# ML Project Report


Dhruv Maheshwari - PES2UG23CS173 | Sem 5 – C | Date: 12th Oct 2025

Divyansh Sharma - PES2UG23CS181 | Team: EC_C_173_181


***Project Title: Intraday Stock Price Forecasting with Bidirectional LSTM***

**Problem Statement**

The main problem addressed is intraday stock price forecasting for a given ticker (GOOG in the example notebook) using high-frequency (5-minute interval) data. The goal is to predict short-term price movements to support potential trading decisions, which is challenging due to the inherent volatility and non-linear nature of financial time series.

**Objective**

The project aims to develop an optimized and robust Bidirectional Long Short-Term Memory (Bi-LSTM) neural network model capable of multi-step prediction to forecast the Close price for the next 5 five-minute intervals based on the stock's recent historical data.

**Dataset Details**

| Detail | Value |
|---|---|
| Source | yfinance library (Yahoo) |
| Size | 1673 samples (1 month - 5min intervals ) |
| Key Features | Open, High, Low, Close, Volume |
| Target Variable | Future Close Price for next 5 intervals |

**Architecture Diagram**

The architecture is a Deep Bidirectional LSTM network.

Layer Structure

- Input Layer: Expects a sequence of 90 time steps and 5 features.
- Bidirectional LSTM (150 units): Allows the network to process the sequence both forward and backward for better context capture. Uses tanh activation and L2 regularization.
- Dropout (0.3): Applied for regularization to prevent overfitting.
- Bidirectional LSTM (100 units): Second Bi-LSTM layer, also with tanh activation and L2 regularization.
- LSTM (50 units): Standard LSTM layer, returning only the final sequence output.
- Dense Layer (10 units): Intermediate dense layer with relu activation.

- Output Layer (5 units): A Dense layer predicting the 5-step future Close prices.

---

**Methodology**

The workflow employed a robust, step-by-step process:

- Data Acquisition: GOOG (Alphabet Inc.) stock price data was fetched for 1 month at 5-minute intervals using the yfinance library.
- Feature Selection: The 'Open, High, Low, Close, Volume' features were selected.
- Scaling: The data was scaled using RobustScaler to minimize the impact of potential outliers, common in financial data.
- Sequence Creation & Splitting: Data was transformed into sequences with a length of 90 time steps to predict a horizon of 5 future steps (multi-step prediction) . The dataset was split into 80% training and 20% testing sets.
- Model Configuration: A deep Bi-LSTM model was compiled using the AdamW optimizer (with a learning rate of 0.0005 ) and Mean Squared Error (MSE) as the loss function. L2 regularization was applied to the LSTM layers.
- Training: The model was trained for up to 100 epochs with a small batch size of 32. EarlyStopping and ReduceLROnPlateau callbacks were used to optimize convergence and prevent overfitting.

---

**Results & Evaluation**

The model was evaluated using the 1-Step Ahead prediction (the first value in the 5-step output) against the corresponding actual value.

| Metric | Value |
|---|---|
| Mean Squared Error (MSE) | 0.3829 |
| Root Mean Squared Error (RMSE) | 0.6188 |
| Mean Absolute Error (MAE) | 0.4336 |
| $R^2$ Score | 0.9741 |

The $R^2$ Score of 0.9741 suggests the model's predictions closely track the actual price movements in the test set. An RMSE of 0.6188 indicates that the average prediction error, in unscaled USD, is approximately 62 cents for the 1-step ahead forecast.

---

**Conclusion**

The deep Bidirectional LSTM model successfully learned the sequential patterns in the intraday stock data. The high $R^2$ score demonstrates strong predictive power for short-term stock price movements. The use of RobustScaler and Bidirectional layers appears to have contributed to the model's performance by making it less sensitive to outliers and better at capturing contextual dependencies in the time series. A key learning point was that the multi-step block prediction approach efficiently generates longer-term forecasts (e.g., the next 30 intervals) by reducing the number of intensive model calls.

---

## Example Screenshot