Stock Market Prediction Using LSTM & Prophet

1 Data Preprocessing

1.1 Data Collection

- **Source:** Yahoo Finance (`yfinance` library).
- **Timeframe:** Extracted historical stock data.
- **Columns Used:** Date, Open, High, Low, Close, Volume.

1.2 Handling Outliers Using IQR

- **Method:** Interquartile Range (IQR) method was applied to detect and remove outliers.
- **Formula Used:**

$$\begin{aligned} Q1 &= \text{25th percentile} \\ Q3 &= \text{75th percentile} \\ \text{IQR} &= Q3 - Q1 \\ \text{Lower Bound} &= Q1 - (1.5 \times \text{IQR}) \\ \text{Upper Bound} &= Q3 + (1.5 \times \text{IQR}) \end{aligned}$$

- **Action:** Values outside these bounds were removed to prevent extreme fluctuations.

1.3 Filling Missing Values

- **Method:** Median imputation was used to fill missing values for numerical columns.

1.4 Feature Selection Using Correlation Matrix

- **Threshold Applied:** Features with an absolute correlation greater than 0.6 with the target (Close Price) were selected.
- **Process:**

1. Compute the correlation matrix.
2. Extract features with $|\text{correlation}| > 0.6$.
3. Generate a heatmap for visualization.

- **Selected Features:**
  - SMA_20 (Simple Moving Average)
  - MACD_Signal
  - Bollinger_High & Bollinger_Low
  - ATR_14 (Average True Range)
  - Cumulative Returns

1.5 Normalization

- **Scaler Used:** `MinMaxScaler()` from `sklearn.preprocessing`.
- **Reason:** LSTMs perform better with scaled inputs between 0 and 1.

2 Feature Engineering & Technical Indicators

- **SMA_20 (Simple Moving Average, 20-day):** Identifies short-term trends.
- **MACD & MACD Signal:** Measures momentum changes and trend strength.
- **Bollinger Bands (Upper & Lower):** Indicates volatility by plotting bands around moving averages.
- **RSI_14 (Relative Strength Index, 14-day):** Identifies overbought or oversold conditions.
- **ATR_14 (Average True Range, 14-day):** Measures market volatility.
- **Cumulative Returns:** Tracks total percentage change in stock price over time.
- **Rolling Volatility (20-day):** Identifies fluctuations in price movements.
- **Bollinger %B:** Normalizes stock price position relative to Bollinger Bands.

3 Model Selection & Hyperparameter Tuning

3.1 LSTM (Long Short-Term Memory) Model

- **Architecture:**
  - **Input Layer:** Sequences of selected stock features.
  - **Two LSTM Layers:** Each with 30 units.
  - **Dropout Layers:** Dropout rate of 0.2 to prevent overfitting.
  - **Dense Layer:** Fully connected output layer.
- **Loss Function:** Mean Squared Error (MSE).
- **Optimizer:** Adam optimizer.
- **Hyperparameters:**
  - **Epochs:** 50
  - **Batch Size:** 32
  - **Validation Split:** 20% of training data.

3.2 Facebook Prophet

- **Why Used?** Handles time-series decomposition and seasonality trends.
- **Forecasting Process:**
  - **Inputs:** `ds` (date) and `y` (closing price).
  - **Trends:** Prophet automatically detects trends and seasonal patterns.
  - **Holidays:** No custom holiday effects were added in this case.

4 Model Evaluation & Insights

4.1 Evaluation Metrics

Metric	Value
Mean Absolute Error (MAE)	4.0004
Mean Squared Error (MSE)	23.5446
Root Mean Squared Error (RMSE)	4.8523
R^2 Score	0.9508
Akaike Information Criterion (AIC)	617.9845
Bayesian Information Criterion (BIC)	621.2575

Table 1: LSTM Model Evaluation Metrics

4.2 Key Insights

- LSTM performed well in capturing stock price trends with a high R^2 score.
- Prophet was better suited for long-term forecasting but was less accurate for short-term movements.
- A hybrid approach (combining LSTM & Prophet) might improve accuracy by leveraging both short-term and long-term trend detection.