Market Sentiment Analysis

Avni Gupta (B23EE1007) | Krish Jain (B23CM1019) | Nishchal Badaya (B23CM1053) | Sagar Ratna Chaudhary (B23CM1034)

Satyam Jha (B23CS1066) | Shivam (B23EE1068)

April 14, 2025

Abstract

Traditional technical analysis often fails to capture the nuanced impact of market sentiment on stock prices. News events across a diverse range of sectors (oil and gas, refining, retail, telecom, digital, renewable energy, and infrastructure) have a significant influence on investor behavior and market trends. RIL, one of India's largest and most diversified conglomerates, is affected by these multiple factors. However, there is a gap in systematically quantifying this sentiment and integrating it with technical indicators.

Contents

1 Introduction

Predicting stock price movements is a complex and crucial task for investors and financial analysts. While traditional technical analysis, relying on historical price and volume data, provides valuable insights, it often falls short in capturing the dynamic and often irrational influence of market sentiment. Market sentiment, reflecting the overall attitude of investors towards a particular stock or the market as a whole, is significantly shaped by a continuous stream of news and events spanning various sectors. These events, whether related to oil and gas, refining, retail, telecom, digital technologies, renewable energy, or infrastructure, can trigger shifts in investor confidence and subsequently impact stock valuations.

Despite the acknowledged importance of market sentiment, there remains a significant gap in systematically quantifying this intangible factor and effectively integrating it with established technical analysis frameworks.

To address this challenge, this project aims to delve into the realm of market sentiment analysis, specifically focusing on its impact on the stock price of Reliance Industries Limited. By leveraging the power of machine learning techniques, this research seeks to develop a methodology for systematically quantifying market sentiment derived from news events across relevant sectors and integrating it with traditional technical indicators. The ultimate goal is to create a more comprehensive and insightful model that can better capture the multifaceted drivers of RIL's stock price fluctuations, thereby offering a more nuanced understanding of market dynamics and potentially improving predictive capabilities.

2 Approaches Tried

Our approach is designed to emphasize the use of traditional, interpretable machine learning techniques, ensuring that every step is explainable. Advanced models (e.g., FinBERT) are applied only sparingly—on a small, representative subset—to generate high-quality labels without incurring excessive computational costs.

2.1 Data Preprocessing and Filtering

- **Preprocessing:** Clean the text data by removing HTML, punctuation, and stopwords; then tokenize and normalize the text.
- **Filtering for RIL Relevance:** Apply a combination of keyword-based filtering and NER to extract articles that mention RIL directly or include relevant entities. This ensures that our analysis focuses on news that can impact RIL.

2.2

2.2.1 Approach 1: Traditional ML Sentiment Estimation via FinBERT, LDA, and Regression

Labeling a Subset of Articles Using FinBERT

- **Subset Labeling:** From the filtered dataset, select a small, representative subset of articles and assign sentiment scores using FinBERT.
- **Rationale:** FinBERT is highly accurate for financial texts, and using it on a limited subset provides reliable sentiment labels. This labeled subset serves as a "ground truth" to train our regression model without the need to process the entire dataset with a computationally expensive advanced model.

Topic Modeling with LDA

- **Applying LDA:** Convert the cleaned text into a document–term matrix using tools such as scikit-learn's `CountVectorizer`. Then apply Latent Dirichlet Allocation (LDA) to extract latent topics from the corpus.
- **How LDA Works:** LDA assumes each document is a mixture of topics and each topic is a probability distribution over words. Using Dirichlet priors for both document–topic and topic–word distributions, LDA uses inference techniques (e.g., Gibbs Sampling) to determine:
  - **Topic–Word Distributions:** Lists of top words for each topic, which are interpretable.
  - **Document–Topic Distributions:** A probability vector for each document indicating the contribution of each topic.
- **Explainability:** LDA's output is highly interpretable; analysts can inspect the top words per topic to understand the thematic structure (e.g., topics about "oil price volatility," "retail performance," "telecom expansion," etc.).

Estimating Sentiment via Traditional ML Regression

- **Training a Regression Model:** With the labeled subset (using FinBERT sentiment scores) and their corresponding LDA-derived topic distributions, train a non-linear regression model using XGBoost or Random Forest.
  - **Inputs:** Topic distributions for each article.
  - **Target:** FinBERT-derived sentiment scores.
- **Interpretability and Advantages:** Traditional ML models like XGBoost and Random Forest are less "black box" than deep learning models. They allow us to examine feature importance and decision pathways, giving us insight into which topics have the most influence on sentiment.
- **Output:** The regression model learns sentiment weights for each topic. Then, for each article, the overall sentiment score is computed as:

$$\text{Estimated Sentiment Score} = \sum_{i=1}^{N} (\text{Topic Probability}_i \times \text{Sentiment Weight}_i)$$

  This approach scales sentiment estimation efficiently over the entire dataset.

2.2.2 Approach 2: Using XGBoost Regression and SARIMAX with EWMA Sentiment

- This approach focuses on directly predicting numerical sentiment scores using a regression model and integrating these scores into a time series forecasting model (SARIMAX) after smoothing.

Sentiment Value Prediction using XGBoost Regression

- **Dataset:** A labeled dataset titled "Indian Financial News" was sourced from Hugging Face. This dataset contains financial news articles relevant to the Indian market, along with pre-assigned sentiment labels or scores.
- **Model Selection:** An XGBoost (Extreme Gradient Boosting) regressor was chosen for the sentiment prediction task. XGBoost is an efficient and scalable gradient boosting framework known for its performance on diverse datasets. It builds trees sequentially, with each new tree correcting errors made by the previous ones.
- **Training and Testing:** The XGBoost regressor was trained and evaluated using the labeled "Indian Financial News" dataset. The model learned to predict numerical sentiment values based on the text features of the news articles.
- **Application:** The trained XGBoost model was then applied to the previously scraped and preprocessed news article dataset (relevant to RIL) to predict a sentiment score for each article.

Dataset Transformation Techniques: Doc2Vec and Word2Vec

- To train the XGBoost regressor on the labeled dataset titled Indian Financial News, it was necessary to transform the textual input into numerical vectors. Two distinct vectorization techniques—Doc2Vec and Word2Vec—were explored for this purpose.
- **Doc2Vec :** It is an extension of Word2Vec that generates fixed-length vector representations for entire documents, capturing the semantic meaning of words and their relationships within a document.
  - It utilizes a neural network where a unique document ID vector is added to the word vectors during training, enabling the model to learn document-level representations.
  - While Doc2Vec provides a comprehensive representation of the entire document, it may not capture granular word-level sentiment effectively, especially for shorter texts or highly contextual sentiment analysis tasks
- **Word2Vec :** Word2Vec focuses on generating continuous vector representations for individual words based on their context in a corpus. It uses models like Continuous Bag of Words (CBOW) or Skip-Gram to learn word embeddings
  - By capturing semantic relationships between words, Word2Vec allows for a more fine-grained understanding of sentiment at the word level, which can be aggregated to represent document-level sentiment
  - Compared to Doc2Vec, Word2Vec is more efficient in capturing sentiment nuances as it operates at the word level, making it better suited for sentiment prediction tasks where word-to-word relationships are critical
- **TF-IDF Weighted Average Technique :** Instead of using a simple average of all word vectors to compute the aggregate vector for each document, a TF-IDF weighted averaging technique was implemented. This approach calculates the importance of each word in a document relative to its occurrence across the corpus and uses these weights to scale the corresponding word vectors.
  - **Benefits of TF-IDF Weighted Averaging:**
    - Enhances representation by giving higher importance to words that are more relevant within a specific document while down-weighting common words that appear frequently across all documents
    - Improves sentiment prediction accuracy by emphasizing critical terms that contribute more significantly to sentiment polarity.
    - Combines the contextual richness of Word2Vec embeddings with the statistical weighting of TF-IDF, resulting in more informative and discriminative document vectors
- Word2Vec was chosen over Doc2Vec due to its ability to capture sentiment at a granular level (word-to-word basis), which is crucial for aggregating sentiment scores across documents. The integration of TF-IDF weighted averaging further refined document-level representations, ensuring that critical terms influenced sentiment predictions effectively. This combination proved more efficient for market sentiment analysis tasks involving financial news articles.

Feature Engineering for Sentiment Value Prediction

- **Feature Extraction:** After converting the text data into numerical vectors using Word2Vec, Principal Component Analysis (PCA) was applied as a feature extraction technique. PCA helps reduce the dimensionality of the feature space by identifying the principal components that explain the maximum variance in the data. This technique was used to retain only the most significant features while preserving the overall structure of the dataset
- **Feature Selection:** To identify and remove irrelevant or redundant features, several feature selection techniques were implemented:
  - **Variance Threshold:** A variance threshold was manually set after analyzing the count of features at different thresholds. Features with variance below this threshold were eliminated, as they contributed manually to distinguishing between sentiment scores.

  - **Correlation Analysis:** Highly correlated features were identified, and one of each pair of correlated features was dropped to avoid redundancy and multicollinearity, which could negatively impact model performance.
  - **Lasso Regression:** Lasso regression was applied to perform feature selection by penalizing less important features through regularization. This technique effectively reduced the number of irrelevant features while maintaining predictive accuracy.

- Despite applying these techniques, it was observed that all features contributed meaningfully to sentiment prediction and collectively resulted in the least MSE. Therefore, no feature reduction was performed, as retaining all features provided optimal performance for the XGBoost regressor.These steps ensured that the model leveraged all available information effectively while maintaining predictive accuracy

2.3 Sentiment Time Series Generation and Smoothing with EWMA

- **Aggregation:** The predicted sentiment scores for individual articles were aggregated over time (e.g., calculating the daily average sentiment score) to construct a sentiment time series.
- **Smoothing:** An Exponentially Weighted Moving Average (EWMA) was applied to the aggregated sentiment time series. EWMA assigns greater weight to more recent data points, helping to smooth out short-term noise and emphasize underlying trends.
- A span of 3 was used for the EWMA calculation. The span parameter relates to the smoothing factor alpha by the formula $= 2 / (\text{span} + 1) = 2/(\text{span}+1)$. With a span of 3, the effective alpha is 0.5, giving significant weight to recent observations.

2.4 Stock Data Integration and Baseline Modeling (ARIMA and SARIMA)

- **Stock Data:** Historical closing price data for Reliance Industries Limited (RIL) was obtained for the period from 2023 to March 2025.
- **Baseline Model:** To identify the most suitable model for predicting stock price time series data, several statistical models were implemented and compared:
  - **ARIMA (Auto-Regressive Integrated Moving Average):** ARIMA was used to model the stock price time series based on its autoregressive and moving average components.
  - **SARIMA (Seasonal Auto-Regressive Integrated Moving Average):**SARIMA extended ARIMA by incorporating seasonality into the model, addressing periodic patterns in the stock price data.

2.5 Incorporating Smoothed Sentiment with SARIMAX

- **SARIMAX Model:** A SARIMAX (Seasonal Autoregressive Integrated Moving Average with Exogenous variables) model was employed. SARIMAX extends SARIMA by allowing the inclusion of external predictor variables (exogenous variables) that might influence the target time series.
- **Sentiment Integration:** The smoothed EWMA sentiment time series, generated in Step 2, was incorporated as an exogenous variable into the SARIMAX model.
- **Training and Testing:** The SARIMAX model was trained on the RIL closing prices and the corresponding EWMA sentiment values for the period 2023-2024. It was then tested on the January 2025 - March 2025 period.
- **ACF and PACF Analysis:** The Auto-Correlation Function (ACF) and Partial Auto-Correlation Function (PACF) plots were analyzed to determine the order of autoregressive and moving average terms. These plots provided insights into the lagged dependencies in the time series data.
- **Parameter Optimization:**AutoARIMA was applied to automatically identify the best parameters for SARIMAX, ensuring optimal model configuration for stock price prediction.

- **Error Analysis:** The prediction errors (e.g., Root Mean Squared Error - RMSE, Mean Absolute Error - MAE) of the SARIMAX model were compared to those of the baseline ARIMA and SARIMA models. This analysis aims to quantify the improvement in stock price prediction accuracy achieved by incorporating the EWMA-smoothed sentiment data.

- Based on comparative analysis, SARIMAX was identified as the best-performing model for this project. Incorporating sentiment scores as an exogenous variable significantly improved prediction accuracy by capturing the influence of market sentiment on stock prices. This combination of statistical modeling and sentiment integration proved to be highly effective for time series forecasting in financial markets.

2.6 Future Stock Price Forecasting Methodology

- **Final Model Training:** The SARIMAX model (predicting RIL closing price using EWMA sentiment as the exogenous variable) was re-trained using the entire available dataset (stock prices and EWMA sentiment from 2023 to March 2025).

- **Forecasting Exogenous Variable:** To forecast RIL stock prices for dates beyond March 2025, future values of the exogenous variable (EWMA sentiment) are required. A separate time series model (e.g., another SARIMA model) was trained specifically on the historical EWMA sentiment time series (2023 - March 2025). This model was used to forecast future EWMA sentiment values for the desired prediction horizon.

- **Sequential Stock Price Prediction:** The forecasted EWMA sentiment values were then fed sequentially into the trained SARIMAX model. For each future day, the predicted stock price was generated using the forecasted EWMA sentiment value for that day, allowing for multi-step-ahead forecasting of RIL's stock price.

3 Experiments and Results

3.1 Dataset Description

3.1.1 Data Source

- **Economic Times News Articles:** We have collected news articles from the Economic Times that cover various industries and sectors affecting the Indian financial market and RIL. These sectors include oil & gas, refining & petrochemicals, retail, telecommunications, digital services, renewable energy, and infrastructure.

3.1.2 Data Duration

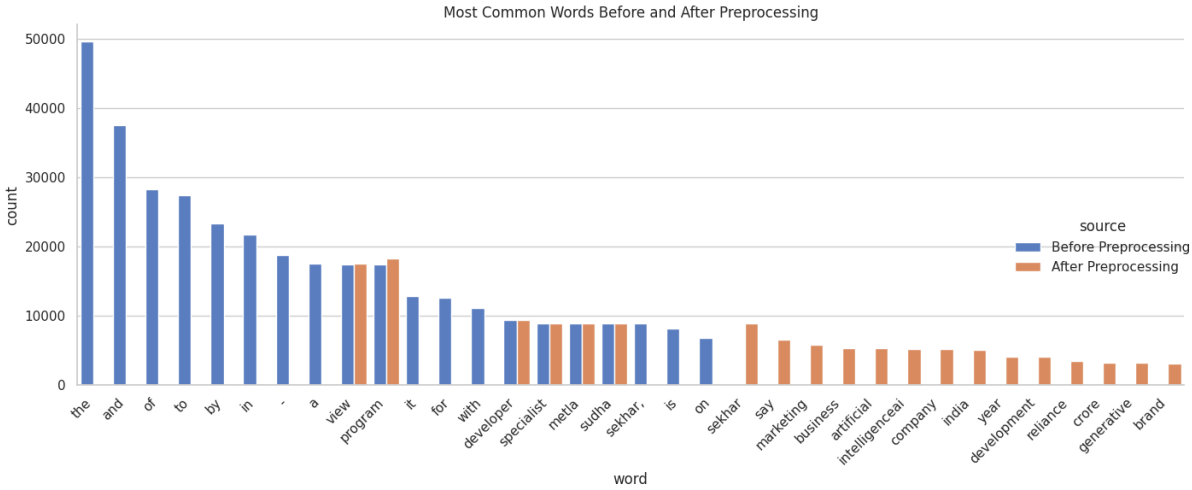
- **Time Period:** The dataset spans from 2023 to the present, capturing recent market dynamics in the post-pandemic era.

3.1.3 Data Characteristics

- **Content:** The scraped data includes article titles, publication dates, full text, and other metadata. Although the dataset covers a wide range of topics, we have filtered it to retain only articles that are directly or indirectly related to RIL.
- **Filtering Process:**
  - **Named Entity Recognition (NER)** techniques were applied to identify mentions of entities such as "Reliance," "RIL," "Mukesh Ambani," etc., ensuring relevance to RIL.
- Keyword-based filtering was also implemented to extract articles containing specific terms related to RIL's operations or influence in the market. This dual-filtering approach ensured that only pertinent articles were included for sentiment analysis and stock price prediction tasks.

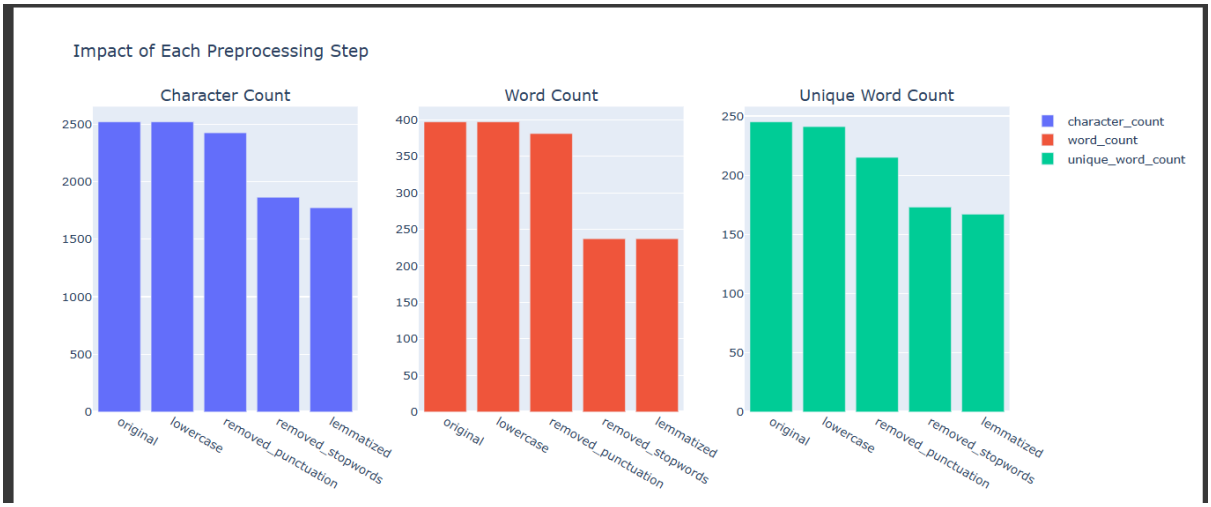
3.1.4 Data Preprocessing

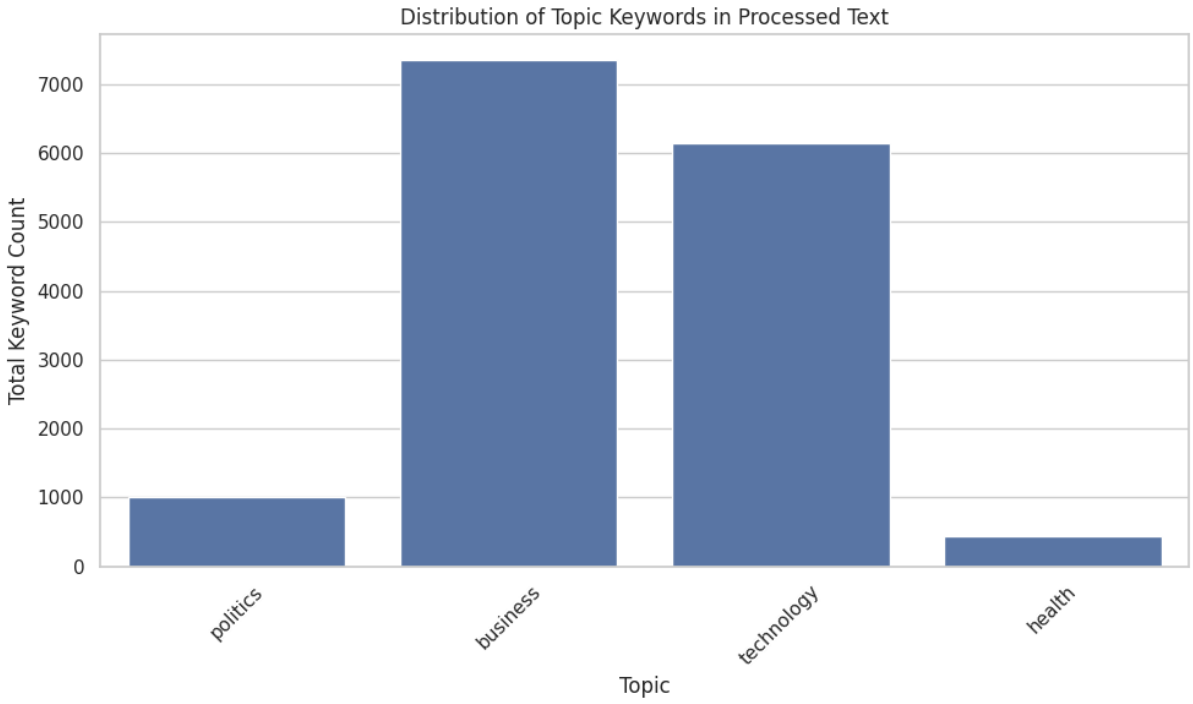
- A detailed preprocessing pipeline was implemented to clean and standardize the text data before analysis:
  - **Lowercasing:** All text was converted to lowercase for uniformity and to prevent case-sensitive mismatches during vectorization.
  - **Removing Punctuation:**Punctuation marks were removed to focus solely on meaningful textual content.
  - **Stopword Removal:** Common stopwords were eliminated to reduce noise and highlight relevant terms in the dataset.
  - **Lemmatization:** Words were reduced to their root forms using lemmatization techniques, ensuring consistency in word representations (e.g., "running" $\rightarrow$ "run").
- **Analysis of Preprocessing Impact:**
  - Word and character counts were analyzed before and after each preprocessing step to evaluate their impact on the dataset size and quality.



– Sample headlines were processed step-by-step, verifying that each transformation (e.g., lowercasing, punctuation removal) was applied correctly and improved data relevance for sentiment analysis tasks.

– The most common words were plotted before and after preprocessing using bar plots to assess their relevance to the task. After preprocessing, domain-specific terms like "reliance," "energy," and "market" became more prominent, validating the effectiveness of the pipeline.





- This robust preprocessing ensured that the data was clean, consistent, and ready for advanced analysis techniques like Word2Vec embedding for sentiment score prediction and statistical modeling for stock price forecasting.

3.2 Experimental Setting:

3.2.1 Sentiment Score Prediction:

- The text data was converted into numerical vectors using Word2Vec embeddings, followed by training and testing an XGBoost regressor on the labeled dataset.
- Various feature engineering techniques were applied, including PCA for feature extraction and methods like variance threshold, correlation analysis, and Lasso regression for feature selection. Despite these efforts, all features were retained as they collectively minimized Mean Squared Error (MSE).
- The sentiment scores for scraped news articles were further smoothed using Exponentially Weighted Moving Average (EWMA) with a span of 3 to calculate daily sentiment scores.

3.2.2 Stock Price Prediction:

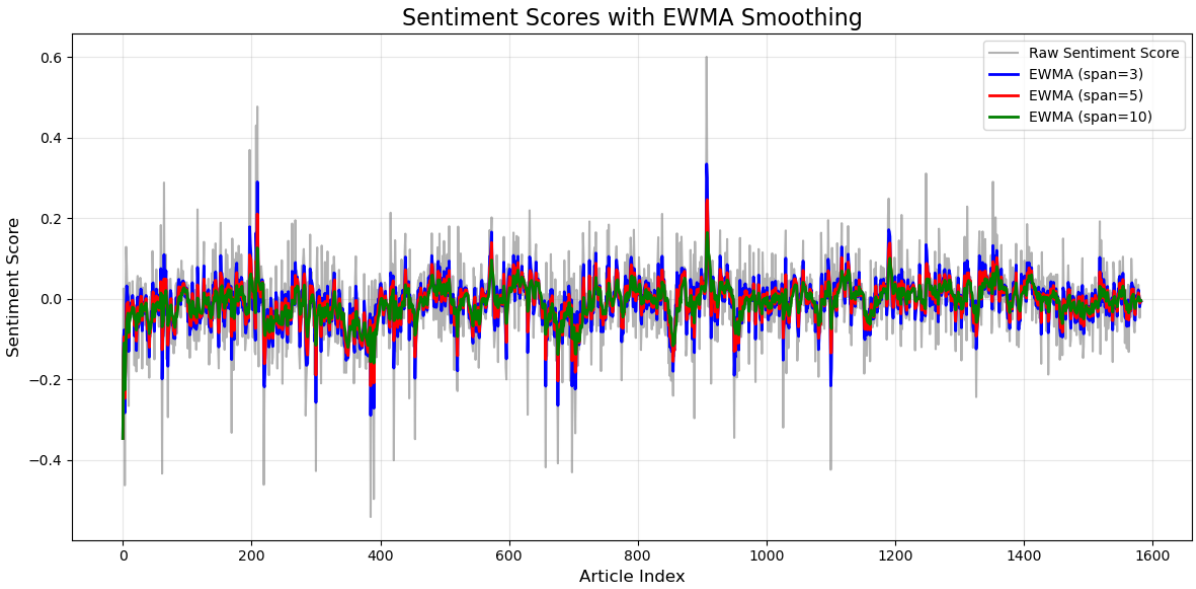
- Statistical models such as ARIMA, SARIMA, and SARIMAX were implemented on stock price time series data to determine the best-performing model.
- SARIMAX was chosen as the optimal model after incorporating sentiment scores as an exogenous variable, which significantly improved prediction accuracy.
- ACF and PACF plots were analyzed to identify lagged dependencies in the time series data, and AutoARIMA was applied to optimize parameters for SARIMAX.

3.3 Results:

3.3.1 Sentiment Score Prediction:

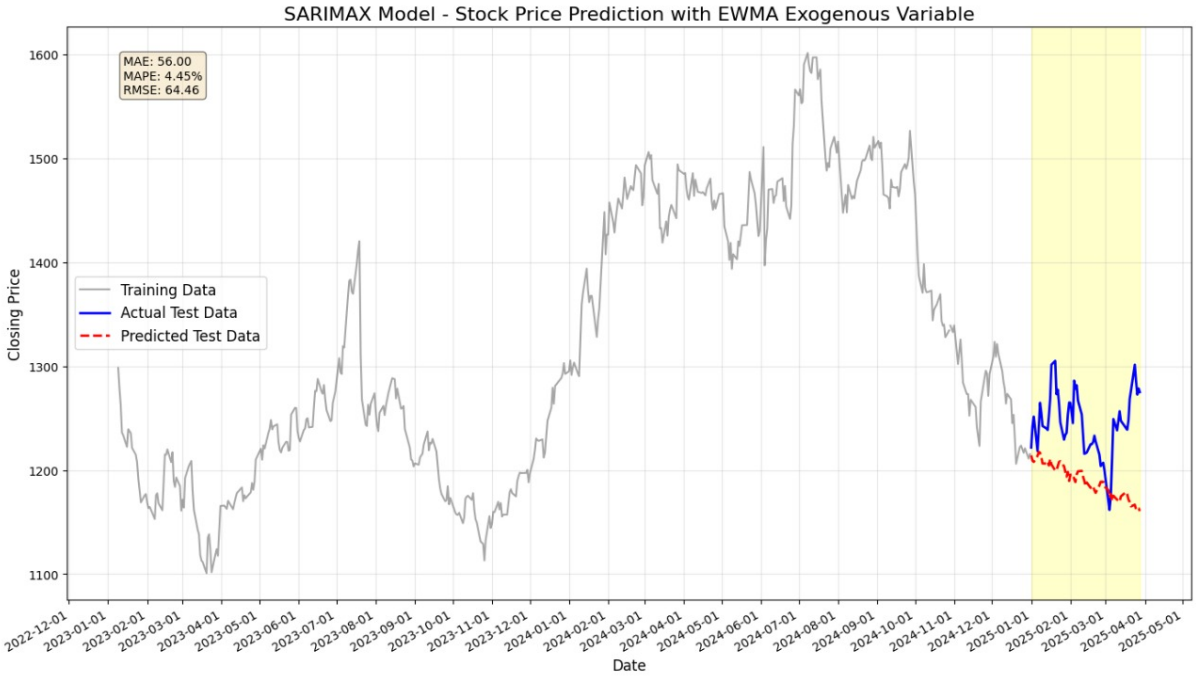
The XGBoost regressor achieved minimal MSE when trained on all features, demonstrating that each feature contributed meaningfully to sentiment prediction.

Smoothed Senitment scores for scraped news articles using EWMA for different spans

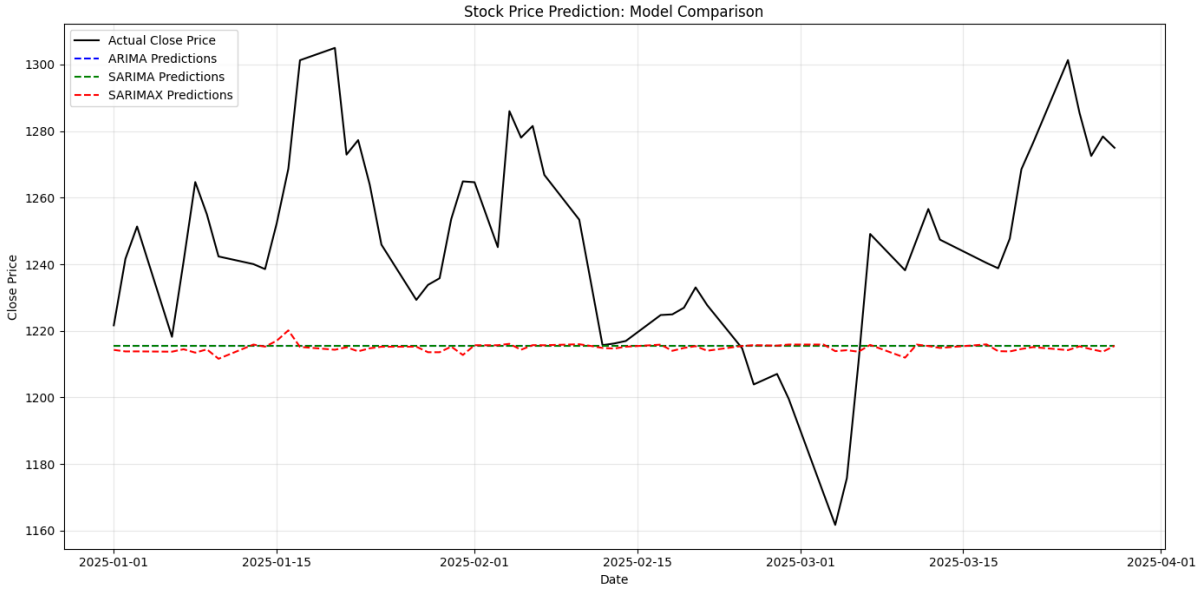


3.3.2 Stock Price Prediction:

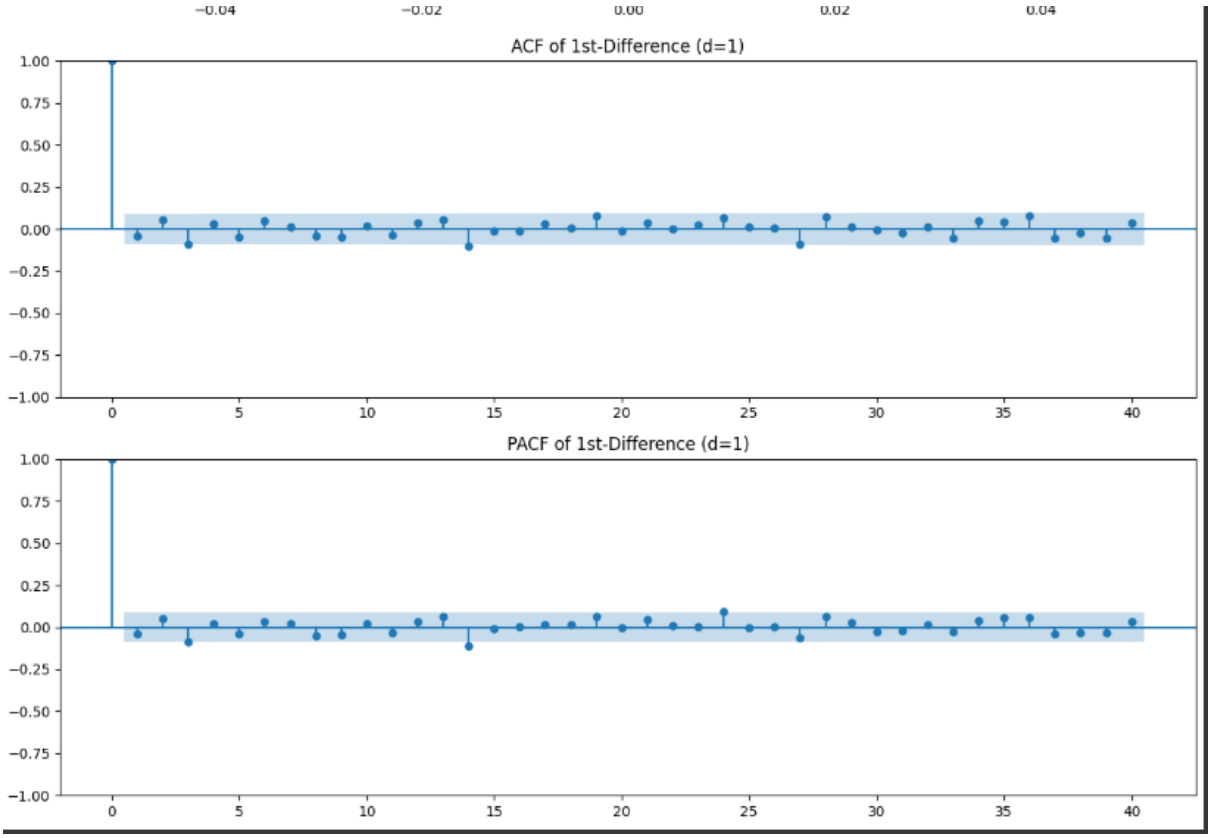
- To predict stock prices, multiple statistical models were implemented and compared to identify the most suitable approach. The models used were ARIMA, SARIMA, and SARIMAX. The SARIMAX model incorporated sentiment scores as an exogenous variable, enhancing its ability to capture external influences on stock price movements. The following observations were made:
  - **Testing Accuracy:** SARIMAX demonstrated better performance compared to ARIMA and SARIMA in terms of testing accuracy. Metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE) confirmed its predictive accuracy.
    SARIMAX Model Evaluation:
    MAE : 37.4163
    RMSE : 43.9098
    MAPE : 2.9738



– **Visual Comparison of Predicted Prices:** Plots comparing the predicted stock prices from each model against actual values revealed that SARIMAX closely followed the trends of the actual stock prices, outperforming ARIMA and SARIMA. This visual evidence further validated the effectiveness of SARIMAX for time series forecasting.



- **ACF and PACF Analysis** Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots were analyzed to determine the optimal parameters for the SARIMAX model:
  - **ACF Analysis:** Helped identify lag values for the moving average component (q) by examining correlations between the time series and its lagged values.
  - **PACF Analysis:**Assisted in determining lag values for the autoregressive component (p) by isolating direct relationships between lagged values and the current value of the time series.


−0.04
−0.02
0.00
0.02
0.04
ACF of 1st-Difference (d=1)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30
35
40
PACF of 1st-Difference (d=1)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
5
10
15
20
25
30
35
40

4 Summary

4.1 Project Overview

This project aimed to perform market sentiment analysis and stock price prediction by leveraging financial news articles and stock price time series data. The primary focus was on Reliance Industries Limited (RIL), using sentiment scores derived from scraped news articles to enhance stock price forecasting accuracy.

4.2 Key Components

4.2.1 Dataset Collection and Filtering:

- News articles were scraped from The Economic Times, focusing on industries relevant to RIL, such as oil gas, petrochemicals, retail, telecommunications, and renewable energy.
- A robust filtering process was applied using Named Entity Recognition (NER) and keyword-based techniques to retain articles directly or indirectly related to RIL.

4.2.2 Data Preprocessing:

- Text preprocessing steps included lowercasing, punctuation removal, stopword removal, and lemmatization.
- Word and character counts were analyzed before and after preprocessing to evaluate the impact of each step on dataset quality.
- Common words were plotted before and after preprocessing to ensure relevance to the task, with domain-specific terms like "reliance," "energy," and "market" becoming prominent post-cleaning.

4.2.3 Sentiment Score Prediction:

- Sentiment scores were predicted using an XGBoost regressor trained on labeled financial news datasets.
- Word2Vec embeddings were used for vectorization, with TF-IDF weighted averaging applied to aggregate word vectors for document representation.
- Feature engineering techniques like PCA, variance thresholding, correlation analysis, and Lasso regression were explored but ultimately all features were retained due to their collective contribution to minimizing Mean Squared Error (MSE).
- Sentiment scores were smoothed using Exponentially Weighted Moving Average (EWMA) with a span of 3 for daily aggregation.

4.2.4 Stock Price Prediction:

- Statistical models including ARIMA, SARIMA, and SARIMAX were implemented on stock price time series data.
- SARIMAX was identified as the best-performing model due to its ability to incorporate sentiment scores as an exogenous variable, significantly improving prediction accuracy.
- ACF and PACF plots were analyzed to determine optimal parameters for SARIMAX, with AutoARIMA applied for parameter optimization.
- Visual comparisons of predicted prices from different models demonstrated SARIMAX's superior performance.

4.3 Results

- Sentiment scores derived from financial news articles effectively captured market sentiment relevant to RIL.
- SARIMAX outperformed other models in stock price prediction tasks, achieving the highest accuracy and closely tracking actual price trends.
- Incorporating sentiment scores as exogenous variables proved critical in enhancing forecasting performance.

4.4 Conclusion

The project successfully demonstrated the integration of sentiment analysis with statistical modeling for stock price prediction. By combining advanced text processing techniques with time series forecasting methods, it provided valuable insights into the relationship between market sentiment and stock price movements for RIL. These findings highlight the potential of sentiment-driven predictive models in financial markets.

4.5 Future Scope

- Extend the analysis to other companies or sectors to validate the generalizability of the approach.
- Explore deep learning-based models such as LSTMs or Transformers for improved time series forecasting.

4.6 Google Cloud

Google Cloud has been a reliable and flexible platform for deploying the PRML (Pattern Recognition and Machine Learning) project. It allowed us to containerize our Streamlit application using Docker and deploy it effortlessly with services like Cloud Run. The seamless integration of various tools made the process smooth, from handling dependencies to managing deployment. Despite a few initial challenges, We found the platform efficient in handling resources and providing real-time access to my ML models through a web interface. Overall, it was a valuable experience working with Google Cloud for this project.

A Contribution of Each Member

1. Avni Gupta: Done Initial Scrapping and combined the whole dataset.Implemented various baseline model like ARIMA and SARIMA and tested their performance.

2. Krish Jain: Done preprocessing detailed analysis by plotting graphs for common words or unique words before and after preprocessing , Implemented Word2Vec technique for transformation of the text data updated parameters for word2vec using grid search.

3. Nishchal Badaya: Integrated Sentiment score as exogenous variable in SARIMAX and tested its accuracy on past data.Done the analysis of ACF and PACF plot to find out the best parameters for SARIMAX to improve the prediction.

4. Sagar Ratna Chaudhary: Implemented Doc2vec technique parralelly with word2vec and tested its accuracy. He also applied TF-IDF weighted average technique to aggregate word vectors over whole document and finally trained XG boost model for it with updated parameters.

5. Satyam Jha: Done intital preprocessing of the scrapped data. Collected RIL stock data. Also made the project page and deployment on Google Cloud.Also made separate SARIMA model for predicting EWMA value.

6. Shivam: Filtered the scrapped dataset using NER and keyword based techniques. Also applied trained XG boost model on the scrapped dataset and predicted sentiment score and integrated it with EWMA.