Stock Market Forecasting

Leveraging ML & LSTM for Next-Day Price Prediction

Presented by Munnangi Sonu Sarojini

AP22110010609

Subject:- Information Retrievel

Presentation Agenda

01

Technical Skills Deep Dive

A look into the programming languages, ML concepts, and tools utilized.

02

Project Overview & Data Pipeline

Understanding the project's scope, objectives, and data flow.

03

Modeling Approach

From baseline models to advanced Deep Learning with LSTM.

04

Real-time Dashboard & Results

Showcasing the interactive prediction application and model performance.

05

Challenges & Key Learnings

Insights gained and future considerations for stock forecasting.

Technical Foundations: ML & Data Science

Machine Learning & Deep Learning

- Regression Models: Linear Regression, Random Forest
- ARIMA Time-Series Forecasting
- Deep Learning: LSTM using PyTorch
- Feature Engineering: SMA, EMA, RSI, MACD, Lags

Data Handling & Analysis

- Data Cleaning & Preprocessing
- Feature Scaling for model optimization
- Missing Value Treatment strategies
- Exploratory Data Analysis (EDA) for insights

Technical Foundations: Tools & Deployment

Git & GitHub

Version control and collaborative development.

VS Code

Integrated Development Environment (IDE).

Streamlit

Rapid app deployment for interactive dashboards.

Project Overview: Stock Market Forecasting

This project aims to predict next-day stock prices by comparing various modeling techniques, ultimately deploying a real-time prediction dashboard.

1

Goal

To accurately forecast next-day stock prices using a tiered approach to modeling.

2

Modeling Hierarchy

Progressing from simple Baseline Models to sophisticated Machine Learning and Deep Learning (LSTM) approaches.

3

Real-time Deployment

Developing an interactive dashboard using Streamlit to visualize predictions.

4

Stocks Analyzed

Focus on major tech stocks (AAPL, MSFT, TSLA) and the broader market index (^GSPC).

End-to-End Data Pipeline

Fetch Raw Data

Automated retrieval from Yahoo Finance.

Clean & Format

Preprocessing for consistency and usability.

Feature Engineering

Creation of relevant technical indicators.

Train Models

Applying various algorithms for prediction.

Evaluate & Plot

Assessing model performance and visualizing results.

Build Real-time Dashboard

Deployment for interactive forecasting.

Feature Engineering: Enriching the Data

Key technical indicators were generated to capture market dynamics and enhance predictive power, transforming raw data into meaningful features.

Technical Indicators Utilized

- **SMA** (Simple Moving Average)
- **EMA** (Exponential Moving Average)
- **RSI** (Relative Strength Index)
- **MACD** (Moving Average Convergence Divergence) & its components
- **Bollinger Bands** (Upper, Mid, Lower)
- Daily Returns & Log Returns
- Volatility Measures
- Price Lags (1-10 days)

Code Snippet Example

```
df["SMA_20"] =
df["Close"].rolling(20).mean()
df["EMA_20"] =
df["Close"].ewm(span=20).me
an()
df["RSI_14"] = RSI(df["Close"],
14)
df["MACD"],
df["MACD_signal"],
df["MACD_hist"] =
MACD(df["Close"])
```

These features provide critical context for our predictive models.

Modeling Progression: Baseline to Deep Learning

Baseline Models

- **Naive Model:** Tomorrow's price equals today's.
- **Rolling Mean:** Average of the last N days.
- **Drift Model:** Incorporates a long-term trend.

Crucial for establishing a performance benchmark.

Machine Learning Models

- **Linear Regression:** Simple, fast, and interpretable.
- **Random Forest Regression:** Ensemble method, robust to overfitting.

Provide more sophisticated predictive power.

LSTM Deep Learning Model

- **Optimized for Time-Series:** Effectively captures temporal dependencies.
- **Handles Noisy Data:** Robust performance on volatile stock data.
- **Architecture:** Two LSTM layers (64 hidden units) and one Dense output layer.

Designed to capture complex patterns in sequential data.

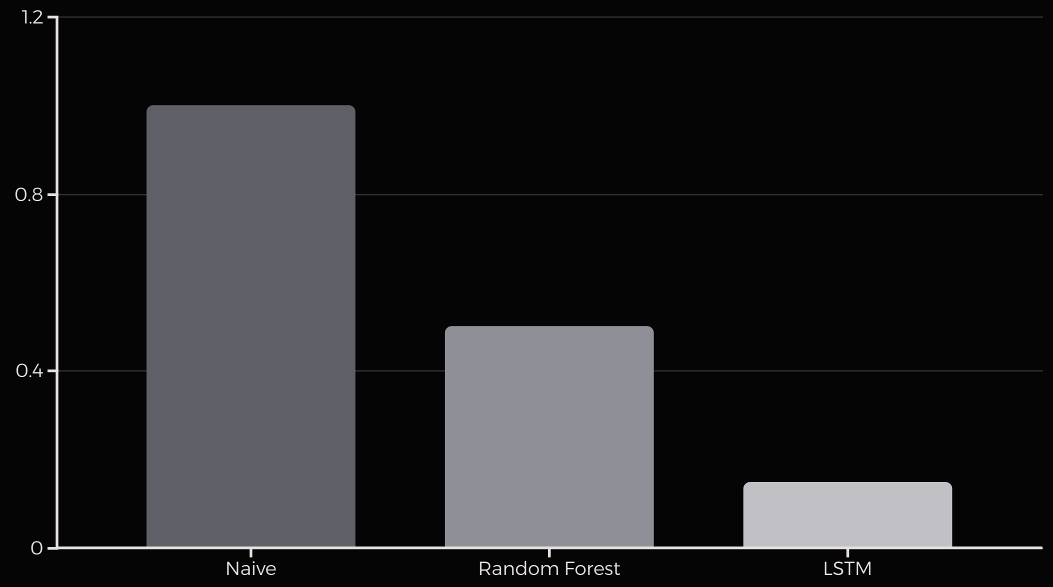
Real-time Dashboard & Model Performance

Streamlit Real-time App

An interactive application demonstrating live stock price predictions:

- Fetches current market data automatically.
- Recomputes features in real-time.
- Loads the pre-trained LSTM model.
- Predicts next-day closing prices.
- Visualizes Actual vs. Predicted values.

Model Comparison (Sample RMSE)



Conclusion: The LSTM model consistently outperforms baseline and traditional ML models, achieving the lowest RMSE, MAE, and MAPE, indicating superior predictive accuracy.