**GenAI-Powered Stock Market Analysis: Real-Time Price Prediction Using Sentiment and Historical Trends**

**Problem Statement:**

The stock market is influenced by a complex interplay of factors, including real-time news, social media sentiment, and historical price trends. However, traditional prediction methods fail to integrate these aspects effectively. This project leverages Generative AI (GenAI) to enhance sentiment analysis and historical data processing, enabling real-time, accurate stock price predictions tailored to the Indian market. The aim is to create an innovative solution that empowers traders and analysts with actionable insights.

**Importance of the Project:**

* Retail and Institutional investors rely on timely insights to make decisions.
* Sentiment data from news and social media significantly impacts stock prices.
* By combining sentiment and historical data, predictions can be more accurate.

**Scope:**

This project aims to develop a **predictive model for Indian stock prices** using **real-time data** and **sentiment analysis** to help retail investors and traders make more informed decisions. The project will combine historical stock data and sentiment data from **free sources** (such as news articles, blogs, and financial reports) to forecast stock price movements. The project will focus on stocks listed in the **Indian stock market** (NSE/BSE).

**Scope Includes:**

1. **Data Collection**:
   * Gather historical stock price data (NSE/BSE).
   * Collect sentiment data from free and open sources such as news websites, blogs, and social media posts from platforms that provide free access, such as Reddit, financial news APIs, or free sentiment data from sources like **NewsAPI** or **GDELT**.
2. **Data Preprocessing**:
   * Clean and preprocess stock price data and sentiment data.
   * Align the sentiment data with corresponding stock prices based on timestamps.
3. **Sentiment Analysis**:
   * Use **Natural Language Processing (NLP)** techniques to analyze sentiment from the text data (e.g., news articles, blog posts).
   * Use open-source sentiment analysis models like **VADER** or **TextBlob** to extract sentiment.
4. **Feature Engineering**:
   * Derive meaningful features from both stock price data (e.g., moving averages, price volatility) and sentiment data (e.g., sentiment score, keyword analysis).
5. **Model Building**:
   * Develop a predictive model using **machine learning algorithms** such as **Random Forest**, **Support Vector Machine (SVM)**, or **LSTM (Long Short-Term Memory)** for time-series forecasting.
   * Optimize the model for accuracy and predictive power.
6. **Deployment**:
   * Deploy the model into an **interactive dashboard** for real-time predictions. Use open-source frameworks like **Streamlit** or **Dash**.
   * Ensure the dashboard is accessible for end-users, with real-time predictions based on live data.
7. **Evaluation**:
   * Assess model performance using appropriate metrics such as **Mean Absolute Percentage Error (MAPE)**, **Precision**, and **Recall**.

**Objectives:**

1. **Data Collection**:
   * Collect reliable and up-to-date historical stock data for Indian stocks (NSE/BSE).
   * Gather sentiment data from open-source platforms such as **Reddit**, **NewsAPI**, **GDELT**, or other free sources.
2. **Data Preprocessing**:
   * Clean and preprocess the data (handling missing values, duplicates, normalization, etc.).
   * Merge sentiment data with stock price data on timestamps to align sentiment with corresponding stock prices.
3. **Sentiment Analysis**:
   * Apply NLP techniques (using open-source libraries such as **TextBlob** or **VADER**) to analyze sentiment from social media, news sources, and blogs.
   * Convert text data into numerical sentiment scores that can be used as features in predictive models.
4. **Model Development**:
   * Build and train machine learning models to predict stock price movements based on historical data and sentiment.
   * Explore multiple models (e.g., **Random Forest**, **SVM**, **LSTM**) to determine the best-performing model for this task.
5. **Model Evaluation**:
   * Evaluate the performance of the predictive models using metrics such as **MAPE**, **Precision**, **Recall**, and **Accuracy**.
   * Perform model optimization (e.g., hyperparameter tuning) to improve performance.
6. **Deployment**:
   * Develop an interactive dashboard for real-time stock price predictions, using frameworks like **Streamlit** or **Dash**.
   * Deploy the dashboard on a free cloud service (e.g., **Heroku**, **Google Cloud**).
7. **Documentation and Reporting**:
   * Document the methodology, results, and challenges faced during the project.
   * Prepare a report and presentation showcasing the findings and demonstrating the predictive model’s capabilities.