

AI-based Stock Price Prediction using Long Short-term Memory (LSTM)

MS. K. Suvitha 1, Parthiban S *2, Maari Muthmuthu K*3, Bharathi V*4, Manikandan N*5
*2,3,4,5 Students, Department of Computer Science and Engineering,
Dhanalakshmi Srinivasan Engineering College (Autonomous), Perambalur, India.

Ms.K.suvitha M.E., Assistant Professor,
Department of Computer Science and Engineering,
Dhanalakshmi Srinivasan Engineering College (Autonomous),
Perambalur , India.

ABSTRACT – Stock market prediction is a highly complex and challenging task due to the nonlinear, dynamic, and volatile nature of financial data. Traditional statistical and machine learning approaches often fail to capture long-term dependencies and hidden patterns present in stock price movements. To address these challenges, this work proposes an Artificial Intelligence-based stock price prediction system using Long Short-Term Memory (LSTM), a specialized deep learning model designed for time series forecasting. The proposed system utilizes historical stock market data including open price, close price, high price, low price, and trading volume to train the LSTM model. Data preprocessing techniques such as normalization and scaling are applied to improve prediction accuracy and model performance. The system is implemented using Python with TensorFlow and evaluated using metrics such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE). The results demonstrate that the LSTM model provides reliable and accurate predictions compared to traditional methods.

Keywords –

Stock Price Prediction, Artificial Intelligence, LSTM, Deep Learning, Time Series Forecasting, Neural Networks.

I. INTRODUCTION

1. Overview

The stock market is one of the most important components of the global financial system, allowing investors to trade shares and companies to raise capital. However, predicting stock prices is extremely difficult due to the influence of multiple unpredictable factors such as economic conditions, market trends, and investor behavior. The highly volatile and non-linear nature of stock price movements makes accurate forecasting a challenging problem.

Traditional approaches such as statistical analysis and technical indicators rely heavily on assumptions and historical trends. These methods are often limited in their ability to capture complex patterns and long-term dependencies in financial data. As a result, their prediction accuracy is often unsatisfactory in real-world scenarios.

2. Objective

The primary objective is to design and implement an LSTM-based stock price prediction system that leverages historical market data to forecast future price trends with high accuracy, providing a reliable decision-support tool for investors and financial analysts.

3. Problem Statement

With the growing complexity of financial markets and limitations of traditional forecasting models, there is a critical need for deep learning-based approaches that can capture temporal dependencies and nonlinear patterns in stock price data. Standard statistical methods and shallow machine learning models often fail to model such dynamics effectively.

4. Project Description

The proposed system comprises four primary components:

- **Data Collection Module** – Accepts historical OHLCV (Open, High, Low, Close, Volume) data from reliable financial

sources.

- **Preprocessing Module** — Handles missing values and applies Min-Max normalization to scale all features between 0 and 1.
- **LSTM Model** — A stacked LSTM network with dropout regularization that learns temporal dependencies in stock price sequences.
- **Prediction and Evaluation Module** — Evaluates model performance using MSE and RMSE and visualizes predicted versus actual prices.

5. Motivation

The motivation stems from the increasing demand for accurate and automated financial forecasting tools. Investors and analysts require reliable predictive models that go beyond traditional methods and can leverage the power of deep learning to identify patterns in sequential financial data, ultimately enabling more informed and data-driven investment decisions.

II. LITERATURE SURVEY

Stock price prediction has been widely studied in recent years due to its importance in financial decision-making. Traditional statistical methods such as linear regression and moving averages have been used for forecasting stock prices. However, these methods often fail to capture complex patterns and non-linear relationships present in financial data.

Machine learning techniques such as Support Vector Machines (SVM), Decision Trees, and Random Forest have been applied to stock prediction. These methods improve performance compared to traditional approaches but still struggle with time series data and long-term dependencies [6].

Deep learning approaches, particularly Recurrent Neural Networks (RNN), have shown promising results in time series prediction. However, standard RNNs suffer from vanishing gradient problems, which limit their ability to learn long-term dependencies. LSTM networks were introduced by Hochreiter and Schmidhuber [1] to overcome these limitations using memory cells and gating mechanisms.

Several studies have demonstrated that LSTM models outperform traditional and machine learning methods in stock price prediction tasks [3][4][5]. Despite these advancements, challenges such as data noise, market volatility, and overfitting still exist. Therefore, there is a continued need for efficient and robust models that can provide accurate predictions with minimal error.

Existing System

Existing forecasting tools such as ARIMA, linear regression, and basic RNNs provide time series prediction capabilities. However, these models are limited in their ability to model long-range dependencies and exhibit degraded performance on highly volatile financial data.

Disadvantages:

- ARIMA and linear regression fail to capture nonlinear patterns in stock data.
- Standard RNNs suffer from vanishing gradient problems, limiting long-term memory.
- Traditional models require manual feature engineering and domain-specific tuning.
- Poor generalization under sudden market fluctuations or anomalies.

III. PROPOSED SYSTEM

The proposed AI-Based Stock Price Prediction system is built on a stacked LSTM architecture integrated with a data preprocessing pipeline and performance evaluation module. The system processes historical OHLCV data by normalizing features and constructing sliding-window sequences, which are then fed into the LSTM model for training and prediction.

The LSTM model employs multiple stacked layers with dropout regularization to prevent overfitting. After training, the model generates predictions on unseen test data, and results are compared against actual stock prices using MSE and RMSE metrics. The system is implemented entirely in Python using NumPy, Pandas, and TensorFlow/Keras.

Key Components of the Proposed System:

- **DataIngestionLayer:** CSV-based historical market data ingested via Pandas with automated preprocessing.

- **PreprocessingPipeline:** Min-Max normalization applied across all features; sliding window of 60 time steps used to create sequences.
- **LSTMArchitecture:** Structured system prompts with code generation rules and output format specifications.
- **ModelConfiguration:** Dual LSTM layers (128 and 64 units) with Dropout(0.2) after each layer and a Dense output layer.
- **EvaluationModule:** MSE and RMSE computed on inverse-scaled predictions to measure real-world accuracy.

Advantages:

- Effectively captures both short-term fluctuations and long-term trends in stock prices.
- Dropout regularization prevents overfitting on historical training data.
- Scalable to any publicly traded stock with sufficient historical data.
- Low prediction error compared to traditional statistical and ML methods.
- End-to-end Python implementation with TensorFlow ensures reproducibility.

IV. MODULES

1. Data Collection and Preprocessing Module:

This module accepts historical stock data in CSV format containing OHLCV fields. Missing values are handled by forward-fill interpolation. Min-Max normalization is applied to scale all feature values between 0 and 1, ensuring stable gradient flow during training. The dataset is split into 80% training and 20% testing subsets. Sliding windows of 60 time steps are used to create input sequences for the LSTM.

2. LSTM Model Construction Module:

The LSTM model is constructed using the Keras Sequential API. It consists of two stacked LSTM layers with 128 and 64 units respectively, each followed by a Dropout layer with rate 0.2 to prevent overfitting. A Dense output layer with a single neuron generates the final price prediction. The model is compiled using the Adam optimizer with Mean Squared Error as the loss function.

3. Training and Optimization Module:

The model is trained over 50 epochs with a batch size of 32. Early stopping is applied with a patience of 5 epochs to prevent overfitting. The training process monitors validation loss and saves the best model checkpoint. Learning rate scheduling reduces the learning rate upon plateau to improve convergence.

4. Prediction and Evaluation Module:

The trained model generates predictions on the test dataset. Predictions are inverse-transformed from normalized scale to actual price values using the stored Min-Max scaler. Performance is evaluated using MSE and RMSE metrics. A visual comparison of predicted versus actual stock prices is plotted using Matplotlib to assess trend accuracy.

V. SYSTEM ARCHITECTURE

The system architecture follows a four-stage pipeline. At the foundation lies the Input Layer which handles CSV ingestion and format validation. The Preprocessing Layer applies normalization, sequence construction, and train/test splitting. The Model Layer comprises the stacked LSTM network with dropout regularization trained using the Adam optimizer. Finally, the Output Layer renders evaluation metrics and predicted versus actual price comparison plots.

Raw data is collected and passed through the preprocessing stages before being fed into the LSTM model. Feature extraction produces normalized time-step sequences. Training data is used to optimize model weights, while testing data evaluates generalization. The trained model then generates predictions that are inverse-scaled and compared against ground truth values.

VI. RESULTS AND DISCUSSION

The proposed LSTM-based stock price prediction model was evaluated using real-world historical stock datasets. The model demonstrates good prediction accuracy and effectively captures stock price trends across different market conditions.

TABLE I. MODEL PERFORMANCE COMPARISON

Me t ric	Training MSE	Test MSE	RMSE
LSTM Model	0.0012	0.0021	0.0458
Linear Regression	0.0089	0.0102	0.1010
SVM	0.0065	0.0078	0.0883
Random Forest	0.0041	0.0053	0.0728

The results show that the LSTM model achieves significantly lower MSE and RMSE compared to traditional methods including Linear Regression, SVM, and Random Forest. The predicted values closely match the actual stock prices, indicating the effectiveness of the temporal modeling approach.

The automated error recovery and dropout regularization resolved overfitting concerns while maintaining prediction stability. The system performs well under different market conditions, though minor deviations occur during periods of sudden market volatility. Multi-step ahead predictions show gradually increasing error, which is expected behavior for autoregressive forecasting models.

Overall, the results confirm that the LSTM model is suitable for stock price prediction and can serve as a reliable decision-support tool for investors and financial analysts.

VII. FUTURE ENHANCEMENTS

- Integration of sentiment analysis from financial news and social media to incorporate external market signals.
- Incorporation of attention mechanisms and Transformer-based architectures for improved long-ranged dependency modeling.
- Multi-stock portfolio analysis enabling simultaneous prediction across correlated equities.
- Real-time prediction pipeline with live market data feeds via financial APIs (Yahoo Finance, Alpha Vantage).
- Hyperparameter optimization using Bayesian search to automatically tune LSTM layer sizes and dropout rates.
- Deployment as a web application with interactive dashboards for non-technical investors.

VIII. CONCLUSION

This work presents an AI-based stock price prediction system using Long Short-Term Memory networks. The model effectively captures temporal patterns in stock market data and provides accurate predictions across diverse market conditions. Compared to traditional methods, the LSTM model offers better performance and handles time series data efficiently.

The stacked LSTM architecture with dropout regularization demonstrates that deep learning models can be applied successfully to financial forecasting with minimal preprocessing overhead. The system can assist investors in making informed decisions by analyzing historical trends and generating reliable future price estimates.

As deep learning and AI continue to advance, systems like the proposed LSTM predictor will become increasingly pivotal in democratizing data-driven financial analysis, ultimately contributing to more informed and accessible investment decision-making.

IX. REFERENCES

- [1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.
- [2] A. Graves, "Long Short-Term Memory," in *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer, 2012, pp. 37-45.
- [3] Y. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654-669, 2018.
- [4] M. Nelson, A. Pereira and R. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 1419-1426.
- [5] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in *2017 ICACCI*, pp. 1643-1647.
- [6] J. Patel, S. Shah, P. Thakkar and K. Kotecha, "Predicting stock and stock price index movement using machine learning techniques," *Expert Systems with Applications*, vol. 42, no. 1, pp. 259-268, 2015.

- [7] K. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, pp. 307-319, 2003.
- [8] X. Ding, Y. Zhang, T. Liu and J. Duan, "Deep learning for event-driven stock prediction," in *IJCAI*, 2015, pp. 2327-2333.
- [9] Q. Lin, Y. Chen, S. Chen and S. Liu, "Deep learning for stock market prediction: A survey," *IEEE Access*, vol. 9, pp. 123-140, 2021.
- [10] Z. Hu, Y. Zhao and M. Khushi, "A survey of forex and stock price prediction using deep learning," *Applied System Innovation*, vol. 4, no. 1, pp. 1-19, 2021.
- [11] B. Rout, S. K. Dash and S. Dash, "A hybrid model for stock market prediction using machine learning techniques," *Procedia Computer Science*, vol. 132, pp. 1351-1362, 2018.
- [12] H. Mo, "Application of LSTM neural network in stock prediction," *Journal of Physics: Conference Series*, vol. 1237, 2019.
- [13] P. K. Babu and N. R. Mohan, "Stock price prediction using deep learning models," *International Journal of Engineering and Advanced Technology*, vol. 8, no. 6, pp. 234-239, 2019.
- [14] J. Brownlee, "Introduction to Time Series Forecasting with Python," *Machine Learning Mastery*, 2017.
- [15] F. Chollet, *Deep Learning with Python*, Manning Publications, 2018.
- [16] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016.
- [17] S. Siarni-Namini, N. Tavakoli and A. S. Namin, "A comparison of ARIMA and LSTM in forecasting time series," in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1394-1401.
- [18] R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, "Deep learning for stock prediction using numerical and textual information," in 2016 IEEE/ACIS ICIS, pp. 1-6.
- [19] H. Bao, X. Yue and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and LSTM," *PLOS ONE*, vol. 12, no. 7, 2017.
- [20] T. Fischer, C. Krauss and D. Schlag, "Applying deep learning to stock market prediction," *Working Paper*, 2018.