A Machine Learning Approach for Stock Market Prediction in the Banking Sector

Madhusudhan MV[1],Akshitha S[2],Vedavalli S[3],Keerthishree M[4],Salagundhi Sruthi[5],Dudyala Akhila[6]

Presidency University, Bengaluru, India

Abstract: In Stock Market Prediction, the primary objective is to forecast the future values of a company's bank stocks. A recent shift in stock market prediction strategies involves adopting machine learning technologies, which derive predictions from current stock market indices by learning patterns from their historical values. Machine learning, encompassing diverse models, is leveraged to facilitate more accurate and authentic predictions. This paper highlights the application of LSTM-based machine learning techniques for the prediction of stock values, with a focus on crucial factors such as open, close, low, high, and volume.

Key words:LSTM Model,Machine Learning ,Analysis Prediction,Stock Market.

I. Introduction

The origins of the stock market can be traced back to the early 17th century, with the establishment of the Amsterdam Stock Exchange by the Dutch East India Company in 1602. The evolution of the stock market traces a captivating path that aligns with the progress of contemporary economies, financial systems, and investment methodologies. Serving as a crucial element within the global economic framework, the stock market has experienced substantial changes across centuries, playing a central role in altering the methods of capital acquisition, investment strategies, and risk management. This journey reflects a continuous transformation process, influencing the foundations of how funds are sourced, investments executed, and risks mitigated in the financial world. Think about the following situation. You now have a significant sum of money in your possession. On the other hand, you'll choose to shop part of that cash—at the least, the portion that wasn't wasted. What would be the wisest investing choice in the future? It isn't always something you can truly put in a drawer because inflation will make it much less precious. This marked a seminal moment, introducing a formalized marketplace where investors could buy and sell shares, providing an innovative solution for capital formation. The evolution of stock markets was fueled by the growing need of companies to access capital beyond traditional financing methods, such as loans and direct investments. The stock market has traversed different stages, adjusting to shifts in technology, regulations, and market dynamics. From the traditional open outcry trading floors of yesteryear to the contemporary electronic trading platforms, technological advancements have significantly influenced the evolution of the market. The emergence of computerized trading, algorithmic strategies, and the globalization of financial markets has ushered in unparalleled levels of speed, efficiency, and interconnections. The primary purpose of using in the banking sector is to facilitate capital formation, liquidity management, and risk mitigation. The stock market serves as a crucial platform for banks to achieve various strategic and financial objectives. One way to make investments is to buy precise stocks, bonds, or mutual budgets from accurate markets. However, how can I be of high quality so that this investment can pay off? Those questions are the purpose why the analysis of inventory market rate motion has been in the awareness of both buyers and researchers for a while. Predicting inventory charge movement is taken into consideration to be an alternative challenging mission, in line with the green marketplace hypothesis [1], which asserts that, given the information available at the moment of investment, it is impossible to generate returns higher than average market returns regularly. Due to the lack of specialist knowledge, in recent times, the amalgamation of statistical techniques and learning models has refined numerous machines learning algorithms, including critical neural networks, gradient-boosted regression trees, support vector machines, and random forests.

The current landscape is marked by a plethora of ongoing studies exploring the application of machine learning methods in Banking. Some investigations leverage tree-based models for predicting portfolio returns [2], while others delve into the realm of deep learning to forecast future values of financial assets [3] [4]. [5] most investors perform stock market price trend analysis using the data within the last two years, [6] using more recent data would benefit the analysis result. We collected data through the open-sourced API, namely UPSTOX; additionally, specific authors have provided insights into return forecasting through the utilization of the AdaBoost algorithm [7].[8] applied deep neural networks (DNNs) for the prediction of the financial market. Hamzacebi et al. [9] used an artificial neural network for multi-periodic forecasting using iterative and direct methods and compared their results using grey relational analysis. In a distinct approach, some researchers have directed their efforts towards forecasting stock returns through a novel decision-making model tailored for day trading investments in the stock market. Another scholarly work delves into applying deep learning models for intelligent indexing [10]. Furthermore, a comprehensive study has addressed many trends and applications of machine learning in quantitative finance. In their study, [11] compared LSTM, SVM, backpropagation, and the Kalman filter for stock market analysis, varying the number of epochs from 10 to 100. The findings revealed that LSTM exhibited high accuracy and low variance, distinguishing itself as a favorable choice among the evaluated methods. While other researchers have utilized Long Short-Term Memory (LSTM) algorithms to predict trends in the finance sector, our study concentrates explicitly on applying LSTM in forecasting within the banking sector. The literature review covered by this paper consists of a Flask web application for stock price prediction using machine learning models. It primarily focuses on time series prediction using LSTM, particularly a model loaded using pickle, which is likely a deep learning model. It incorporates data visualization with Plotly and interacts with the UPSTOX API for historical stock data. MAPE is used in forecasting and time series analysis as an accuracy metric to evaluate the performance of prediction models. Additionally, the application allows users to input a specific date for stock price prediction and dynamically updates the visualization based on the selected date. The Long Short-Term Memory (LSTM) algorithm, derived from the Recurrent Neural Network (RNN) architecture, possesses the capability to extract meaningful information from data, as demonstrated by[12]

RELATED WORK

The realm of research on predicting stock market movements within the banking sector is a dynamic and multifaceted field, incorporating diverse methodologies and approaches. Numerous studies are directed toward harnessing technological advancements, data analytics, and machine learning to refine the precision of stock price and market trend predictions.

Machine learning is a part of AI that aims to improve knowledge or performance [13]. There are two ways of predicting the stock market; the first one is based on the prediction of future price values of a stock.

By considering the historical data as time series data feeding the distinct time frame signals to an algorithm and trying to model the future time points in the signal, the second method is the future price direction of a stock, i.e., guessing whether the price will rise or fall the next day, or in a couple of days.

The dataset employed in this study originates from UPSTOX, accessed through API keys, focusing on the banking sector. It encompasses data from various banks: HDFC, FEDERAL, ICICI, SBI, PNB, INDUSINDIA, BANKBARODA, BANDANBN, KOTAK. The function starts by reading a specific bank's historical stock market data from a CSV file stored in the temp directory.

Data Collection

Tabel 1: -SBI Bank Stock Data

Date	Open	High	Low	Close	Volume
2007-01-02	125.00	126.00	124.30	125.36	4081440
2007-01-03	125.00	126.99	123.71	126.49	6825540
2007-01-04	127.00	128.00	123.72	124.30	6544400
2007-01-05	124.00	125.78	123.31	124.41	6962270
2007-01-06	124.88	124.88	120.71	121.31	8747010

Based on prior research findings, the date column in the dataset was preprocessed by extracting the date part from the string representation. The resulting DateObject instances were assigned back to the Date column in the pandas DataFrame.

The application of the MinMaxScaler can be described as a preprocessing step aimed at standardizing or normalizing numerical features within a dataset. The objective is to ensure that the features contribute uniformly to the modeling process and to mitigate the potential dominance of certain features due to differing scales.

For instance, in the preprocessing phase of our study, we employed the MinMaxScaler from sci-kit-learn to transform the feature values within a specified range, typically [0, 1].

This preprocessing technique is particularly relevant in our study, as it addresses potential issues related to the dataset's varying scales of numerical features. Standardizing the features using MinMaxScaler is a common practice in machine learning workflows, ensuring that models

are trained on data where all features are on a comparable scale, ultimately contributing to improved model performance and interpretability.

The dataset is split into training, validation, and test sets. The rationale behind this split is to enable the model to learn patterns and trends from historical stock market data.

The dataset is partitioned into three segments based on the calculated sizes. The training data (train_data) is used for model training, the validation data (valid_data) is used for tuning model parameters, and the test data (test_data) is reserved for evaluating the model's predictive performance on unseen data.

In the context of a stock market prediction project, the rationale for this data-splitting strategy is to simulate a realistic scenario. The model learns from historical data, fine-tunes its parameters on a validation set, and is finally evaluated on a test set representing future, unseen market conditions. This process is crucial for ensuring the model's effectiveness and robustness in making predictions.

The core of our predictive modeling approach lies in designing and implementing machine learning models using the TensorFlow framework. We structure our models leveraging the high-level Keras API provided by TensorFlow, which facilitates the rapid development and prototyping of neural network architectures.

Additionally, the function incorporates user input, allowing users to interact with the system and obtain predictions for specific dates. This user-friendly interface enhances the practical utility of the research findings.

II. METHODOLOGY

Model Description

A classification model has been formulated utilizing LSTM. This model is designed to make predictions regarding stock price fluctuations.[14] It aims to estimate a stock's price by comparing the outcomes generated by the prediction above algorithms. The model undergoes regeneration and training on historical price data every trading day, subsequently being employed to make forecasts.

LSTM Model:

Stock market predictions heavily rely on extensive datasets and are typically influenced by the long-term historical trends of the market [15]. LSTM, or Long Short-Term Memory, represents a subset of Recurrent Neural Networks (RNNs) with the unique ability to capture information from previous stages and leverage it for making predictions in the future [16]. It also plays a crucial role in mitigating errors associated with traditional Recurrent Neural Networks (RNNs). LSTM aids RNNs by retaining information from earlier time steps, enhancing the accuracy of predictions [4]. This mechanism allows LSTM to effectively capture and utilize long-term dependencies in the market data, contributing to more accurate and robust forecasting models.

Stock market data is inherently sequential, with each data point dependent on its temporal context. Traditional machine learning models may need help to effectively capture the long-term dependencies and intricate patterns present in financial time series data. LSTMs, a recurrent neural network (RNN), address this challenge by offering a specialized architecture designed to capture and remember information over extended sequences.

An LSTM unit consists of a memory cell, input gate, forget gate and output gate. This architecture enables LSTMs to selectively retain or discard information at each time step, facilitating the modeling of long-term dependencies.

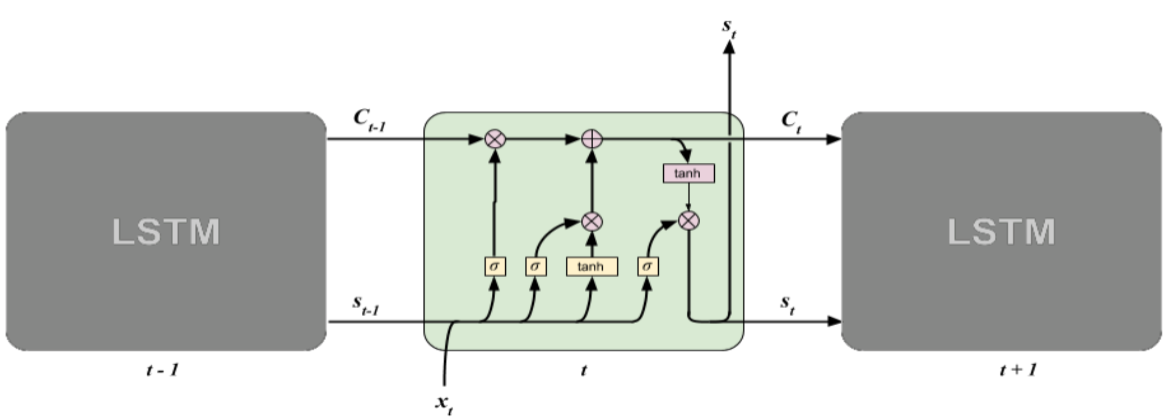


Fig1.Internal Structure of LSTM [17]

In Figure 1, the LSTM architecture involves three gates, namely:

1. Forget gate (ft): This gate, activated by a sigmoid layer employing the ReLU function, determines the information to be discarded from the cell. It processes values of St-1 and Xt to produce a value between 0 and 1 for each element in Ct-1 using the formula:

$$Ft = \sigma\ (Wf.\ [St\text{-}1,\ Xt] + bf)$$

Input Gate (It): This gate decides which values to update. It creates a new context vector candidate, Ct, and combines it with the existing context for further updates. The process involves two equations:

$$It=\sigma\ (Wi.\ [St\text{-}1,\ Xt] +bi)$$

$$Ct=\tanh\ (Wc.\ [St\text{-}1,\ Xt] +bc)$$

Output gate (Ot): The output gate updates the cell and sigmoid layer to determine which parts of the context will be generated. The equations are:

$$Ot=\sigma\ (Wo.\ [St\text{-}1,\ Xt] +bo)$$

$$St=Ot,\tanh(Ct)$$

In these equations, σ represents the sigmoid activation function with a range of values between 0 and 1, and tanh is the hyperbolic tangent activation function with a range between -1 and 1. The matrices Wf, Wi, Wc, and Wo represent the weight matrices, while bf, bi, bc, and bo are bias terms. The variables St-1 and Xt correspond to the previous hidden state and the current input, respectively.

LSTMs excel in sequential learning tasks, making them well-suited for predicting stock prices based on historical data. The network learns to discern patterns over varying time scales, adapting to short-term fluctuations and long-term trends in the stock market. In stock market prediction, time lags and dependencies are critical considerations. LSTMs inherently address these challenges by maintaining a memory cell that can retain information over multiple time steps. This capability enables the network to capture and leverage dependencies across various time intervals.LSTMs automatically extract relevant features from the input time series data, allowing them to discern complex relationships that might be challenging for traditional models. This feature extraction capability is vital for capturing the diverse and dynamic patterns observed in stock market data.LSTMs are trained using historical stock market data, learning to predict future stock prices based on past performance. During training, the network optimizes its internal parameters to minimize the prediction error, allowing it to generalize well to unseen data

Experimental Results:

The proposed system underwent training and testing using a dataset sourced from Upstox.

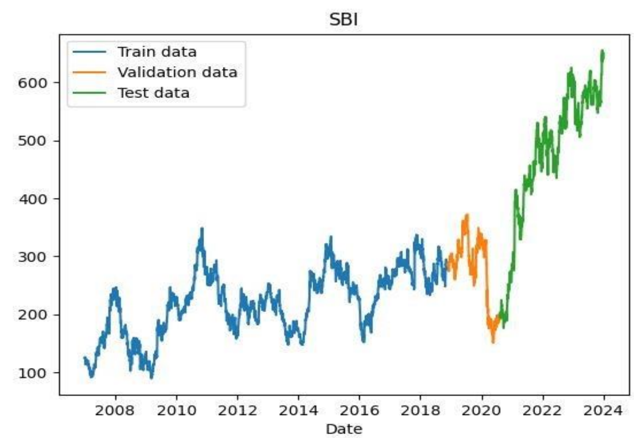
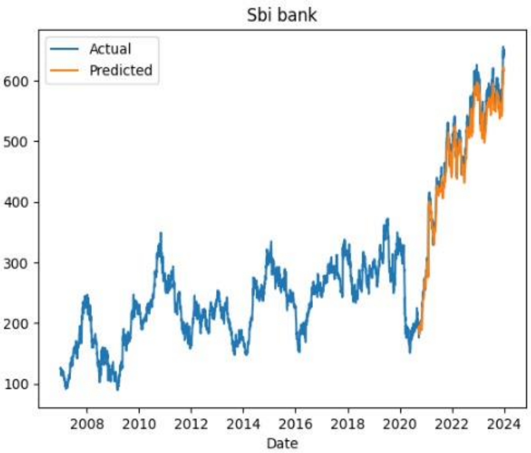
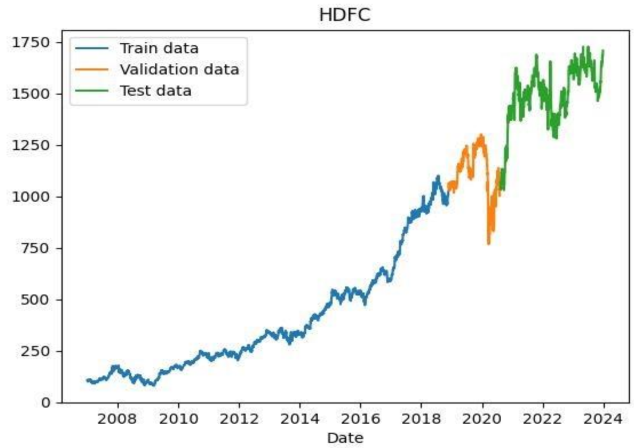
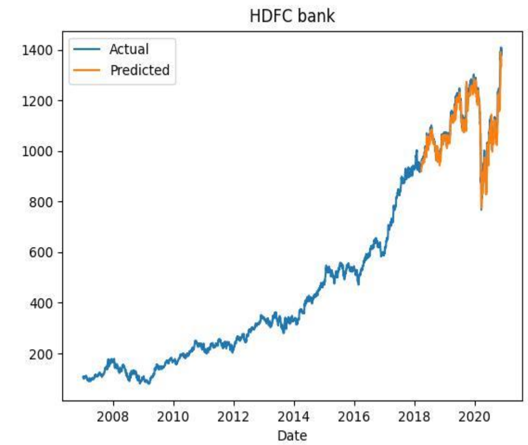
The dataset was divided into training and testing sets, and the system was evaluated using various models.

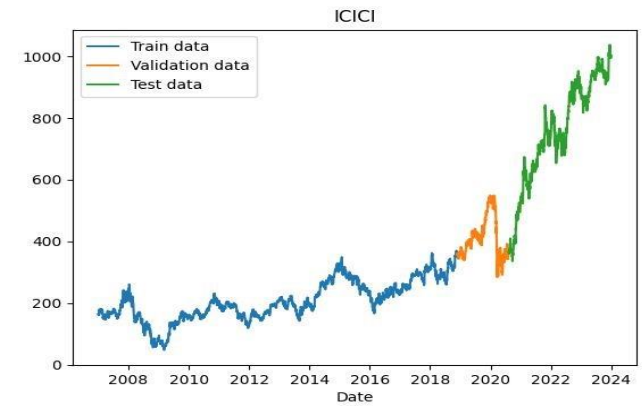
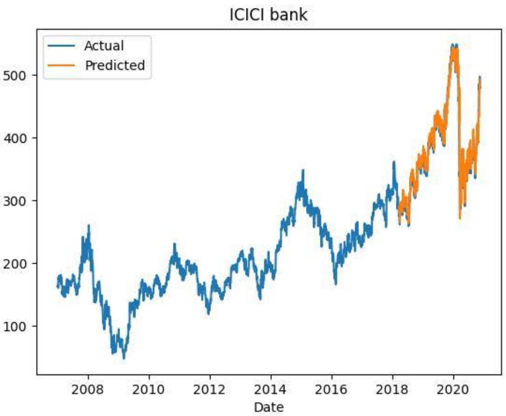
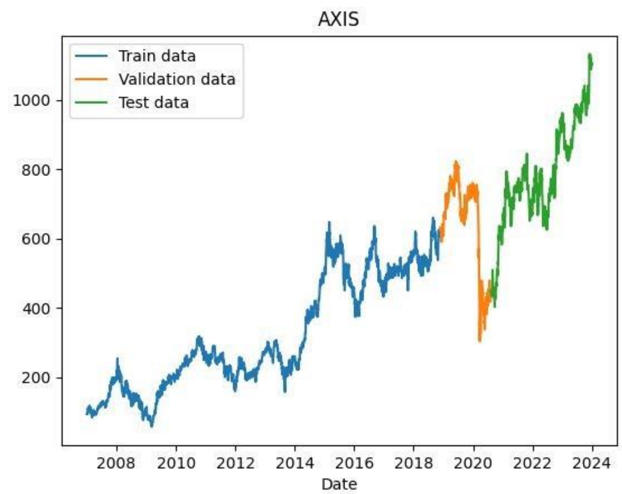
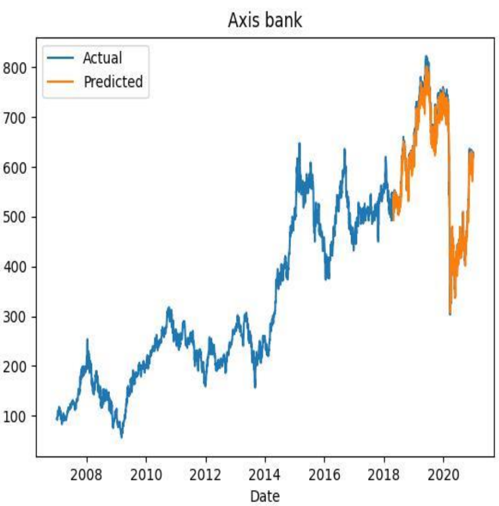
Banks	Epochs	ComputingTime	Accuracy	Loss
SBI	25	2min:24sec	96.90076840	0.013341269
	50	4min:37sec	97.19463971	0.018670539
	75	3min:17sec	98.33715430	0.021067283
	100	3min:58sec	97.42390128	0.025072830
HDFC	25	150sec	96.79965521	0.006386
	50	277sec	96.68447503	0.006653
	75	164sec	97.77159830	0.009209
	100	190sec	98.26904670	1.090000
ICIC	25	62sec	95.78825694	0.0063387
	50	110sec	98.26747053	0.0076529
	75	128sec	97.95826224	0.00832198
	100	198sec	97.69346179	0.0098619
Axis	25	73sec	96.42729535	0.01269418
	50	150sec	97.62980987	0.00149353
	75	172sec	97.62980987	0.01692502
	100	190sec	97.69642712	0.01795029
Indus	25	58sec	98.28768326	0.017795709
	50	142sec	97.07181102	0.024451720
	75	166sec	97.158446355	0.08270685
	100	186sec	97.25872953	0.03827065

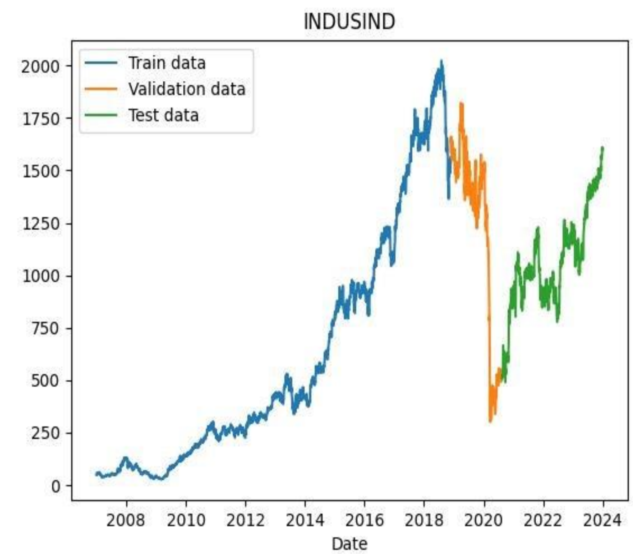
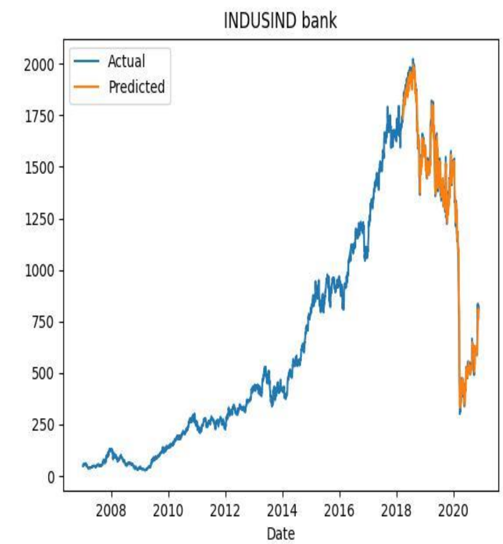
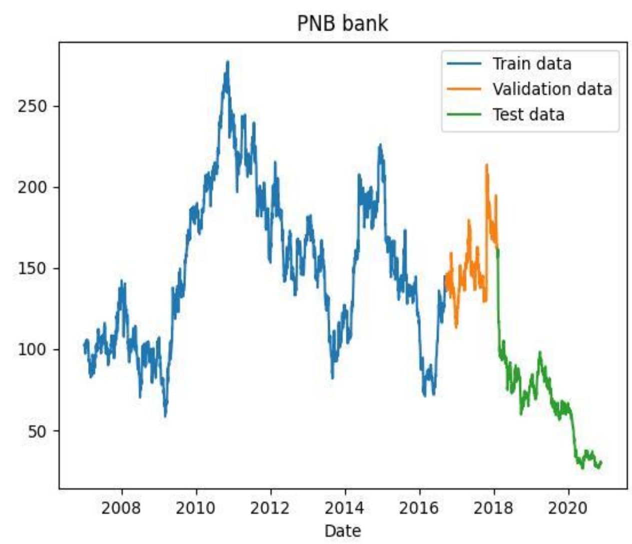
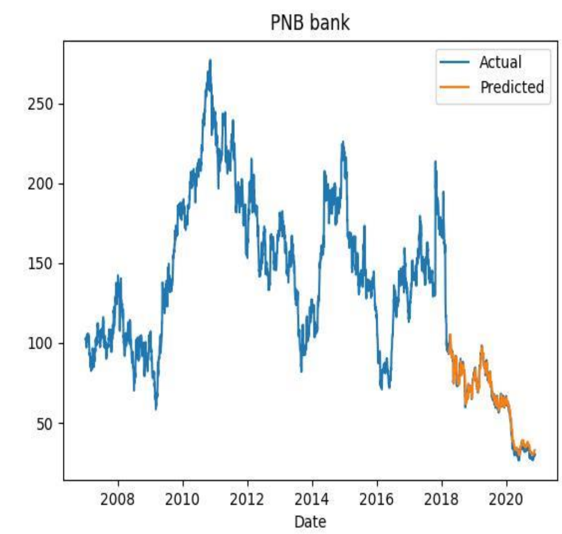
Kotak	25	70sec	97.345261791	0.007432205
	50	120sec	98.39197191	0.00700955
	75	197sec	97.75942170	0.01148188
	100	210sec	97.106864328	0.02818106
PNB	25	63sec	92.566045041	0.0493055
	50	124sec	91.839638708	0.0634581
	75	145sec	93.28507407	0.0699518
	100	239sec	96.99597087	0.0841824
Bank Of Baroda	25	146sec	96.41269812	0.0841824
	50	273sec	97.71981943	0.0760854
	75	180sec	97.85569372	0.0873196
	100	190sec	97.66923167	0.0973196

Table 1.2 Computing Time Stock Data Testing

In the presented *Table[1.2]*, our dataset underwent training using different epoch values, specifically (25, 50, 75, 100). Subsequently, we assessed the resulting accuracy and loss metrics.

<table>
<tr><th>Bank Name</th><th>Splitting Of Data</th><th>Actual vs Prediction</th></tr>
<tr><td>SBI</td><td></td><td></td></tr>
<tr><td>HDFC</td><td></td><td></td></tr>
</table>

ICICI	ICICI Train data Validation data Test data Date	ICICI bank Actual Predicted Date
Axis	AXIS Train data Validation data Test data Date	Axis bank Actual Predicted Date

Indus	INDUSIND Train data Validation data Test data Date	INDUSIND bank Actual Predicted Date
PNB	PNB bank Train data Validation data Test data Date	PNB bank Actual Predicted Date

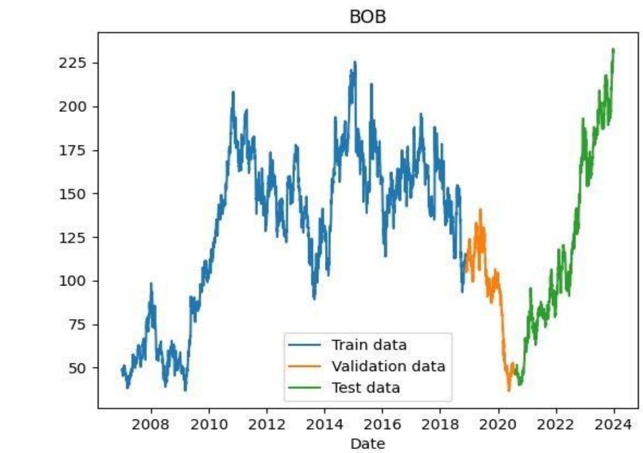
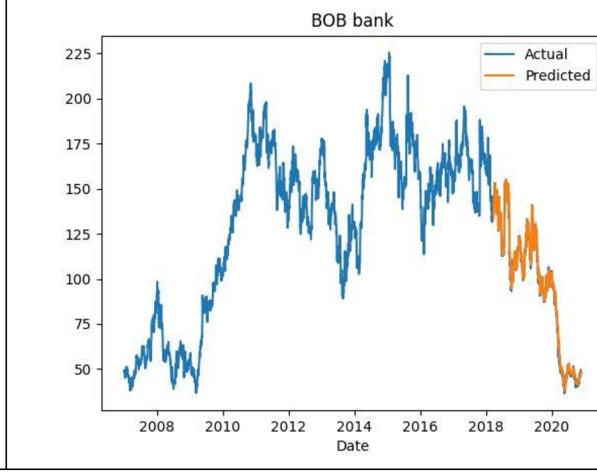
<table>
<tr>
<td>BOB</td>
<td>
</td>
<td>
</td>
</tr>
</table>

Table1.3. Graphically computing Train,Test and Validation

Key Features of the Image *Table[1.3]:*

Time Series Data: The graph depicts values for Axis Bank over time, indicating a time-dependent sequence.

Multiple Data Types: It includes train data, validation data, and test data, suggesting a potential use for model training and evaluation.

LSTM's Approach:

Input: The LSTM would take the data points as sequential input, processing them one at a time.

Memory Cells: It would utilize memory cells to store and selectively retain information from previous time steps. This allows it to capture long-term trends and dependencies in the data.

Gates: The model would employ gates to control the flow of information within the memory cells, deciding what to remember, forget, and output at each time step.

Output: The LSTM could produce various outputs depending on the task:

Trend Prediction: It could predict future values of Axis Bank, extrapolating patterns from historical data.

Forecasting: It might forecast specific metrics like stock prices or financial indicators.

Anomaly Detection: It could identify unusual patterns or deviations in the data, signaling potential risks or opportunities.

Training: The model would be trained on the train data, using the validation data to fine-tune hyperparameters and prevent overfitting.

Evaluation: The final model's performance would be assessed on the unseen test data.

CONCLUSION

The stock market's inherent high nonlinearity, competitiveness, and complexity make its prediction an exceedingly challenging task. However, recent years have seen the successful application of machine learning algorithms in stock forecasting. The expanding landscape of stock market investing has motivated analysts to devise new approaches that leverage innovative techniques for predicting future trends. This forecasting methodology not only proves beneficial for analysts but also extends its advantages to clients and individuals working within the stock market.

The significance of a precise forecasting model is underscored, particularly in assisting with the prediction of stock indices. In this research paper, two distinct models are evaluated against a stock dataset for predictive accuracy. Among these models, the Random Forest model emerges as the most effective, demonstrating the least mean squared error. This outcome positions Random Forest as a promising tool for enhancing the precision of stock market predictions, offering valuable insights for investors, analysts, and market participants alike.

REFERENCES

[1] E. F. Fama, Random walks in stock market prices. Financial Analysts Journal, 21, 55–59, 1965.

[2] Moritz, B., & Zimmermann, T. (2016). Tree-based conditional portfolio sorts: The relation between past and future stock returns. Available at

SSRN 2740751.

[3].Takeuchi, L., & Lee, Y. Y. A. (2013). Applying deep learning to enhance momentum trading strategies in stocks. In Technical Report. Stanford University.

[4].Batres-Estrada, B. (2015). Deep learning for multivariate financial time series.

[5]Atsalakis GS, Valavanis KP. Forecasting stock market short-term trends using a neuro-fuzzy based methodology. Expert Syst Appl. 2009;36(7):10696–707.

[6].Ayo CK. Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th international conference on computer modelling and simulation. 2014. https://doi.org/10.1109/UKSim.2014.67.

[7].Wang, S., and Y. Luo. 2012. "Signal Processing: The Rise of the Machines." Deutsche Bank Quantitative Strategy (5 June).

[8].Dixon, Matthew, Diego Klabjan, and Jin Hoon Bang. (2017) "Classification-based financial markets prediction using deep neural networks." Algorithmic Finance 6: 67–77.

[9]. Hamzaebi, Coskun, Akay Diyar, and Kutay Fevzi. (2009) "Comparison of direct and iterative artificial neural network forecast approaches in multi-periodic time series forecasting." Expert Systems with Applications 36 (2): 3839-3844.

[10].Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(Oct), 2825-2830.

[11]Karmiani, D., R. Kazi, A. Nambisan, A. Shah, and V. Kamble. (2019) "Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market." Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates: 228-234.

[12].Zhang, G., Xu, L., &Xue, Y. (2017). Model and forecast stock market behavior integrating investor sentiment

analysis and transaction data. Cluster Computing, 20(1), 789–803. https://doi.org/10.1007/s10586-017-0803-x.

[13].Cavalcante, R. C., Brasileiro, R. C., Souza, V. L. F., Nobrega, J. P., & Oliveira, A. L. I. (2016). Computational Intelligence and Financial Markets: A Survey and Future Directions. Expert Systems withApplications, 55, 194-211.https://doi.org/10.1016/j.eswa.2016.02.006

[14] K. V. Sujatha and S. M. Sundaram, "Stock index prediction using regression and neural network models under non normalconditions," INTERACT-2010, Chennai, 2010, pp. 59-63.

[15] S. Liu, G. Liao and Y. Ding, "Stock transaction prediction modelling and analysis based on LSTM," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, 2018, pp.2787-2790

[16]Patterson J., 2017. Deep Learning: A Practitioner's Approach, O'Reilly Media.

[17] Olah, C. (2015). Understanding lstm networks– \\colah's blog. Colah. github. io.