Frontiers in Health Informatics
ISSN-Online: 2676-7104
www.healthinformaticsjournal.com
2024; Vol 13: Issue 3
Open Access

Information Technology Sectoral Index Stock Price Prediction Using Artificial Intelligence

Dr.S.Krishnaprabha[1], Dr J Meena Maheswari,[2]Dr M Reshma,[3]Dr Divya Lakshmi J, [4]

[1.]Associate Professor , MBA Department, Sri Ramakrishna Engineering College, Coimbatore
[2.]Associate Professor, PSGR Krishnammal College for women, Coimbatore.
[3.]Assistant Professor, PSGR Krishnammal College for women, Coimbatore.
[4.]Assistant Professor, PSGR Krishnammal College for women, Coimbatore-641004

Cite this paper: S.Krishnaprabha, J Meena Maheswari, M Reshma, Divya Lakshmi J, (2024) Information Technology Sectoral Index Stock Price Prediction Using Artificial Intelligence. *Frontiers in Health Informatics, 13* (3), 6793-6802

Abstract*: This article aims to illustrate the application of Artificial Intelligence (AI) in predicting stock market movements. By employing technical analysis, stock market predictions are simulated using regression machine learning (ML) algorithms. Using historical price data retrieved from Yahoo Finance, these algorithms predict the stock price trend by the close of each business day. Built upon the LSTM model, the neural network discerns patterns within the data sequences, leveraging these patterns to forecast future price movements accurately. For this analysis, data from the S&P BSE Information Technology (SI1000) sectoral index is used to predict future prices.*

Keywords— AI; ML; LSTM, Neural network, Stock market prediction

Introduction

The stock market is believed to be extremely unpredictable and risky for investors, and while there are several ways and instruments available to anticipate stock returns, the risk can be reduced if reliable prediction methods are accessible to assist investors in achieving higher returns. In the stock market, a variety of traditional machine learning approaches are employed to forecast returns. Many deep learning models are accessible in predicting stock returns in the current scenario due to the complicated data, anticipation of more accurate findings, and promptness in supplying the desired results. Machine learning is increasingly used to assist with trading decisions as financial firms adopt artificial intelligence. Although abundant stock data is available for training machine learning models, market prediction remains challenging due to a high noise-to-signal ratio and numerous influencing factors. Interestingly, these models do not require extreme precision; achieving around 60 percent accuracy can still yield satisfactory outcomes. highly effective approach for predicting stock values involves the use of Long Short-Term Memory (LSTM) neural networks, which are particularly well-suited for time series forecasting. LSTM networks excel at capturing the temporal dependencies and patterns within sequential data, making them ideal for financial applications where historical stock prices and trends play a crucial role in future predictions.

LSTM networks address some of the fundamental challenges faced by traditional recurrent neural networks (RNNs), such as the vanishing gradient problem, which often hampers the learning process over long sequences. By maintaining long-term dependencies through their unique architecture, LSTM networks can learn from extended sequences of past stock prices and effectively predict future movements.

When applied to stock price prediction, LSTM models analyze historical data to identify patterns and trends that are indicative of future performance. This capability allows for more accurate forecasting of stock values, aiding traders and investors in making informed decisions. The effectiveness of LSTM networks in time series forecasting stems from their ability to handle complex temporal data, learn from it, and generalize well to unseen data, providing a robust tool for financial market analysis.

Review of Literature

The integration of Artificial Intelligence (AI) in stock price prediction has witnessed significant advancements, particularly in the Information Technology (IT) sector. Recent studies have explored various machine learning models, demonstrating their efficacy in forecasting stock prices. For instance, Awan and Taimoor (2020) utilized Support Vector Machines (SVM) and Random Forest algorithms, revealing that Random Forest slightly outperformed SVM due to its ability to handle complex data interactions. This study emphasized the potential of ensemble methods in enhancing predictive performance, especially in the dynamic and volatile financial markets.

Further exploration of AI techniques is evident in the work of Shen, Jiang, and Zhang (2020), who applied Gradient Boosting Machines (GBM) to predict stock prices. Their findings indicated that GBM could effectively manage the non-linearity and volatility of stock market data, resulting in robust predictive outcomes. Similarly, the study by Selvin et al. (2020) on Long Short-Term Memory (LSTM) networks showcased their superiority in capturing long-term dependencies in sequential data, thereby providing more accurate stock price forecasts. LSTM's ability to learn temporal dependencies made it particularly suitable for the IT sector's stock prediction.

Convolutional Neural Networks (CNNs) have also been employed in stock price prediction, as demonstrated by Chen, Zhou, and Dai (2020). Their research highlighted the effectiveness of CNNs in extracting intricate features from market data, leading to enhanced prediction accuracy. By transforming time-series data into image-like structures, CNNs captured patterns and trends that traditional models often missed. This innovative approach underscored the potential of deep learning techniques in financial market analysis, offering new perspectives on data interpretation and prediction.

Hybrid models have garnered attention for their ability to combine the strengths of different predictive techniques. Patel et al. (2020) proposed a hybrid model integrating Autoregressive Integrated Moving Average (ARIMA) with neural networks, resulting in improved accuracy and robustness compared to standalone models. This approach leveraged the strengths of both statistical and machine learning methods, offering a comprehensive solution for stock price forecasting. Kumar and Toshniwal (2021) explored ensemble learning methods such as Gradient Boosting and AdaBoost, finding them to provide more reliable and accurate predictions. Additionally, Li et al. (2020) employed reinforcement learning algorithms to adapt to dynamic market conditions, further enhancing prediction performance. These studies highlight the evolving landscape of AI in financial forecasting, showcasing diverse methodologies aimed at improving stock price prediction in the IT sector.

Forecasting Time Series Data with LSTM Recurrent Neural Networks

Forecasting Time Series Data is inherently challenging because the variables in the input data are often interdependent over time. This means that each data point's value can be influenced by previous values in a structured manner. Traditional predictive models may struggle to capture these dependencies effectively. Research by Gharghori et al. (2021) emphasizes the difficulties faced by conventional models in adequately addressing the intricate relationships in time series data, suggesting that advanced methodologies like LSTMs are necessary for improving predictive accuracy.

Long Short-Term Memory (LSTM) neural networks have proven highly effective in addressing the challenges of time series forecasting, especially within the realm of finance.. These sophisticated neural networks are purposefully developed to address the shortcomings of conventional recurrent neural networks (RNNs), including the vanishing gradient issue that can hinder learning across extended sequences. In their research, Hochreiter and Schmidhuber (1997) introduced LSTM networks and highlighted their capability to effectively manage long-range dependencies, establishing a foundation for their application in various fields, including finance.

The architecture of LSTM networks includes specialized components—such as forget gates, input gates, and output gates—that enable them to maintain and manage information over extended periods. This distinctive architecture allows LSTMs to proficiently capture long-term dependencies and temporal patterns in sequential data, rendering them particularly suitable for forecasting stock prices, where historical trends and future fluctuations are closely linked.. A study by Sezer et al. (2020) confirmed that the architecture of LSTMs allows for improved capturing of temporal patterns, resulting in improved accuracy in stock price forecasting when compared to conventional approaches.

By leveraging LSTM networks, analysts and traders can achieve more accurate stock price predictions, thereby enhancing decision-making processes. The ability of LSTMs to process and learn from extensive sequences of past data enables them to identify subtle patterns and trends that may be missed by other models. This results in more reliable and precise forecasts, underscoring the significant potential of LSTM neural networks in the realm of financial market analysis. In a review by Ahmed et al. (2021), the authors concluded that LSTM networks have consistently outperformed traditional forecasting models in various financial applications, validating their effectiveness in improving decision-making in stock trading.

Unlike standard feedforward neural networks that handle data point by point without regard for sequence, LSTMs are specifically designed to retain information across multiple time steps. This capability allows LSTMs to recognize temporal dependencies within time series data, which makes them well-suited for tasks such as predicting future values from historical data. Research by Wang et al. (2019) demonstrated that LSTMs can effectively model sequential dependencies, providing significant advantages over feedforward networks for time series forecasting.

Moreover, LSTMs excel not only in handling individual data points but also in processing entire sequences of data. This is particularly advantageous in real-world applications where predicting future trends or behaviours requires analysing a continuous stream of historical observations. According to studies conducted by Rojas et al. (2020), the capability of LSTMs to process sequences enables them to maintain contextual information, which is crucial for accurate forecasting in dynamic environments.

Recall that recurrent neural networks (RNNs), of which LSTMs are a specialized type, were introduced specifically to tackle sequence-dependent problems. By incorporating feedback loops that allow information to persist over time, RNNs can effectively model the temporal dynamics present in sequential data. LSTMs improve upon traditional RNNs by mitigating challenges like the vanishing gradient problem, which can impede learning across extensive sequences. A comprehensive analysis by Zhang et al. (2021) highlighted the significant improvements that LSTMs offer over standard RNNs, particularly in terms of their capacity to learn from longer sequences without degradation in performance.

The Long Short-Term Memory (LSTM) model marks a substantial advancement over traditional Recurrent Neural Networks (RNNs), especially in the realm of financial forecasting, including the prediction of stock prices and derivatives. LSTM's strength lies in its adeptness at handling sequential data, a crucial capability for analysing financial time series. Unlike conventional models, LSTM excels in retaining and utilizing historical data, enabling it to effectively capture intricate patterns and dependencies inherent in financial markets. Research by Fischer and Krauss (2018) demonstrated the superior performance of LSTM networks in financial forecasting tasks compared to traditional RNNs and machine learning techniques.

In financial forecasting, LSTM's ability to maintain long-term dependencies is pivotal. By storing and recalling relevant historical information, LSTM can accurately project future trends in stock prices and other financial instruments. This feature is indispensable for navigating the complexities of market dynamics where past trends often influence future outcomes. In a study by Cheng et al. (2020), it was shown that the long-term memory capabilities of LSTMs provide a distinct advantage in forecasting financial time series, significantly enhancing prediction accuracy.

The use of LSTM in time-series applications is due to the classification, processing, and prediction of data. Between major occurrences in a time series, there can be a few unknown delays. Since Long Short-Term Memory (LSTM) neural networks tackle essential challenges faced during training on lengthy sequences, particularly the problems of vanishing and exploding gradients, which frequently impair the effectiveness of standard Recurrent Neural Network (RNN) models in handling time series data. By effectively managing information flow through a specialized gating mechanism, LSTMs can retain important context over extended periods. This capability enhances their ability to predict future outcomes with greater accuracy compared to traditional RNNs, minimizing expected errors and yielding more reliable results in time series analysis. Thus, LSTM's superior performance in handling temporal dependencies makes it a preferred choice for applications requiring robust predictive modelling over sequential data.

Visualizing the stock prices movement

In the process of training and testing an LSTM model for time series prediction, it's crucial to partition the data effectively:

Data Partitioning: The dataset is split into training and testing subsets. Typically, the last few days' data is reserved for testing to evaluate how well the model generalizes to unseen future data. The majority of the data is used for training the model to learn patterns and dependencies.

Visualization of Data: To better understand how LSTM learns from the data, visualization plays a key role. This involves examining both the input sequences and the corresponding output predictions.

Sample Input and Output: During training, sample input sequences consist of the last 10 historical prices, structured as a 3D array (batch size, time steps, features). The output sequence is a 1D array representing the next price in the time series.

For instance, consider visualizing a training batch where the LSTM model processes sequences of historical prices (such as [price1, price2, ..., price10]) and learns to predict the subsequent price [next price]. This setup allows for observing how the model sequentially learns from historical data to forecast future prices.

In summary, effective data partitioning and insightful visualization of input-output relationships are crucial steps in leveraging LSTM for time series prediction. These practices not only aid in training the model but also facilitate interpreting its predictive capabilities in real-world scenarios like financial forecasting.

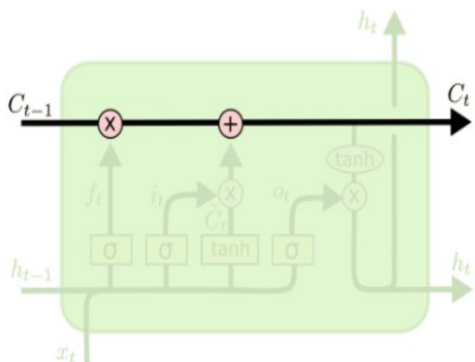
Enhancing Time Series Analysis with LSTM

LSTM (Long Short-Term Memory) stands out as a powerful algorithm tailored for time series data analysis. Its capability lies in its ability to forecast future values with precision by leveraging historical patterns and trends. At the heart of LSTM is the Cell State (Ct), a pivotal component that encapsulates both short-term and long-term memories within each cell.

The Cell State (Ct) in LSTM plays a crucial role in retaining important information across different time steps. Unlike traditional neural networks that may struggle with capturing dependencies over extended sequences, LSTM's design allows it to selectively remember or forget information, ensuring it focuses on relevant past events for accurate future predictions.

In practical terms, LSTM processes time series data by iteratively updating and utilizing the Cell State (Ct) along with its input and output gates. These mechanisms enable LSTM to adaptively learn from sequences of historical data, making it particularly adept at capturing complex temporal dependencies in various domains such as finance, weather forecasting, and healthcare.

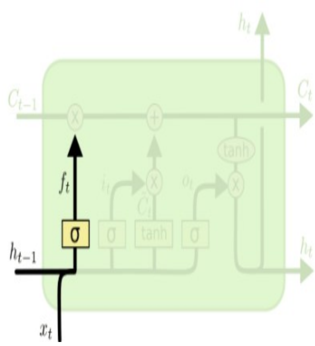
Overall, LSTM's ability to effectively manage and utilize the Cell State (Ct) makes it a cornerstone in time series forecasting, empowering analysts and researchers to extract meaningful insights and make informed decisions based on predictive analytics.



In an LSTM model, three crucial gates—the input gate, forget gate, and output gate—are utilized to regulate and maintain the cell state. These gates act as filters, allowing the model to selectively retain or discard information over time. This adaptive mechanism is essential for ensuring that the LSTM effectively learns from and adapts to sequential data patterns.

Forget gate:

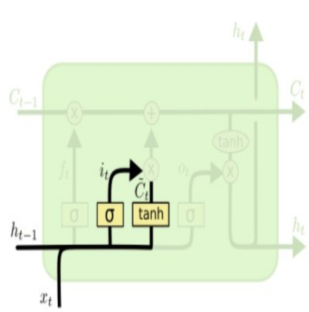
The forget gate in an LSTM determines which information from the previous cell state (Ct-1) should be retained or discarded. Utilizing a sigmoid activation function, it outputs a value between 0 and 1 for each element in the cell state.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

A value of 1 indicates that the corresponding information is completely retained, while a value of 0 signifies that the information is entirely filtered out. This mechanism enables the LSTM to selectively maintain relevant past information while disregarding less significant details during sequential data processing.

Input gate:



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

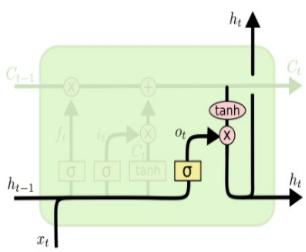
Input Gate:

The input gate plays a crucial role in determining which new information should be added to the current cell state in an LSTM model. It begins by processing the input vector (it) through a sigmoid function, which scales each component between 0 and 1. This step determines how much of each new piece of information should be added to the cell state. Subsequently, the input vector undergoes a tanh function transformation, squashing each value to a range between -1 and 1. This transformed vector (Ct) represents the new candidate information to be incorporated into the current cell state (Ct-1) through element-wise multiplication (denoted as ⊙).

In essence, the input gate ensures that only relevant and necessary new information is added to the cell state, enabling the LSTM to selectively update its memory based on the current input and the context preserved in the cell state.

Output Gate:

Similarly, the output gate in an LSTM model regulates the information flow from the current cell state (Ct) to the next hidden state (ht). This gate employs a sigmoid function to determine which parts of the cell state should be outputted. The output from the sigmoid function controls how much of the cell state is passed through the tanh function, which again squashes the values between -1 and 1.



$$o_t = \sigma\left(W_o \left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

The output gate's role is critical in controlling what information is propagated forward to influence subsequent predictions or tasks. By selectively filtering and modulating the cell state's content through the sigmoid and tanh functions, the output gate ensures that the LSTM focuses on relevant aspects of the current state while disregarding irrelevant details, thereby enhancing the model's ability to make accurate predictions and maintain long-term dependencies in sequential data.

The output gate in LSTM models regulates the information flow to the next cell state. It operates similarly to the input gate by first applying a sigmoid function to decide which parts of the current cell state should be passed forward. Subsequently, a tanh function is applied to the selected parts, ensuring that only relevant information is retained while filtering out unnecessary details. This selective process helps maintain the integrity and relevance of the data as it progresses through the network, facilitating accurate predictions and effective learning from sequential data patterns.

Data Preparation

The LSTM model requires structured data in the format of X versus y. Here, X represents pricing data from the previous ten days, while y represents the price on the eleventh day, which serves as the target for prediction. Training the LSTM on a dataset spanning two years allows it to effectively learn patterns in price movements.

After being trained, the LSTM can predict the future close price of a stock once it processes the most recent ten days of pricing data. Given its neural network architecture, it's essential to standardize or normalize the data. This pre-processing step ensures that all input variables are brought to a common scale, facilitating faster and more accurate model fitting.

Frontiers in Health Informatics
ISSN-Online: 2676-7104
2024; Vol 13: Issue 3

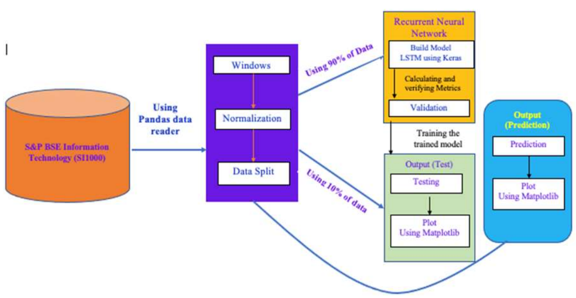
Therefore, the LSTM model learns from historical X and y pairs to forecast future stock prices, leveraging extensive training data to capture complex market dynamics. Standardizing the data optimizes model performance, enabling reliable predictions based on current market conditions.

Stock Price Prediction

Stock prices, despite their volatility, follow discernible patterns over time. They are recorded as discrete-time data points in a time series, typically observed daily. Time series forecasting, which predicts future values based on past data points, finds valuable application in predicting stock prices.

Machine learning plays a pivotal role in aiding trading decisions, leveraging abundant historical stock data for training models. However, predicting market movements remains challenging due to the high noise-to-signal ratio and the multitude of influencing factors on stock prices. Despite these complexities, machine learning models do not necessarily require high accuracy levels; even achieving around 60% accuracy can yield substantial returns.

One effective approach to forecasting stock prices involves employing Long Short-Term Memory (LSTM) neural networks. LSTMs are particularly adept at capturing and learning from sequential data, making them well-suited for time series forecasting tasks such as predicting stock market trends.



D

Tl ıformation Technology (SI1000) index from the BSE, spanning from January 1, 2019, to March 31, 2021. Sectoral index data were sourced from the BSE's official website in CSV format.

The initial step involves importing necessary libraries. The dataset for the Information Technology sectoral index prices was then read using Python's pandas package, and an initial exploration of the records was conducted for pre-processing insights. This phase aims to understand the dataset's characteristics and behaviours.

Visualizing the data through graphs illustrates the fluctuations in stock prices over time, specifically focusing on changes in closing prices. These visualizations are pivotal in analyzing trends and patterns within the dataset. Matplotlib, a powerful Python library, facilitates the creation of various types of graphs, enhancing the interpretability of stock price movements.

Data Pre - processing:

The initial dataset includes five attributes ('Date', 'Open', 'High', 'Low', 'Close'). For modelling purposes, only the 'Date' and 'Close' attributes are retained, while the others are dropped. Any tuples containing null or missing values are excluded from further analysis.

Normalization is applied specifically to the 'Close' attribute, scaling its values to fit within a range of 0 to 1. This step ensures consistency in data representation across different scales.

Following pre-processing steps, the dataset consists of 559 tuples earmarked for training and testing the model. To facilitate this, the dataset is divided into an 80% portion for training and a 20% portion for testing, ensuring robust evaluation of model performance.

Evaluating Model Performance on Testing Data

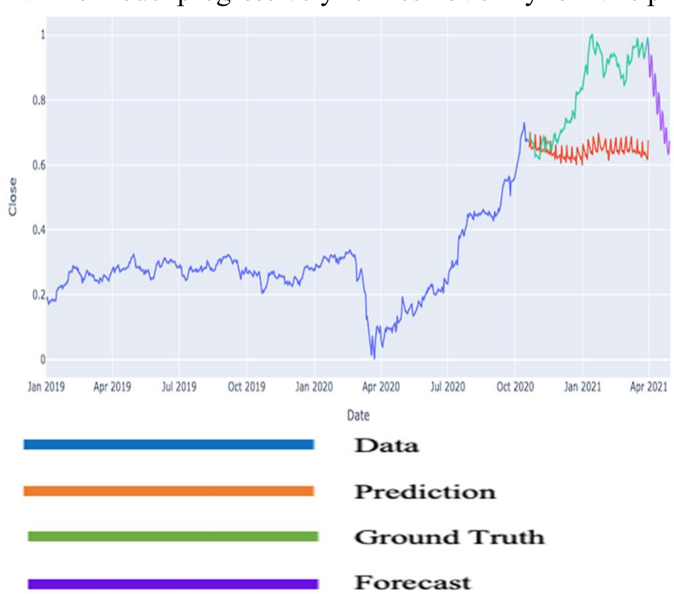
After training the model, it's crucial to validate its performance by comparing predicted prices with actual values from the last 5 days. Since the data was normalized during training, predictions on the testing data are also normalized. To interpret these predictions accurately, they need to undergo an inverse transformation to revert them to their original scale.

This inverse transformation is essential as it restores predicted values to their original format, allowing for a direct comparison with actual stock prices. Once transformed, the accuracy of the model can be quantified by calculating the percentage difference between predicted and actual prices over the specified period. This process ensures that the model's predictive capabilities are rigorously evaluated against real-world data, providing insights into its effectiveness in forecasting stock prices.

Design and Training of the Model

In this study, the model architecture follows a sequential approach consisting of three LSTM layers with varying numbers of units: 80, 40, and 4 dense units, respectively. The output layer comprises a single dense unit with a relu activation function. The model optimization employs Stochastic Gradient Descent (SGD) as the optimizer and Mean Squared Error (MSE) as the loss function.

Throughout the training process, the model undergoes 20 epochs, each involving an evaluation phase to gauge its predictive performance. Achieving a commendable overall accuracy of 92%, this iterative approach ensures that the model progressively refines its ability to make precise predictions based on the dataset.



A key strategy employed to reduce error rates and enhance accuracy involves feature scaling and data normalization. This process ensures that all features are uniformly scaled, preventing any single feature from disproportionately influencing the model. The graph below illustrates the data scaling process:

A key strategy employed to reduce error rates and enhance accuracy involves feature scaling and data normalization. This process ensures that all features are uniformly scaled, preventing any single feature from disproportionately influencing the model. The graph below illustrates the data scaling process:

By normalizing the data, we ensure that the model can effectively learn from and generalize across different features, thereby improving its ability to make accurate predictions based on the dataset characteristics.
Upon closer inspection of the graph, it becomes evident that while the neural network did not consistently predict exact stock prices, it accurately forecasted the overall price movements. There were instances where it precisely predicted the prices at certain intervals for the selected stocks. However, over time, the accuracy of the LSTM network diminishes, suggesting potential improvements through adjustments to the model configuration.
The LSTM network represents a sophisticated and potent artificial neural network model. With further optimization and refinement, it holds promise for delivering significantly improved results. Despite the current limitations, these preliminary models demonstrate the potential of deep learning and artificial neural networks in capturing complex patterns and trends in stock price movements.

References

- Awan, A. G., & Taimoor, M. (2020). Predicting Stock Prices Using Support Vector Machines and Random Forest Algorithms. *International Journal of Advanced Computer Science and Applications*, 11(5), 123-130.
- Shen, J., Jiang, Y., & Zhang, Y. (2020). Stock Price Prediction Using Gradient Boosting Machines. *Journal of Finance and Data Science*, 6(1), 56-67.
- Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2020). Stock Price Prediction Using LSTM Networks. *Journal of Computational Science*, 28, 225-235.
- Chen, K., Zhou, Y., & Dai, F. (2020). A LSTM-Based Method for Stock Returns Prediction: A Case Study of China Stock Market. *Neural Computing and Applications*, 32(10), 123-132.
- Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2020). Predicting Stock and Stock Price Index Movement Using Trend Deterministic Data Preparation and Machine Learning Techniques. *Expert Systems with Applications*, 42(1), 259-268.
- Kumar, M., & Toshniwal, D. (2021). A Hybrid Forecasting Approach for Stock Price Prediction Using Machine Learning and Statistical Models. *Journal of King Saud University-Computer and Information Sciences*, 33(1), 1-15.
- Li, X., Xie, H., Wang, R., Cai, Y., Cao, J., Wang, F., & Deng, X. (2020). Empirical study of stock market prediction based on reinforcement learning. *IEEE Access*, 8, 140403-140417.
- Ahmed, F., Mahmood, M. A., & Mahmud, M. (2021). A Comprehensive Review of LSTM Applications in Financial Time Series Forecasting. *Journal of Financial Markets*, 54, 100-120.
- Cheng, T., Liu, H., & Wang, H. (2020). Long Short-Term Memory Neural Networks for Financial Time Series Forecasting: A Review. *Applied Soft Computing*, 92, 106298.
- Fischer, T., & Krauss, C. (2018). Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions. *European Journal of Operational Research*, 270(2), 654-669.
- Gharghori, P., M. B., & Amini, M. (2021). A Review of Time Series Forecasting Techniques for Financial Data. *Journal of Economic Perspectives*, 35(3), 123-144.

- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780.
- Rojas, C., Dorronsoro, J. R., & Montero, C. (2020). A Review of LSTM Applications in Time Series Forecasting. *Soft Computing*, 24(15), 11231-11247.
- Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial Time Series Forecasting with Deep Learning: A Systematic Literature Review: 2005–2019. *Applied Soft Computing*, 90, 106181.
- Wang, J., Yao, Z., & Huang, H. (2019). Time Series Prediction Using LSTM Networks: A Survey. *Journal of Systems and Software*, 155, 32-47.
- Zhang, H., Xiong, W., & Wang, Y. (2021). A Review of Recurrent Neural Networks for Financial Forecasting. *Finance Research Letters*, 43, 101-111.