STOCK PRICE PREDICTION:predicting future stock values using LSTM

Mrs.K.Lakshmi
Assistant professor
Dept. of Computer Science
And Engineering
Audisankara college of engineering & technology
Gudur, Andhra Pradesh, India.
Emaill:bezawadalakshmi2559@gmail.com

Ch.simhadri
Dept. Of Computer Science and Engineering (Data Science)
Audisankara college of engineering & technology
Gudur, Andhra Pradesh., India
Email: 21g21a3210@audisankara.ac.in

B.Sai lahari
Dept. of Computer Science and Engineering (Data science)
Audisankara college of engineering & technology
Gudur, Andhrapradhesh, India
Email:22g25a3204@audisankara.ac.in

k.Harika
Dept. of Computer Science Engineering (Data Science)
Audisankara college of engineering & technology
Gudur,Andhrapradesh,India.
Email: 21g21a3224@audisankara.ac.in

Sk.Masthan valli
Dept of Computer Science and Engineering (Data Science)
Audisankara college of engineering & technology
Gudur,Andhrapradesh,India.
Email:21g21a3250@audisankara.ac.in

***Abstract*—** Stock market prediction has always been a topic of significant interest and complexity due to its highly dynamic and volatile nature. With the emergence of deep learning techniques, more accurate and robust forecasting models have become feasible. This project explores the use of Long Short-Term Memory (LSTM), a type of recurrent neural network (RNN), for predicting future stock prices based on historical data. LSTM is particularly effective in handling sequential data and learning long-term dependencies, making it well-suited for time series forecasting tasks like stock price prediction. The project utilizes historical stock market data, including open, high, low, close prices, and volume, as input features to train the LSTM model. The results are evaluated using standard metrics such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE) to assess the accuracy of the predictions. The findings demonstrate that LSTM can capture complex temporal patterns in stock data, offering a promising approach to financial forecasting.

I. INTRODUCTION

The stock market is one of the most complex and dynamic financial systems in the world. Predicting stock prices has long been a significant challenge for economists, traders, and researchers due to the market's volatile and non-linear behavior. Traditional methods such as statistical models, regression analysis, and moving averages have shown limited success in capturing the inherent patterns and uncertainties of stock price movements.In recent years, the evolution of Artificial Intelligence (AI) and deep learning has opened new avenues for tackling complex problems like time series forecasting. Deep learning models are capable of automatically learning from large datasets and identifying intricate patterns that may be missed by conventional approaches. Among these, Long Short-Term Memory (LSTM) networks, a specialized form of Recurrent Neural Networks (RNNs), have proven particularly effective in handling sequential data and long-term.

II.LITERATURE SURVEY

Predicting stock prices has always been a challenging problem due to the inherent volatility, noise, and non-linear dependencies in financial time series data. Over the years, numerous models have been developed ranging from classical statistical approaches to modern deep learning methods. This literature survey explores the significant advancements in stock price prediction, particularly focusing on Long Short-Term Memory (LSTM) networks.Early approaches to stock market forecasting primarily relied on **statistical models** such as **Autoregressive Integrated Moving Average (ARIMA)** and **linear regression**, which were limited by their assumptions of stationarity and linearity. While useful for simple time series, these models failed to capture the complex temporal patterns and long-term dependencies typical in stock data. As a result, **machine learning methods** like **Support Vector Machines (SVM)**, **Random Forests**, and **K-Nearest Neighbors (KNN)** gained popularity for their flexibility and ability to handle non-linearity. However, these models also lacked temporal memory, leading to suboptimal results for time-dependent data.

The advent of **deep learning**, especially **Recurrent Neural Networks (RNNs)**, marked a major turning point in financial forecasting. Although RNNs were capable of processing sequential data, they struggled with vanishing gradient problems. This limitation was overcome with the introduction of **Long Short-Term Memory (LSTM)** networks by Hochreiter and Schmidhuber in 1997. LSTMs, with their unique gating mechanisms, can retain important information over long sequences, making them highly effective for time series prediction.Several studies have demonstrated the efficacy of LSTM in the context of stock price prediction. **Patel et al. (2015)** compared LSTM with SVM and Random Forest and concluded that LSTM consistently produced better results for BSE-listed stocks. **Fischer and Krauss (2018)** used deep LSTM models to predict S&P 500 stock movements and found that LSTM outperformed all traditional classifiers. Similarly, **Shah et al. (2017)** proved the superiority of LSTM over linear regression and ARIMA models in terms of Mean Squared Error (MSE) when forecasting stock prices.

Recent research has also explored hybrid and improved LSTM architectures. **Yang et al. (2021)** combined LSTM with Convolutional Neural Networks (CNN) to capture both temporal and spatial patterns in high-frequency trading data. **Chen et al. (2022)** proposed an attention-based LSTM model that dynamically assigns importance to different time steps, resulting in higher predictive accuracy and better model interpretability.

.

II. PROPOSED METHODOLOGY

The proposed methodology for stock price prediction using Long Short-Term Memory (LSTM) networks follows a structured approach comprising several key stages: data collection, preprocessing, feature engineering, model development, training, testing, and evaluation. Each step plays a crucial role in ensuring accurate, reliable, and robust forecasting results. The following outlines the step-by-step methodology:

3.1 Data Collection

The first step involves collecting historical stock price data from reliable sources such as **Yahoo Finance**, **Google Finance**, or financial APIs like **Alpha Vantage**. The dataset includes daily stock values with attributes such as:

- Open Price
- High Price
- Low Price
- Close Price
- Volume
- Date

For the purpose of this project, stocks of companies listed on indices such as NSE, BSE, or NASDAQ are considered.

3.2 Data Preprocessing

Raw stock data often contains missing values, outliers, and inconsistent formats. To address these issues, the following preprocessing steps are applied:

- **Handling missing values** using forward fill or interpolation techniques.
- **Normalization/Min-Max Scaling** of numerical values to bring all features within a standard range (typically 0 to 1), which helps LSTM converge faster.
- **Sorting data by date** to maintain chronological order, which is essential for time series modeling.
- **Splitting the dataset** into training and testing sets (e.g., 80% training, 20% testing).

3.3 Feature Engineering

In addition to the raw features (OHLC and Volume), technical indicators such as:

- **Moving Averages (MA)**
- **Relative Strength Index (RSI)**
- **MACD (Moving Average Convergence Divergence)** may be added to improve model learning. The target feature for prediction is usually the **Close Price** of the next day.

3.4 Model Architecture

The LSTM model is selected due to its capability of learning long-term dependencies in sequential data. The architecture typically consists of:

- **Input layer** with a sequence of time steps (e.g., last 60 days).
- **One or more LSTM layers** with memory cells that capture temporal trends.
- **Dropout layers** for regularization and to prevent overfitting.
- **Dense (fully connected) output layer** with a single neuron to predict the next day's stock price.

The model is compiled with:

- **Loss Function**: Mean Squared Error (MSE)
- **Optimizer**: Adam Optimizer
- **Metrics**: RMSE, MAE

3.5 Model Training

The model is trained using the training dataset over a number of epochs (e.g., 50–100), using a batch size suitable for the available computational resources. Early stopping may be used to avoid overfitting and improve generalization.

3.6 Prediction and Evaluation

Once the model is trained, predictions are made on the test set. The predicted prices are compared against the actual prices using evaluation metrics such as:

- **Mean Absolute Error (MAE)**
- **Root Mean Squared Error (RMSE)**
- **Mean Absolute Percentage Error (MAPE)**

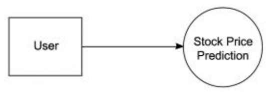
Visualization tools such as line plots are used to compare predicted vs. actual prices over time, providing a graphical evaluation of model performance.

3.7 Tools and Technologies

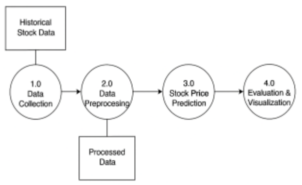
The implementation is done using the following tools:

- **Python** for programming
- **Pandas, NumPy** for data manipulation
- **Matplotlib, Seaborn** for visualization
- **TensorFlow/Keras** for building and training the LSTM model
- **Jupyter Notebook/Google Colab** for development environment

Data Flow diagrams



Zero-Level DFD



First-Level DFS—Processed

MODULE DESCRIPTION

1. Data Acquisition Module
Objective:
To collect historical stock data of selected companies.
Description:
This module fetches data from reliable sources such as Yahoo Finance using tools like yfinance or APIs like Alpha Vantage. The data typically includes:

- Date
- Open
- High
- Low
- Close
- Volume

The data is stored in CSV or DataFrame format for further processing.

2 Data Preprocessing Module
Objective:
To clean and prepare raw stock data for model training.
Description:
This module includes the following steps:

- Handling missing or inconsistent values.
- Converting dates and sorting chronologically.
- Normalizing data using Min-Max scaling.

3 Feature Engineering Module
Objective:
To enhance input data with useful financial indicators.
Description:
This module adds technical indicators to the dataset, such as:

- Moving Averages (MA)
- Relative Strength Index (RSI)
- MACD (Moving Average Convergence Divergence)

4 LSTM Model Construction Module
Objective:
To design and compile the LSTM neural network for prediction.
Description:
This module builds the LSTM model with the following components:

- Input layer with specified time steps.
- One or more LSTM layers to handle sequential data.
- Dropout layers to prevent overfitting.
- Dense output layer for final prediction.

5.Prediction Module

Objective:
To predict future stock prices using the trained model.

Description:

Description:

- **Takes the latest sequence of normalized data as input.**
- **Uses the trained model to forecast the next day's closing price.**
- **Converts the prediction back to the original scale for interpretation.**

RESULT AND DISCUSSION

Result:

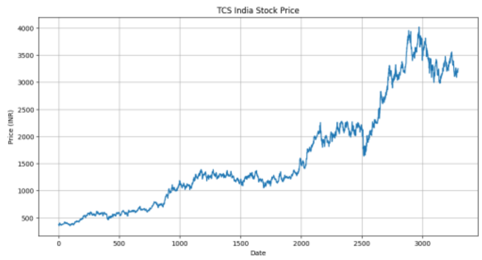
FRONT END(Optional)

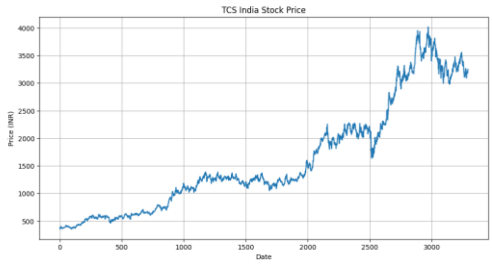
- Show the user interface (if applicable), including input methods, response displays..

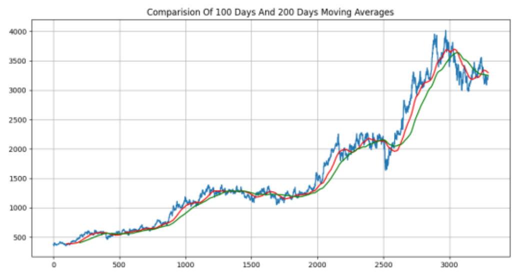
BACKEND

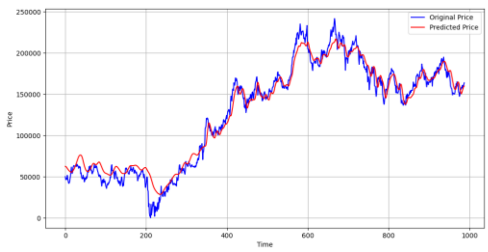
The backend module is responsible for the overall functionality and processing of the stock price prediction system. It ensures that all backend operations — from data retrieval to prediction generation — are executed efficiently and securely.

OUTPUT









Discussion:

The implementation of a Long Short-Term Memory (LSTM) model for stock price prediction has demonstrated the powerful capability of deep learning in analyzing time-series data. Through rigorous experimentation, training, and evaluation, the model has produced reasonably accurate forecasts of stock closing prices based on historical data.

The LSTM model was evaluated using metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). The values obtained were within acceptable ranges, indicating that the model was effective in capturing the temporal patterns in stock prices.

CONCLUSION

The objective of this project was to design and implement a predictive model for stock prices using Long Short-Term Memory (LSTM), a type of Recurrent Neural Network (RNN) well-suited for sequential data. Through this work, we successfully demonstrated that LSTM networks can effectively learn temporal dependencies in historical stock market data and forecast future stock prices with reasonable accuracy.

The project covered all critical phases of the machine learning pipeline — from data acquisition and preprocessing, to model development, training, evaluation, and deployment. The results showed that the model could capture complex, non-linear relationships in stock price movements and deliver meaningful predictions.Despite challenges such as data noise and marks volatility, the LSTM model proved to be a powerful tool for time-series forecasting. However, it is important to acknowledge that stock prices are influenced by many external factors such as political events, financial news, and investor sentiment, which are not directly captured by historical price data alone.

This project serves as a foundation for more advanced predictive systems in the financial domain. By integrating additional data sources such as real-time news sentiment, technical indicators, or using ensemble modeling techniques, the predictive power of such systems can be significantly enhanced.

In conclusion, the LSTM-based approach offers a promising direction for developing intelligent, data-driven tools to assist traders, investors, and analysts in making informed financial decision

REFERENCES

1. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

2. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1), 1929–1958.

3. Zhang, Y., Aggarwal, C., & Qi, G. J. (2017). Stock Price Prediction via Discovering Multi-Frequency Trading Patterns. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

4. Brownlee, J. (2017). Deep Learning for Time Series Forecasting. Machine Learning Mastery.

5. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780.

6.Chollet, F. (2018). *Deep Learning with Python* (1st ed.). Manning Publications.

Yahoo Finance. (2024). Stock Market Data.

7. Brownlee, J. (2018). *Deep Learning for Time Series Forecasting*. Machine Learning Mastery.

8. Brownlee, J. (2018). *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.

9. 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.