97.41

The results of this experiment demonstrate that GRUs are superior models to forecast stock price changes instead of ANN and LSTM models. Combining Empirical Mode Decomposition and normalization further enhanced The model's capacity to explain multi-scale and temporal phenomena dynamics. GRU has proven to be highly predictive, with minimal error against real figures, which demonstrates that the tool is exceptionally strong in financial forecasting tasks. The study is however limited to the S and P 500 index daily data. It can also be improved further by extending it to high-frequency and multimodal data.

Conclusion and Future Work

The effectiveness of GRU in predicting stock prices is demonstrated by the results provided in this study DL networks have been shown to possess the ability to adapt to the complexity and volatility of stock markets in a unique way. The system employed Empirical Mode Decomposition, normalisation, and feature engineering to enhance the quality of signals and time representation, enabling the GRU to build multi-scale financial trends. Its performance in terms of error measuring demonstrated its excellence with the following values: R^2 = 97.41, RMSE = 0.0528, MAE = 0.0416 and MAPE = 0.2909%. The GRU model is more resilient to sequential dependence and nonlinear market phenomena, as it is more accurate and computationally efficient than both ANN and LSTM. These results indicate that there is a need to combine the decomposition models with better neural designs to overcome the weaknesses of the conventional statistical approaches. Future research could apply this framework to intraday high-frequency data, which may involve multimodal data, such as emotion data, financial news, and economic statistics. It can also be improved in terms of performance and other applications in various stock markets by developing hybrid models, such as GRU-CNN models. Such extensions would result in a powerful decision support system to enable risk management and, when such extensions are used in the real world, the optimization of investment strategies [51-62].

References

1. Zhong X, Enke D (2017) Forecasting daily stock market return using dimensionality reduction, Expert Syst. Appl 67: 126-139.
2. Balasubramanian A (2018) Ai-Driven Predictive Maintenance And Process Automation In Industrial Plc Systems: A Case Study In The Oil And Gas Industry, Int. J. Core Eng. Manag 5: 14-28.
3. Shah D, Isah H, Zulkernine F (2019) Stock Market Analysis: A Review and Taxonomy of Prediction Techniques, Int. J. Financ. Stud 7: 2.
4. Cheng CH, Chen TL, Wei LY (2010) A hybrid model based on rough sets theory and genetic algorithms for stock price forecasting, Inf. Sci. (Ny) 180: 1610-1629.
5. Rajavel V, Balaji G, Gomathinayagam AV (2018) Eye Gaze Peculiarities Detection in Children with Autism using a Head-free cam, Int. J. Eng. Sci. Res. Technol 5: 868-876.
6. Rao DD (2009) Multimedia-Based Intelligent Content Networking for Future Internet, in 2009 Third UKSim European Symposium on Computer Modelling and Simulation 55-59.
7. G HM, V EA, Menon K, KPS (2018) NSE Stock Market Prediction Using Deep-Learning Models, Procedia Comput. Sci 132: 1351-1362.
8. Balaji AJ, Ram DSH, Nair BB (2018) Applicability of deep learning models for stock price forecasting an empirical study on BANKEX data, Procedia Comput. Sci 143: 947-953.
9. Shen G, Tan Q, Zhang H, Zeng P, Xu J (2018) Deep learning with gated recurrent unit networks for financial sequence predictions, Procedia Comput. Sci 131: 895-903.
10. Balasubramanian A (2019) AI-Driven Optimization of Urban Mobility: Integrating Autonomous Vehicles with Real-Time Traffic and Infrastructure Analytics, Int. J. Innov. Res. Creat. Technol 5: 1-13.
11. Wen M, Li P, Zhang L, Chen Y (2019) Stock Market Trend Prediction Using High-Order Information of Time Series, IEEE Access 7: 28299-28308.
12. Yang J, Li Y, Chen X, Cao J, Jiang K (2019) Deep Learning for Stock Selection Based on High Frequency Price-Volume Data https://arxiv.org/abs/1911.02502.
13. Arévalo A, Nino J, León D, Hernandez G, Sandoval J (2018) Deep learning and wavelets for high-frequency price forecasting,in International Conference on Computational Science 385-399.
14. Xie Q, Cheng G, Xu X, Zhao Z (2018) Research Based on Stock Predicting Model of Neural Networks Ensemble Learning, MATEC Web Conf https://www.matec-conferences.org/articles/matecconf/abs/2018/91/matecconf_eitce2018_02029/matecconf_eitce2018_02029.html.
15. Huynh HD, Dang LM, Duong D (2017) A New Model for Stock Price Movements Prediction Using Deep Neural Network, in Proceedings of the Eighth International Symposium on Information and Communication Technology 57-62.
16. Nelson DMQ, Pereira ACM, Oliveira RAde (2017) Stock market's price movement prediction with LSTM neural networks, in 2017 International Joint Conference on Neural Networks (IJCNN) 1419-1426.
17. Qiu X, Ren Y, Suganthan PN, Amaratunga GAJ (2017) Empirical Mode Decomposition-based ensemble deep learning for load demand time series forecasting, Appl. Soft Comput 54: 246-255.
18. Nayak SC, Misra BB, Behera HS (2014) Impact of data normalization on stock index forecasting, Int. J. Comput. Inf. Syst. Ind. Manag. Appl 6: 13.

19. Cho K, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, et al. (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation https://arxiv.org/abs/1406.1078.
20. Rahman MO, Hossain MS, Junaid TS, Forhad MSA, Hossen MK (2019) Predicting prices of stock market using gated recurrent units (GRUs) neural networks, Int. J. Comput. Sci. Netw. Secur 19: 213-222.
21. Di L, Persio, Honchar O (2017) Recurrent neural networks approach to the financial forecast of Google assets, Int. J. Math. Comput. Simul 11: 7-13.
22. Bhatt MV, Sheth NN, Patel TN (2010) Forecasting Share Prices: A Good Practice or a Waste of Time, Indian J. Financ 4: 47-53.
23. Rehman F, Kamal Y, Amin SU (2017) The relationship between idiosyncratic, stock market volatility and excess stock returns, Public Financ. Quarterly= Pénzügyi Szle 62: 311-325.
24. Sagir AM, Sathasivan S (2017) The use of artificial neural network and multiple linear regressions for stock market forecasting, Matematika 33: 1-10.
25. Sethia A, Raut P (2019) Application of LSTM, GRU and ICA for stock price prediction, Smart Innov. Syst. Technol 107: 479-487.
26. Rajiv C, Mukund Sai VT, Venkataswamy Naidu, Sriram G, Mitra P (2022) Leveraging Big Datasets for Machine Learning-Based Anomaly Detection in Cybersecurity Network Traffic. J Contemp Edu Theo Artific Intel https://www.cmjpublishers.com/wp-content/uploads/2025/06/leveraging-big-datasets-for-machine-learning-based-anomaly-detection-in-cybersecurity-network-traffic.pdf.
27. Sandeep Kumar C, Srikanth Reddy V, Ram Mohan P, Bhavana K, Ajay Babu K (2022) Efficient Machine Learning Approaches for Intrusion Identification of DDoS Attacks in Cloud Networks. J Contemp Edu Theo Artific Intel JCETAI/101.
28. Bhumireddy JR, Chalasani R, Tyagadurgam MSV, Gangineni VN, Pabbineedi S, et al. (2020) Big Data-Driven Time Series Forecasting for Financial Market Prediction: Deep Learning Models. Journal of Artificial Intelligence and Big Data 2: 153-164.
29. Nandiraju SKK, Chundru SK, Vangala SR, Polam RM, Kamarthapu B, et al. (2022) Advance of AI-Based Predictive Models for Diagnosis of Alzheimer's Disease (AD) in healthcare. Journal of Artificial Intelligence and Big Data 2: 141-152.
30. Tyagadurgam MSV, Gangineni VN, Pabbineedi S, Penmetsa M, Bhumireddy, et al. (2022) Designing an Intelligent Cybersecurity Intrusion Identify Framework Using Advanced Machine Learning Models in Cloud Computing. Universal Library of Engineering Technology https://ulopenaccess.com/papers/ULETE_SV01/ULETE2022SI_003.pdf.
31. Vangala SR, Polam RM, Kamarthapu B, Kakani AB, Nandiraju SKK, et al. (2022) Leveraging Artificial Intelligence Algorithms for Risk Prediction in Life Insurance Service Industry https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5515286.
32. Polam RM, Kamarthapu B, Kakani AB, Nandiraju SKK Chundru SK, et al. (2021) Data Security in Cloud Computing: Encryption, Zero Trust, and Homomorphic Encryption. International Journal of Emerging Trends in Computer Science and Information Technology 2: 70-80.
33. Gangineni VN, Pabbineedi S, Penmetsa M, Bhumireddy JR, Chalasani R, et al. (2022) Efficient Framework for Forecasting Auto Insurance Claims Utilizing Machine Learning Based Data-Driven Methodologies. International Research Journal of Economics and Management Studies https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5580771.
34. Vattikonda N, Gupta AK, Polu AR, Narra B, Buddula DVKR, et al. (2022) Blockchain Technology in Supply Chain and Logistics: A Comprehensive Review of Applications, Challenges, and Innovations. International Journal of Emerging Research in Engineering and Technology 3: 99-107.
35. Narra B, Vattikonda N, Gupta AK, Buddula DVKR, Patchipulusu HHS, et al. (2022) Revolutionizing Marketing Analytics: A Data-Driven Machine Learning Framework for Churn Prediction. International Journal of Artificial Intelligence, Data Science, and Machine Learning 3: 112-121.
36. Polu AR, Narra B, Buddula DVKR, Patchipulusu HHS, Vattikonda N, et al. (2022) BLOCKCHAIN TECHNOLOGY AS A TOOL FOR CYBERSECURITY: STRENGTHS, WEAKNESSES, AND POTENTIAL APPLICATIONS https://zenodo.org/records/15335349.
37. Bhumireddy JR, Chalasani R, Tyagadurgam MSV, Gangineni VN, Pabbineedi S, et al. (2022) Big Data-Driven Time Series Forecasting for Financial Market Prediction: Deep Learning Models. Journal of Artificial Intelligence and Big Data 2: 153-164.
38. Nandiraju SKK, Chundru SK, Vangala SR, Polam RM, Kamarthapu B, et al. (2022) Advance of AI-Based Predictive Models for Diagnosis of Alzheimer's Disease (AD) in healthcare. Journal of Artificial Intelligence and Big Data 2: 141-152.
39. HK K (2020) Design of Efficient FSM Based 3D Network on Chip Architecture. INTERNATIONAL JOURNAL OF ENGINEERING 68: 67-73.
40. Krutthika HK (2019) Modeling of Data Delivery Modes of Next Generation SOC-NOC Router. In 2019 Global Conference for Advancement in Technology (GCAT) 1-6.
41. Ajay S, Satya Sai Krishna Mohan G, Rao SS, Shaunak SB, Krutthika HK, et al. (2018) Source Hotspot Management in a Mesh Network on Chip. In VDAT 619-630.
42. Nair TR, Krutthika HK (2010) An Architectural Approach for Decoding and Distributing Functions in FPUs in a Functional Processor System arXiv preprint arXiv:1001.3781.
43. Gopalakrishnan Nair TR, Krutthika HK (2010) An Architectural Approach for Decoding and Distributing Functions in FPUs in a Functional Processor System. arXiv e-prints, arXiv-1001.
44. Krutthika HK, Aswatha AR (2021) Implementation and analysis of congestion prevention and fault tolerance in network on chip. Journal of Tianjin University Science and Technology 54: 213-231.
45. Krutthika HK, Aswatha AR (2020) FPGA-based design and architecture of network-on-chip router for efficient data propagation. IIOAB Journal 11: 7-25.
46. Krutthika HK, Aswatha AR (2020) Design of efficient FSM-based 3D network-on-chip architecture. International Journal of Engineering Trends and Technology 68: 67-73.
47. Krutthika HK, Rajashekhara R (2019) Network-on-chip: A survey on router design and algorithms. International Journal of Recent Technology and Engineering 7: 1687-1691.
48. Polam RM, Kamarthapu B, Kakani AB, Nandiraju SKK, Chundru SK, et al. (2021) Big Text Data Analysis for Sentiment Classification in Product Reviews Using Advanced Large Language Models. International Journal of AI, BigData, Computational and Management Studies 2: 55-65.
49. Gangineni VN, Tyagadurgam MSV, Chalasani R, Bhumireddy

JR, Penmetsa M (2021) Strengthening Cybersecurity Governance: The Impact of Firewalls on Risk Management. International Journal of AI, BigData, Computational and Management Studies 2: 10-63282.
50. Pabbineedi S, Penmetsa M, Bhumireddy JR, Chalasani R, Tyagadurgam MSV, et al. (2021) An Advanced Machine Learning Models Design for Fraud Identification in Healthcare Insurance. International Journal of Artificial Intelligence, Data Science, and Machine Learning 2: 26-34.
51. Kamarthapu B, Kakani AB, Nandiraju SKK, Chundru SK, Vangala SR, et al. (2021) Advanced Machine Learning Models for Detecting and Classifying Financial Fraud in Big Data-Driven. International Journal of Artificial Intelligence, Data Science, and Machine Learning 2: 39-46.
52. Tyagadurgam MSV, Gangineni VN, Pabbineedi S, Penmetsa M, Bhumireddy JR, et al. (2021) Enhancing IoT (Internet of Things) Security Through Intelligent Intrusion Detection Using ML Models. International Journal of Emerging Research in Engineering and Technology 2: 27-36.
53. Vangala SR, Polam RM, Kamarthapu B, Kakani AB, Nandiraju SKK, et al. (2021) Smart Healthcare: Machine Learning-Based Classification of Epileptic Seizure Disease Using EEG Signal Analysis. International Journal of Emerging Research in Engineering and Technology 2: 61-70.
54. Kakani AB, Nandiraju SKK, Chundru SK, Vangala SR, Polam RM, et al. (2021) Big Data and Predictive Analytics for Customer Retention: Exploring the Role of Machine Learning in E-Commerce. International Journal of Emerging Trends in Computer Science and Information Technology 2: 26-34.
55. Penmetsa M, Bhumireddy JR, Chalasani R, Tyagadurgam MSV, Gangineni VN, et al. (2021) Next-Generation Cybersecurity: The Role of AI and Quantum Computing in Threat Detection. International Journal of Emerging Trends in Computer Science and Information Technology 2: 54-61.
56. Polu AR, Vattikonda N, Gupta A, Patchipulusu H, Buddula DVKR, et al. (2021) Enhancing Marketing Analytics in Online Retailing through Machine Learning Classification Techniques. Available at SSRN 5297803.
57. Kalla D (2022) AI-Powered Driver Behavior Analysis and Accident Prevention Systems for Advanced Driver Assistance. International Journal of Scientific Research and Modern Technology (IJSRMT) https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5063253.
58. Dinesh K (2022) Navigating the link between internet user attitudes and cybersecurity awareness in the era of phishing challenges. International Advanced Research Journal in Science, Engineering and Technology https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4690554.
59. Kalla D, Kuraku DS, Samaah F (2021) Enhancing cyber security by predicting malwares using supervised machine learning models. International Journal of Computing and Artificial Intelligence 2: 55-62.
60. Katari A, Kalla D (2021) Cost Optimization in Cloud-Based Financial Data Lakes: Techniques and Case Studies. ESP Journal of Engineering & Technology Advancements (ESP-JETA) 1: 150-157.
61. Kalla D, Smith N, Samaah F, Polimetla K (2021) Facial Emotion and Sentiment Detection Using Convolutional Neural Network. Indian Journal of Artificial Intelligence Research (INDJAIR) 1: 1-13.
62. Polu AR, Buddula DVKR, Narra B, Gupta A, Vattikonda N, et al. (2021) Evolution of AI in Software Development and Cybersecurity: Unifying Automation, Innovation, and Protection in the Digital Age Available at SSRN 5266517.