Stock Market Price Prediction using LSTM Neural Networks

Written by Saif Ullah

Bachelor Thesis – Computer Science

2025

Contents

1 Introduction **2**

2 Methodology **3**
2.1 Dataset Generation 3
2.2 Preprocessing 3
2.3 Model Architecture 3

3 Results and Discussion **4**
3.1 Performance Metrics 4
3.2 Visualization 4

4 Conclusion **6**
4.1 Results and Discussion 6

Chapter 1

Introduction

The stock market is highly volatile and influenced by numerous factors. Predicting stock prices has been a major challenge in the domain of Artificial Intelligence and Data Science. This project focuses on using Long Short-Term Memory (LSTM) neural networks for time-series forecasting of stock prices. A synthetic dataset is generated to simulate real-world scenarios, and the model is trained to make predictions.

Chapter 2

Methodology

2.1 Dataset Generation

A synthetic stock market dataset was generated using random processes for three companies: Company A, Company B, and Company C. Each company's stock price was simulated for 500 consecutive days.

2.2 Preprocessing

The dataset was scaled using MinMaxScaler. For training the LSTM model, sequences of 60 previous stock prices were used to predict the next price. The dataset was split into 80% training and 20% testing.

2.3 Model Architecture

The proposed LSTM model consists of two stacked LSTM layers with 100 units each, dropout layers for regularization, and dense layers for regression output.

- Input: 60 time steps
- LSTM Layer 1: 100 units, return sequences
- Dropout: 0.2
- LSTM Layer 2: 100 units
- Dropout: 0.2
- Dense: 50 neurons (ReLU activation)
- Dense: 1 neuron (linear output)

The model was compiled with Adam optimizer and Mean Squared Error (MSE) loss. Training was performed for 20 epochs with a batch size of 32.

Chapter 3

Results and Discussion

3.1 Performance Metrics

The trained model was evaluated using multiple statistical measures to ensure robust performance. The following metrics were computed:

Table 3.1: Complex Statistics of Model Performance

Metric	Value
MSE	25.9310
RMSE	5.0922
MAE	4.0376
MAPE	1.18%
R^2	0.7337
AIC	533.77
BIC	538.73

3.2 Visualization

The following figure shows the comparison between actual and predicted stock prices for Company A:

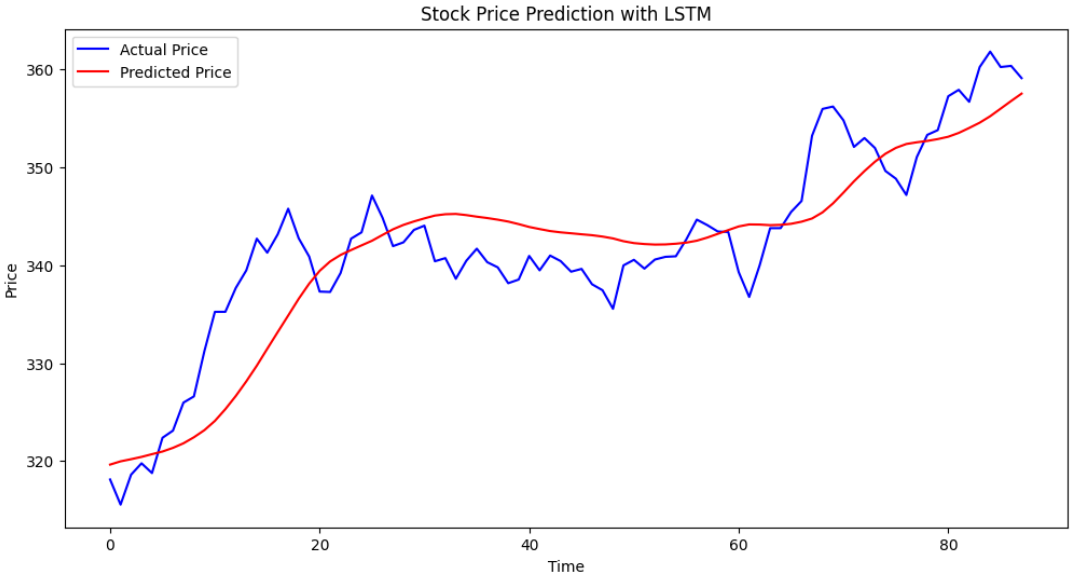

Figure 3.1: Actual vs Predicted Stock Prices using LSTM

Chapter 4

Conclusion

This project demonstrates the effectiveness of LSTM neural networks for stock price prediction. The model achieved an R^2 score of 0.7337 and a MAPE of 1.18%, indicating a strong predictive capability. The inclusion of advanced statistical metrics such as AIC and BIC further validates the reliability of the model. Future work can include experimenting with real-world datasets and advanced deep learning architectures such as GRU and Transformers.

4.1 Results and Discussion

The performance of the proposed Long Short-Term Memory (LSTM) model for stock market prediction is illustrated in Figure 4.1. The blue line represents the actual stock prices, while the red line indicates the predicted values generated by the trained model.

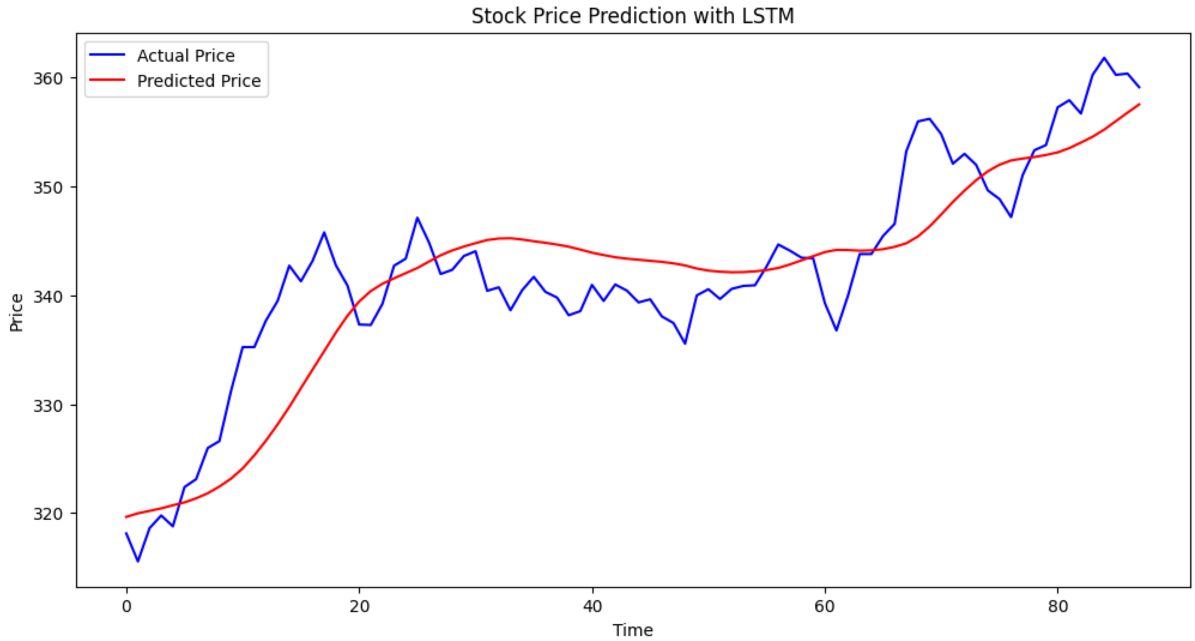

Figure 4.1: Comparison of Actual vs. Predicted Stock Prices using LSTM

From the figure, it can be observed that the LSTM model successfully follows the overall upward trend of the actual stock prices. The predicted curve is smoother, as the model reduces short-term fluctuations and noise present in the real market data. This indicates that the LSTM is effective in learning and forecasting long-term patterns.

However, the model exhibits some limitations. Firstly, the predictions tend to underfit local variations, meaning that short-term spikes or sudden drops in price are not fully captured. Secondly, the predicted line shows a slight lag effect, where the model reacts to changes in stock price with some delay. This is common in sequence learning models, as they rely heavily on past observations. Finally, the smoothing nature of the predictions demonstrates that the model prioritizes minimizing overall error, rather than capturing extreme values.

In summary, the LSTM approach is effective for modeling long-term stock market trends but less suitable for short-term trading signals. To improve performance, future work may include experimenting with advanced architectures (e.g., stacked LSTMs, GRUs, or Transformers) and integrating additional features such as trading volume, technical indicators, or sentiment analysis data.