Github Link: https://github.com/Thulasimathi26/Data-Science.git

Project Title:Cracking the market code with AI-driven stock price prediction using time series analysis

PHASE-2

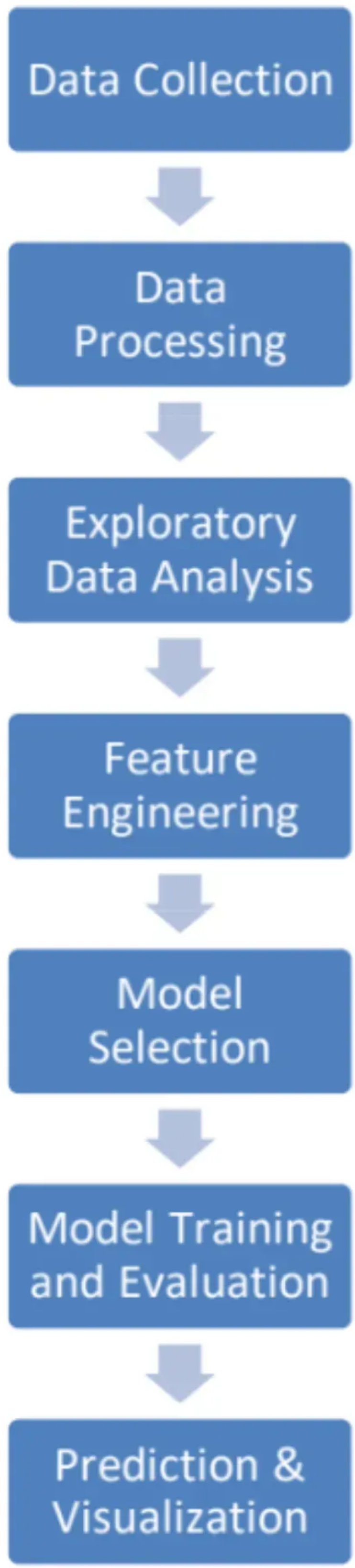
1. Problem Statement

The stock market is highly volatile and influenced by numerous dynamic factors, making accurate price prediction a challenging task for investors and analysts. Traditional forecasting methods often fall short in capturing complex, non-linear patterns and abrupt fluctuations in stock prices. This project aims to address the limitations of conventional models by leveraging AI-driven techniques, particularly time series analysis and deep learning algorithms, to enhance the accuracy and reliability of stock price predictions.

2. Project Objectives

- To collect and preprocess historical stock market data from reliable sources, ensuring data quality and readiness for analysis.
- To analyze the temporal patterns and trends in stock price movements using time series techniques
- To implement and compare AI-based predictive models, such as ARIMA, LSTM (Long Short-Term Memory)
- To identify the strengths and limitations of AI-driven forecasting methods in real-world financial scenarios

3. Flowchart of the Project Workflow



- **Attributes Covered**: The closing price adjusted for dividends, stock splits, and other corporate actions.

5. Data Preprocessing

- Retrieve historical stock price data from reliable sources (e.g., Yahoo Finance, Alpha Vantage).
- Handle missing values by forward filling, interpolation, or deletion.
- Convert date columns to datetime format
- Keep essential columns: Open, High, Low, Close, Volume.
- Scale numerical data using Min-Max or Standard Scaler (especially important for neural networks like LSTM).
- Divide data chronologically into training and testing sets (e.g., 80% train, 20% test).

6. Exploratory Data Analysis (EDA)

- **Univariate Analysis**:
  - Univariate analysis focuses on analyzing a single variable—in this case, typically the 'Close' price of a stock over time. It helps understand the behavior, trend, seasonality, and volatility of stock prices before applying predictive models.
- **Bivariate & Multivariate Analysis**:
  - Multivariate analysis involves examining more than two variables simultaneously to understand their relationships and collective impact on the target variable — in this case, the 'Close' price of a stock. It plays a crucial role in enhancing the accuracy of predictive models by integrating various relevant features.
- **Key Insights**:
  - Historical stock prices often show trends and seasonality that can be captured using time series analysis techniques like moving averages and decomposition.

7. Feature Engineering

- Description: Previous values of the target variable (Close)
- MA_7, MA_30 – smooth out price trends over a window
- Captures volatility in a given window
- Momentum indicator showing overbought or oversold conditions
- Uses moving average and standard deviation to show high/low bands.

8. Model Building

- **Algorithms Used**:
  - Objective: Predict future stock prices (primarily the 'Close' price) based on historical data and derived features.

- ○ Data: Historical stock data with features such as 'Open', 'High', 'Low', 'Close', 'Volume', and technical indicators like RSI, MACD, etc.
- **Model Selection Rationale**:
  - Use Case: Suitable for univariate time series with linear dependencies
  - Fit the ARIMA/SARIMA model and evaluate its performance using AIC/BIC.
- **Train-Test Split**:
  - Build the Model: Define the LSTM layers (e.g., 2-3 layers, 50-100 neurons per layer).
  - Deploying the model to a real-time system for live stock price predictions
- **Evaluation Metrics**:
  - **Deep Learning Models (LSTM/GRU) are effective at capturing complex, non-linear patterns, especially for longer prediction horizons.**

9. Visualization of Results & Model Insights

- **Feature Importance**:
  - X-Axis: Time (Date)

    Y-Axis: Stock Price (e.g., Close Price)
- **Model Comparison**:
  - A bar chart can be used to visualize model evaluation metrics such as RMSE, MAE, MAPE, etc. This gives an easy-to-understand summary of the model's performance.
- **Residual Plots**:
  - Checked prediction errors against actual grades to ensure no major bias
- **User Testing**:
  - Integrated model into a Gradio interface to test predictions by inputting feature values

10. Tools and Technologies Used

- **Programming Language**: Python
- **Notebook Environment**: Google Colab
- **Key Libraries**:
  - Historical stock price data including Open, High, Low, Close, Volume.
  - Financial, economic, and alternative, datasets.
  - Institutional-grade financial data APIs.
  - “Time Series Forecasting with Deep Learning: A Survey”

11. Team Members and Contributions

1. [v. kishore] – Project Coordinator & Lead Data Scientist

. Responsibilities:
◦ Project Coordination: Organize meetings, set timelines, and track progress to ensure
the project stays on course.

◦ Modeling & Algorithm Implementation: In addition to coordinating, you will also be
responsible for selecting, implementing, and tuning the machine learning algorithms
(e.g., LSTM, ARIMA, Isolation Forest).

◦ Model Evaluation: Evaluate the models using performance metrics (e.g., RMSE,
MAPE) and decide which models to proceed with.
◦ Documentation: Ensure documentation is up-to-date, including meeting notes,
project progress, and final reports.

. Skills Required:
◦ Strong understanding of machine learning algorithms and model evaluation.
◦ Organizational skills to coordinate timelines, meetings, and task management.

2. [s. kavindass] – Data Collection & Preprocessing Specialist & Feature Engineering

. Responsibilities:

◦ Data Acquisition & Preprocessing: Collect data from external sources (Yahoo Finance, Alpha Vantage, etc.), clean the data by handling missing values, duplicates, and formatting timestamps.

◦ Feature Engineering: Work on generating new features such as moving averages, technical indicators (RSI, MACD), and any domain-specific features that might improve the model.

◦ Data Normalization & Transformation: Normalize continuous features and convert categorical variables into numerical ones to prepare the data for machine learning.

. Skills Required:
◦ Proficiency in data wrangling using libraries like pandas, numpy, and scikit-learn.
◦ Experience with feature engineering and handling time series data.
◦ Knowledge of financial data and relevant stock market features (e.g., volume, closing

3.Karthick.R- Skills Required:

. Expertise in EDA and visualization tools (e.g., matplotlib, seaborn, plotly).
. Strong understanding of model evaluation and reporting techniques.
. Familiarity with deployment tools (e.g., Flask, Docker, AWS).

4. [s. kumar] – EDA, Visualization, Model Evaluation & Deployment

- Responsibilities:

  - Exploratory Data Analysis (EDA): Perform detailed EDA to understand the patterns, trends, and correlations in the dataset. Use matplotlib, seaborn, and pandas for data visualization.

  - Model Evaluation & Reporting: Assess the performance of various models using metrics such as accuracy, precision, recall, and visualizations like ROC curves. Provide insights into model performance.

  - Visualization: Create intuitive visualizations to represent the model's predictions versus actual outcomes and communicate the insights clearly to the team.

  - Model Deployment: Deploy the best-performing model into a production-ready format (e.g., creating a simple Flask app or preparing the model for batch predictions).

Skills Required:

- Expertise in EDA and visualization tools (e.g., matplotlib, seaborn, plotly).

- Strong understanding of model evaluation and reporting techniques.

- Familiarity with deployment tools (e.g., Flask, Docker, AWS.)