Cracking the Market Code with AI-Driven Stock Price Prediction using Time Series Analysis

PHASE -2

Student Name. : S.AKASH

Register Number: 620523243006

Institution : CMS COLLEGE OF ENGINEERING

Department : ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Date of Submission:

Problem Statement:

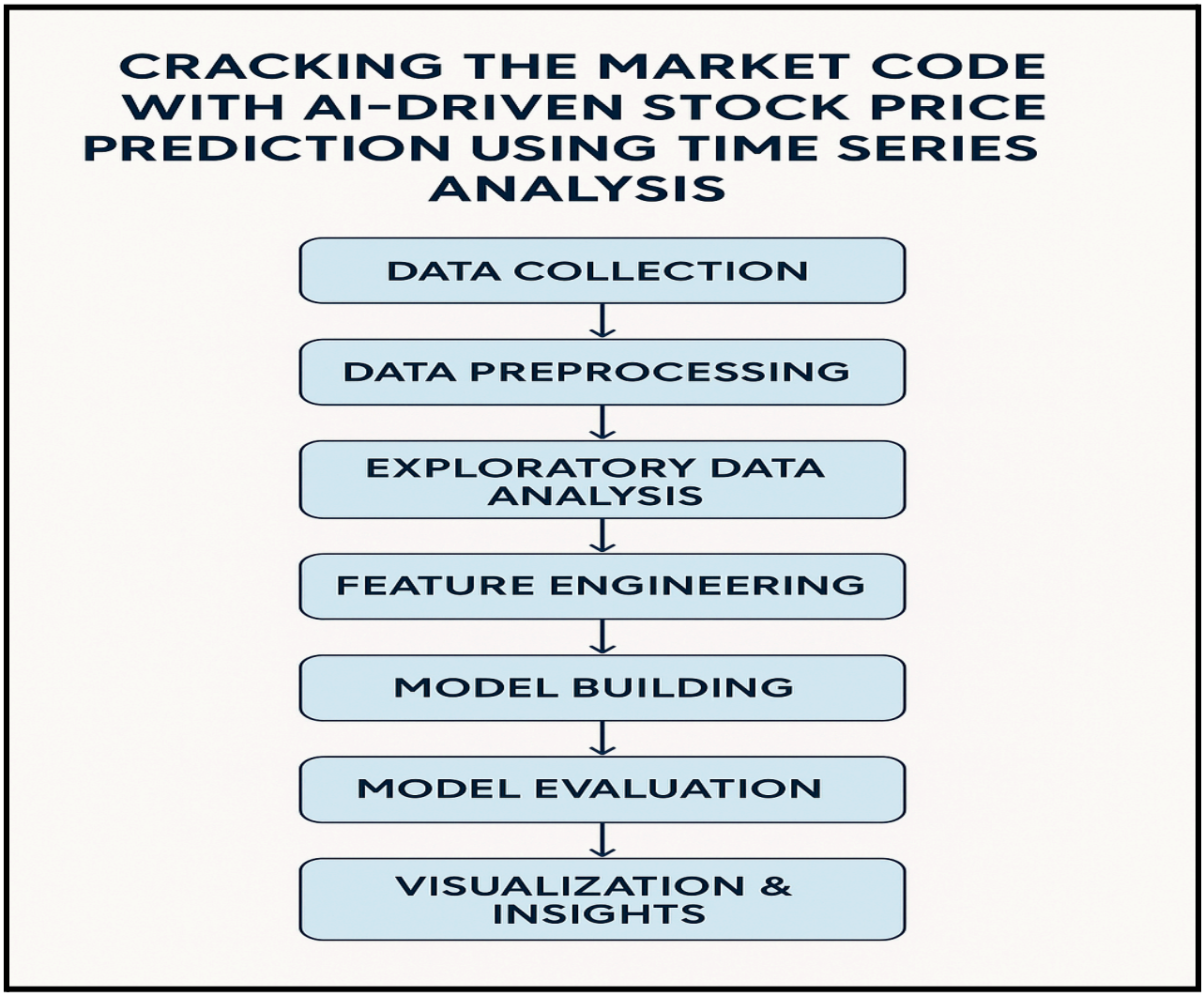
Predicting stock prices accurately is a complex and challenging task due to the inherent volatility and non-linearity of financial markets. Traditional methods often struggle to capture the intricate patterns and dependencies within time series data, leading to limited predictive power. This unpredictability poses significant risks for investors and financial institutions. There is a growing need for more sophisticated and data-driven approaches that can leverage the power of Artificial Intelligence (AI) and time series analysis techniques to improve the accuracy and reliability of stock price forecasts.

Project Objectives:

- ★ **Data Acquisition and Preparation:** To collect historical stock price data and relevant macroeconomic indicators from reliable sources and perform necessary data cleaning and preprocessing.
- ★ **Exploratory Data Analysis (EDA):** To conduct a thorough analysis of the data to understand its statistical properties, identify trends, seasonality, and potential correlations between different features.
- ★ **Feature Engineering:** To create relevant features from the raw data and external sources that can potentially improve the predictive power of the models. This includes lagged variables, technical indicators, and potentially sentiment analysis scores.
- ★ **Model Building and Selection:** To develop and evaluate various time series forecasting models, including traditional statistical models (e.g., ARIMA, Exponential Smoothing) and advanced AI-driven models (e.g., Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, Transformer networks).

- ★ **Model Training and Optimization:** To train the selected models using the preprocessed data and optimize their hyperparameters to achieve the best possible performance.
- ★ **Performance Evaluation:** To rigorously evaluate the performance of the trained models using appropriate evaluation metrics (e.g., Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), R-squared, Directional Accuracy) on unseen test data.
- ★ **Visualization of Results and Model Insights:** To visualize the predicted stock prices against the actual prices and to provide insights into the factors influencing the model's predictions. This may involve feature importance analysis for certain models.
- ★ **Development of a Prototype System:** To potentially develop a basic prototype system or dashboard to demonstrate the model's predictions.

Flow Chart of the Workflow:



Data Description:

- ★ **Time Series Data:**
  - Date
  - Open Price
  - High Price
  - Low Price
  - Close Price
  - Adjusted Close Price (accounting for dividends and stock splits)
  - Volume
- ★ **Potential External Data Sources:**
  - **Economic Indicators:** GDP growth rate, inflation rate, interest rates, unemployment rate, etc.
  - **Company-Specific News and Announcements:** Earnings reports, new product launches, mergers and acquisitions, etc. (potentially used for sentiment analysis).
  - **Market Sentiment Data:** Indices like VIX, news sentiment scores.
  - **Technical Indicators:** Moving averages, RSI, MACD, etc. (will be generated through feature engineering).

Data Preprocessing:

- ★ **Data Cleaning:** Handling missing values (imputation or removal), identifying and treating outliers.
- ★ **Data Transformation:** Scaling or normalization of the data to ensure that different features have comparable ranges, which can improve model performance (e.g., Min-Max scaling, Standardization).
- ★ **Time Series Specific Preprocessing:**
  - Checking for stationarity of the time series and applying transformations if necessary (e.g., differencing).
  - Resampling the data to different frequencies (e.g., daily, weekly, monthly) if required.
  - Splitting the data into training, validation, and testing sets to avoid overfitting and evaluate the model's generalization ability.

Exploratory Data Analysis (EDA):

- ★ **Visualizations:**
  - Time series plots of stock prices and volume.
  - Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots to identify the order of ARIMA models.
  - Histograms and distribution plots to understand the data distribution.

- Scatter plots to explore correlations between different features.
- Box plots to identify outliers.
- Decomposition of time series into trend, seasonality, and residuals.
-

★ **Statistical Analysis:**
- Descriptive statistics (mean, median, standard deviation, etc.).
- Tests for stationarity (e.g., Augmented Dickey-Fuller test).
- Correlation analysis.
- Seasonality analysis.

Feature Engineering:

★ **Lagged Variables:** Including past values of the target variable (stock price) and other relevant features as predictors. The number of lags will be determined based on the ACF and PACF plots and experimentation.
★ **Technical Indicators:** Calculating and incorporating popular technical indicators used in financial analysis, such as:
- Moving Averages (Simple Moving Average, Exponential Moving Average)
- Relative Strength Index (RSI)
- Moving Average Convergence Divergence (MACD)
- Bollinger Bands
- Stochastic Oscillator

★ **Volatility Measures:** Calculating historical volatility based on price fluctuations.
★ **Date and Time Features:** Extracting temporal features like day of the week, month, quarter, year, which might capture seasonal patterns.
★ **External Data Integration:** Incorporating relevant macroeconomic indicators or sentiment scores (if available).

Model Building:

★ **Statistical Time Series Models:**
- **Autoregressive Integrated Moving Average (ARIMA):** Identifying the optimal (p, d, q) parameters based on ACF and PACF plots and using techniques like AIC or BIC for model selection.
- **Exponential Smoothing (ETS):** Applying different variations like Simple Exponential Smoothing, Holt's Linear Trend, and Holt-Winters' Seasonal Method based on the data's characteristics.
- **Seasonal ARIMA (SARIMA):** Extending ARIMA to account for seasonality in the data.

★ **AI-Driven Models:**
- **Recurrent Neural Networks (RNNs):** Utilizing basic RNN architectures to capture sequential dependencies in the time series data.
- **Long Short-Term Memory (LSTM) Networks:** Employing LSTM networks, which are well-suited for capturing long-range dependencies and mitigating the vanishing gradient problem in RNNs.
- **Gated Recurrent Units (GRUs):** Another type of RNN that is often more

computationally efficient than LSTMs.
    - **Transformer Networks:** Exploring Transformer architectures, which have shown remarkable success in sequence modeling tasks, including time series forecasting.

Visualization of Results & Model Insights:

- ★ **Predicted vs. Actual Price Plots:** Visualizing the predicted stock prices against the actual prices over the test period to assess the model's accuracy.
- ★ **Error Plots:** Plotting the residuals (the difference between predicted and actual values) to check for any systematic biases or patterns.
- ★ **Performance Metrics Summary:** Presenting the calculated evaluation metrics (MSE, RMSE, MAE, R-squared, Directional Accuracy) in a clear and concise manner.
- ★ **Feature Importance Analysis (for relevant models):** If using models like tree-based methods or attention mechanisms in neural networks, we will attempt to visualize the importance of different features in the model's predictions.
- ★ **Scenario Analysis (optional):** Exploring how the model's predictions change under different input scenarios.

Tools and Technologies Used:

- ★ **Programming Language:** Python
- ★ **Data Analysis and Manipulation Libraries:** Pandas, NumPy
- ★ **Data Visualization Libraries:** Matplotlib, Seaborn, Plotly
- ★ **Time Series Analysis Libraries:** Statsmodels
- ★ **Machine Learning and Deep Learning Libraries:** Scikit-learn, TensorFlow, Keras, PyTorch
- ★ **Data Acquisition:** Libraries like yfinance, requests (for API access)
- ★ **Development Environment:** Jupyter Notebooks, Google Colab, or a suitable IDE.

Team Members and Contribution:

- ★ **[Team Member 1 Name]:** (e.g., Project Lead, Data Acquisition & Preprocessing, Statistical Modeling) - Responsible for data collection, cleaning, initial data exploration, and implementation of statistical time series models (ARIMA, ETS).
- ★ **[Team Member 2 Name]:** (e.g., Feature Engineering, AI/ML Modeling) - Focused on feature engineering techniques, building and training AI/ML models (RNNs, LSTMs, Transformers), and hyperparameter tuning.
- ★ **[Team Member 3 Name]:** (e.g., Exploratory Data Analysis, Visualization & Evaluation) - Conducted in-depth exploratory data analysis, created visualizations, and performed model evaluation and interpretation of results.