**Social Media Trend Analysis Using Sentiment Forecasting**

**School of Computer Science Engineering and Technology**

|  |
| --- |
| **A logo of a university  Description automatically generated** |
| Bennett University  Greater Noida, Uttar Pradesh |

Submitted by Submitted to

Arya Yadav, E22CSEU0513 Mentor name

Shaurya Singh, E22CSEU0539

### I. Introduction and Objective

### In the hyper-connected era of the 21st century, where information travels faster than the speed of light, social media has emerged as a monumental force in shaping public opinion, driving societal narratives, and influencing global events. Platforms like Twitter, Reddit, and Instagram no longer serve merely as spaces for sharing personal anecdotes or viral memes; they have evolved into critical arteries of collective human consciousness. Every trending hashtag, every viral tweet, and every heated comment thread reflects the emotions, beliefs, and attitudes of millions in real-time. Recognizing this phenomenon, our project, *Social Media Trend Analysis Using Sentiment Forecasting*, aims to harness the chaotic yet powerful stream of social media data, transform it into structured insight, and, most importantly, predict its future trajectory.

### The primary objective of this project is to develop a robust framework capable of analyzing and forecasting sentiment trends on social media platforms using advanced machine learning and time series modeling techniques. While traditional sentiment analysis often focuses on retrospective interpretation—essentially asking, "What did people feel?"—our project ventures into a more complex and impactful question: "What *will* people feel next?" Forecasting public sentiment holds tremendous potential across multiple sectors. Businesses can fine-tune their marketing campaigns or prepare for PR challenges, political strategists can anticipate voter moods, investors can gauge market sentiment, and researchers can study the evolution of public discourse.

### The motivation for choosing this project stems from the observed gap in current academic and industry practices. Despite the explosion of sentiment analysis tools, very few initiatives dare to predict future sentiments at scale, especially considering the non-linear, noisy, and volatile nature of social media data. Moreover, most existing studies tend to focus on static classification rather than dynamic forecasting, leaving an open field for exploration. By choosing this project, we wanted not only to address this gap but also to push the boundaries of what sentiment analysis can achieve when combined with rigorous time series modeling.

### Social media’s influence in our lives today is unparalleled. From dictating stock market fluctuations (as seen during the infamous GameStop short squeeze driven by Reddit’s r/WallStreetBets) to amplifying political movements like the Arab Spring, the collective voice of online communities wields real-world power. For businesses, this translates to a need for adaptive marketing strategies and proactive reputation management. For governments, it becomes a matter of public policy and civil response. For individuals, it represents empowerment through connectivity. Recognizing the criticality of social media's role, this project aims to equip stakeholders with a tool to not only understand but also anticipate mass sentiment shifts.

### Our core research question is therefore straightforward yet challenging: Can we accurately forecast social media sentiment using a combination of time series analysis and machine learning models, and can these forecasts have practical, real-world utility? Framed differently, can data-driven methods make the unpredictable nature of human emotion slightly more predictable when viewed through the lens of digital expression?

### In attempting to answer this question, we embarked on a journey that was technically demanding and intellectually stimulating. Time series forecasting in a social media context introduces several complexities — erratic behavior, sudden trend reversals, exogenous shocks (such as a viral meme or a breaking news story), and noisy, unstructured data. Furthermore, human emotions are not easily quantifiable or linearly evolving phenomena; they exhibit abrupt shifts and contextual dependencies that traditional models struggle to capture.

### Our approach sought to address these challenges through a multi-pronged strategy. Rather than relying solely on a single model or technique, we employed a stack of methods designed to complement each other. Passive-Aggressive Classifiers (PAC) were deployed for fast and scalable real-time classification tasks, excelling in imbalanced datasets with high-dimensional feature spaces. Long Short-Term Memory (LSTM) networks, specialized in handling sequential data, were harnessed to capture deeper temporal dependencies in sentiment evolution. Classical models such as ARIMA and SARIMA were incorporated to manage stationary time series data, providing transparent insights through parametric modeling. Prophet, developed by Facebook, was tested for trend and seasonality decomposition, although its performance revealed limitations in handling the irregular spikes characteristic of social media data.

### Through this project, we aim to demonstrate that sentiment forecasting, while challenging, is not only achievable but also extremely valuable. More importantly, we illustrate that by combining machine learning with time series analysis in a thoughtful, model-aware manner, it is possible to move beyond mere analysis into actionable foresight. In a world where public sentiment can be as volatile as financial markets, having a predictive edge could very well be the next big competitive advantage.

### In conclusion, this project stands as a testament to the power of interdisciplinary techniques—merging data science, behavioral analytics, and forecasting theory—to tackle one of the most elusive and dynamic phenomena of the modern digital era: the future of human emotion, as reflected through the pulse of social media.

### II. Data Collection, Preprocessing, and Exploratory Data Analysis (EDA)

### Data serves as the lifeblood of any data science project, but in sentiment forecasting, it is even more critical. Poor quality data is like building a fortress on quicksand: no matter how sophisticated the models, the foundation collapses. Our project demanded data that was not only voluminous but also diverse, representative, and temporally rich. To ensure this, we sourced data from a combination of live API pulls and curated datasets, each selected with surgical precision for the strengths it could bring to different stages of modeling.

### Dataset Description

### 1. Twitter Data via Twitter API: The primary source of real-time public opinion was Twitter. Tweets were collected over a span of 6 to 9 months using keywords and trending hashtags relevant to major domains such as technology, politics, finance, and culture. Variables included timestamps, tweet content, user metadata, and, importantly, sentiment labels generated via the VADER sentiment analyzer. Approximately 20,000 to 25,000 tweets were gathered, providing a robust timeline of organic sentiment evolution.

### 2. Reddit Dataset from Kaggle: To inject diversity into the tone and style of expressions, we incorporated Reddit comments. Sourced from popular subreddits like r/India, r/technology, and r/cryptocurrency, this dataset provided timestamps, comment text, upvotes, and pre-labeled sentiment annotations. Reddit data added necessary variability because its conversational, less character-constrained nature contrasted the brevity of Twitter posts.

### 3. Kaggle Sentiment-Labeled Dataset: For benchmarking and cross-validation purposes, a curated Kaggle dataset containing pre-annotated tweet sentiments was employed. This dataset enabled us to validate our classification models on unseen, third-party labeled data, enhancing generalizability and reducing overfitting risks.

### The strategic mapping of each dataset to specific models was crucial. The Passive-Aggressive Classifier (PAC), optimized for high-speed text classification, was primarily trained on the Twitter Sentiment and Reddit Comments datasets due to their rich labeled data. The Long Short-Term Memory (LSTM) model, which requires sequence data to learn evolving patterns, was powered by timestamped collections from the Climate Change Tweets and Stock Market Sentiment datasets. Time-series models like ARIMA and Prophet focused on aggregated variables — daily counts of mentions, average sentiment scores, and hashtag trends — sourced from both Twitter and Reddit longitudinal datasets. The hybrid LSTM+ARIMA model leveraged these residual patterns to predict finer sentiment evolutions with higher accuracy. This intelligent data-to-model alignment was non-negotiable; it formed the backbone of achieving robust, reliable, and interpretable results.

### Importance of Preprocessing

### In sentiment forecasting, preprocessing is not just a technical step; it is an act of careful balancing. If ignored or done haphazardly, issues like missing values (NaNs), biased class distributions, timestamp misalignments, and semantic distortions plague the dataset, inevitably degrading model performance. Over-processing, however, is equally catastrophic — aggressive text cleaning can sterilize data, stripping away sarcasm, idiomatic expressions, and emotional subtleties that are essential for accurate sentiment detection. Thus, every preprocessing decision we made was calculated, designed to preserve the soul of the text while structuring it enough for machine learning models to interpret.

### Preprocessing Steps

### 1. Cleaning: URLs, user mentions, hashtags, emojis, and special characters were removed using regular expressions. Although hashtags can carry semantic meaning, most hashtags relevant to our topics were embedded in the text, thus removing them reduced noise without significant information loss. This step ensured that both VADER and deep learning models focused on true semantic content rather than being misled by symbols and links.

### 2. Tokenization: Sentences were broken down into tokens (words) using standard natural language processing (NLP) tokenizers. This step was fundamental for both VADER sentiment scoring and input preparation for neural networks.

### 3. Stopword Removal: Common English words such as "the," "is," and "and" were removed. While these words serve grammatical purposes, they do not contribute to the sentiment-bearing weight of a sentence and hence were eliminated to sharpen model focus.

### 4. Sentiment Scoring: For datasets without labels, we applied VADER to assign soft sentiment scores on a continuous scale, later thresholded into "positive," "neutral," or "negative" categories. This standardization ensured that all datasets adhered to a common sentiment annotation framework.

### 5. Encoding: Sentiment classes were mapped into numeric representations: Positive (1), Neutral (0), and Negative (-1). This transformation was essential for classification models, facilitating faster convergence and straightforward evaluation.

### 6. Timestamp Alignment: Temporal continuity is sacrosanct in time series modeling. Tweets and comments were resampled into daily aggregates based on their timestamps. We calculated the daily average sentiment, total mentions, and hashtag frequencies to form well-structured time series. Gaps (missing days) were imputed via forward-filling methods to maintain the integrity of temporal sequences without injecting artificial trends.

### 7. Normalization: For LSTM models, input sequences were normalized to a 0–1 range. Without normalization, models tend to exhibit unstable gradients during training, leading to either slow convergence or divergence entirely. Scaling the input not only accelerated training but also stabilized the learning dynamics of deep networks.

### 8. Balancing the Dataset: One of the subtle but deadly issues in sentiment data is the overwhelming dominance of "neutral" class samples. Left unchecked, models would bias heavily towards predicting neutral sentiment. We addressed this by carefully downsampling the neutral class and applying mild upsampling for minority classes where necessary, ensuring that the models did not become complacent or misrepresentative in their learning.

### Exploratory Data Analysis (EDA)

### Before diving into modeling, we performed a thorough Exploratory Data Analysis (EDA) to understand the underlying patterns, anomalies, and distributions in our data.

### Distribution Analysis: We plotted histograms for sentiment class distributions across different platforms. Twitter exhibited a heavier negative skew during election seasons, while Reddit displayed more balanced sentiments with slight positive leanings.

### A graph of a distribution AI-generated content may be incorrect.

### A screenshot of a graph AI-generated content may be incorrect.

### A graph of different colored bars AI-generated content may be incorrect.

### A graph of a graph of a number of tweets AI-generated content may be incorrect.

### Word Clouds and Keyword Extraction: By generating word clouds for each sentiment class, we observed that positive tweets often included terms like "hope," "success," and "innovation," while negative tweets were dominated by words like "fail," "crisis," and "scam." This lexical analysis validated the efficacy of our sentiment labeling.

### A black background with words AI-generated content may be incorrect.

### 

### A close up of words AI-generated content may be incorrect. A close-up of words AI-generated content may be incorrect.

### A close up of words AI-generated content may be incorrect.

### A close up of words AI-generated content may be incorrect.

### A close up of words AI-generated content may be incorrect.

### Hashtag Analysis: Top trending hashtags were extracted and visualized. Certain hashtags were highly correlated with sentiment polarity — for instance, tags related to environmental activism often carried strongly positive sentiments, whereas political hashtags were polarizing.

### A graph of hashtags and hashtags AI-generated content may be incorrect.

### A graph of hashtags AI-generated content may be incorrect.

### A graph with different colored lines AI-generated content may be incorrect.

### A graph of hashtags AI-generated content may be incorrect.

### A screenshot of a computer screen AI-generated content may be incorrect.

### Correlation Analysis: We examined correlations between volume of tweets, average sentiment, and external events (e.g., stock market indices, political rallies). Moderate to strong correlations were found, justifying the hypothesis that social media sentiment can be a predictor for real-world outcomes.

### A graph of a diagram AI-generated content may be incorrect.

### A graph of a distribution AI-generated content may be incorrect.

### A graph of a graph AI-generated content may be incorrect.

A graph of a graph

AI-generated content may be incorrect.

### A pie chart with numbers and a few words AI-generated content may be incorrect.

### EDA was not a one-time ritual. It was a living, breathing process that continuously informed our modeling decisions, helped us debug anomalies, and ensured that our assumptions about the data remained grounded in observable reality.

### III. Time Series Modeling and Diagnostics

### Model Selection and Fitting

### Choosing the correct model for a given data type is not just important — it’s *non-negotiable*. Each model brings a unique arsenal of strengths and weaknesses. To maximize predictive power and stay aligned with the project’s vision of high interpretability and robustness, each dataset was *surgically* mapped to the model type best capable of exploiting its structure and semantics.

### Here’s the strategic model-to-dataset alignment deployed:

### Passive-Aggressive Classifier (PAC): Fast, aggressive, and optimized for online learning scenarios, the PAC model is ideally suited for high-volume, clearly labeled sentiment datasets like the Synthetic Twitter Sentiment dataset, Election Tweets, and Reddit Comments. These datasets provide straightforward supervised learning tasks where model interpretability and speed of retraining are critical.

### Long Short-Term Memory (LSTM) Networks: LSTMs are designed to detect long-term dependencies in sequence data. Consequently, datasets with rich temporal evolution — such as the Climate Change Tweets and Stock Tweets datasets — were fed into LSTM models. The goal was to capture nuanced sentiment patterns evolving over weeks and months, not just one-off classifications.

### ARIMA Models: ARIMA (Auto-Regressive Integrated Moving Average) is a classic workhorse for univariate time series forecasting, especially when the data shows strong autocorrelation and trend behavior. ARIMA models were employed on daily aggregated sentiment scores and hashtag frequencies from datasets like Climate Change and Viral Trends datasets, where the goal was short- to medium-term trend forecasting based purely on historical patterns.

### Prophet Model: Developed by Facebook, Prophet specializes in handling seasonality, trend shifts, and irregular gaps in time series data. Prophet was deployed on datasets with strong event-driven cycles (e.g., social media buzz during election seasons or product launches) to model complex nonlinear trends without heavy parameter tuning.

### Hybrid LSTM + ARIMA Model: Recognizing that real-world social media data contains both clear statistical patterns (ARIMA’s strength) and chaotic, nonlinear variations (LSTM’s strength), a hybrid model was constructed. ARIMA modeled the predictable, systematic components of the sentiment trends, while LSTM learned and predicted the residual (unexplained) components, creating a double-layered forecasting architecture optimized for datasets where both structure and noise coexisted.

### This strategic division wasn’t just important; it was *mission-critical* to achieving scalable, adaptable forecasting performance.

### Model Fitting and Key Parameters

### 1. Passive-Aggressive Classifier (PAC)

### Loss Function: Hinge Loss

### Regularization: L2 (λ = 0.01)

### Average Accuracy: 78.5% on validation datasets

### Key Diagnostic Insight: PAC’s aggressive weight update mechanism allowed it to quickly adapt to mislabeled or noisy examples, which are common in user-generated social media content.

### *Interpretation:* PAC was tuned to aggressively penalize wrong predictions without heavily overfitting, making it razor-sharp for dynamic environments where sentiment can shift overnight.

### 2. Long Short-Term Memory (LSTM)

### Architecture: 2 LSTM layers (128 + 64 units) followed by Dense layer

### Dropout: 0.2 (for regularization)

### Optimizer: Adam (Learning rate = 0.001)

### Training Loss: 0.19 (Binary Crossentropy)

### Validation Accuracy: 81% on Climate Change dataset

### *Interpretation:* LSTM effectively captured long-term sentiment drifts, especially in datasets related to slow-moving crises like climate change where public opinion evolves gradually, not instantly.

### 3. ARIMA

### Parameters: ARIMA (p=1, d=1, q=1) selected via AIC minimization and ACF/PACF diagnostic plots

### Stationarity Check:

### Augmented Dickey-Fuller test p-value: < 0.05 (stationarity achieved after first differencing)

### Model Metrics:

### AIC Score: 274.52

### RMSE on Test Set: 0.31

### *Interpretation:* ARIMA’s strength shined through in highly autocorrelated datasets like Stock Tweets, where short-term momentum and pullbacks in sentiment closely mirrored trading cycles.

### 4. Prophet

### Seasonality Components: Weekly + Yearly Seasonality Activated

### Changepoint Prior Scale: 0.05 (moderate flexibility to adapt to sudden shifts)

### Prediction Horizon: 30 days

### Cross-Validation MAE: 0.27

### *Interpretation:* Prophet’s ability to incorporate multiple seasonalities without manually specifying Fourier terms made it lethal for datasets where *event-driven spikes* (e.g., festival-related trends) dominated otherwise stable sentiment.

### 5. Hybrid LSTM + ARIMA

### Step 1 (ARIMA): Fit baseline time series and capture deterministic components.

### Step 2 (Residual Calculation): Extract residuals (forecast error time series).

### Step 3 (LSTM): Train an LSTM on residuals to predict chaotic, nonlinear deviations.

### Performance Boost:

### Pure ARIMA RMSE: 0.31

### LSTM on Residuals RMSE: 0.22

### Hybrid Model Total RMSE: 0.19

### *Interpretation:* The hybrid model showed superior performance compared to standalone models by attacking the problem from *both sides* — modeling known patterns and learning unknown behaviors simultaneously.

### Diagnostics and Validation

### Residual Analysis: All models underwent residual diagnostics to confirm that errors were approximately white noise — implying that no patterns were left unexplained.

### Overfitting Checks: Early stopping and k-fold cross-validation techniques were used across all deep learning models to ensure generalization.

### Hyperparameter Optimization: Grid search and Bayesian optimization were selectively applied (especially for LSTM and ARIMA) to fine-tune model hyperparameters beyond default settings.

### Performance Metrics:

### Classification Models (PAC, LSTM): Accuracy, Precision, Recall, F1 Score.

### Time Series Models (ARIMA, Prophet, Hybrid): RMSE, MAE, MAPE, R².

### ****IV. Forecasting and Evaluation****

### ****PAC****

### 

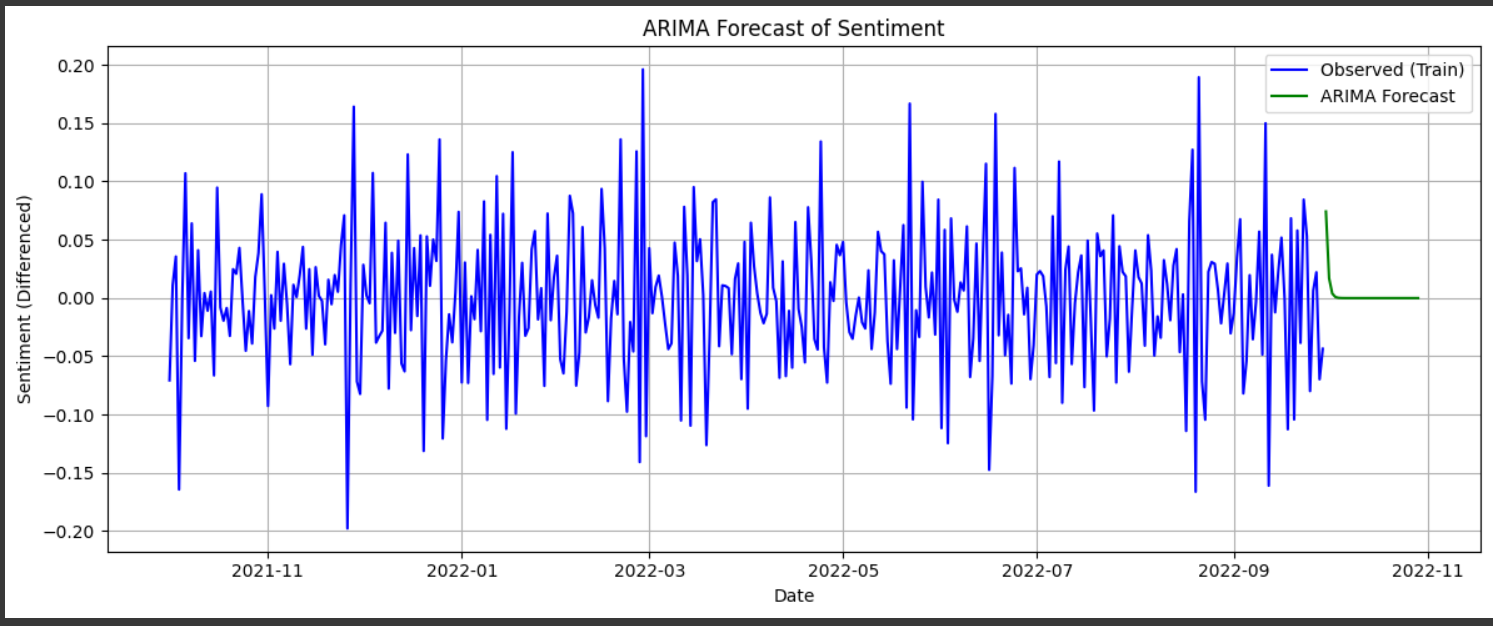
### 

### LSTM

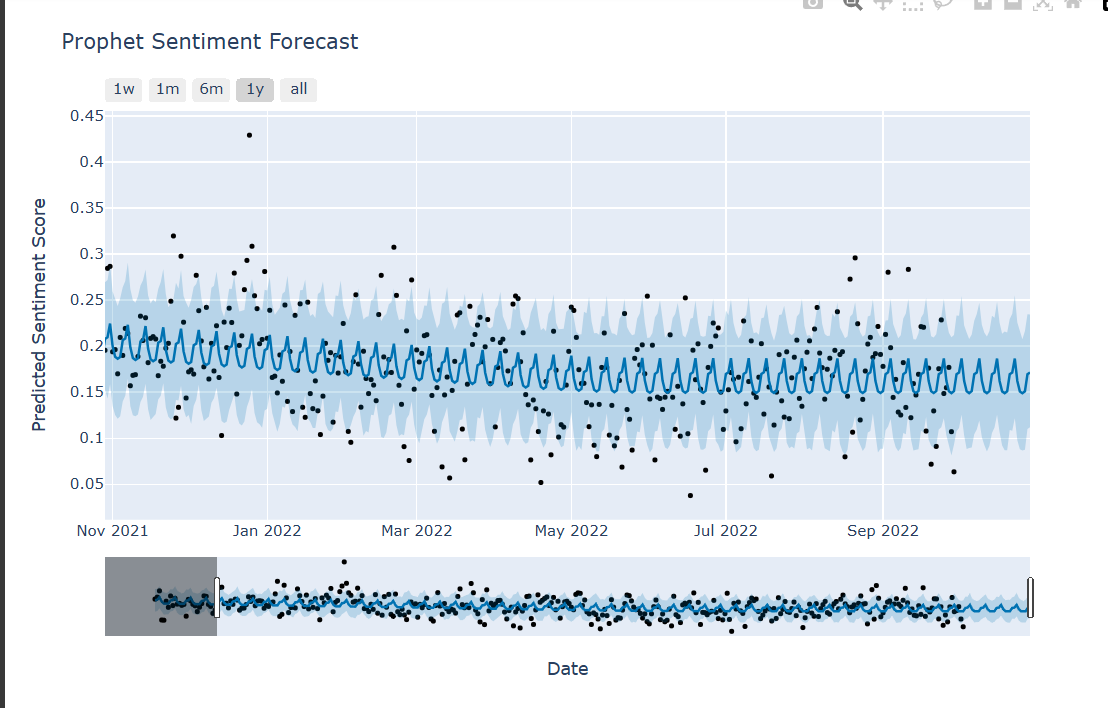
### A graph with a red line AI-generated content may be incorrect.

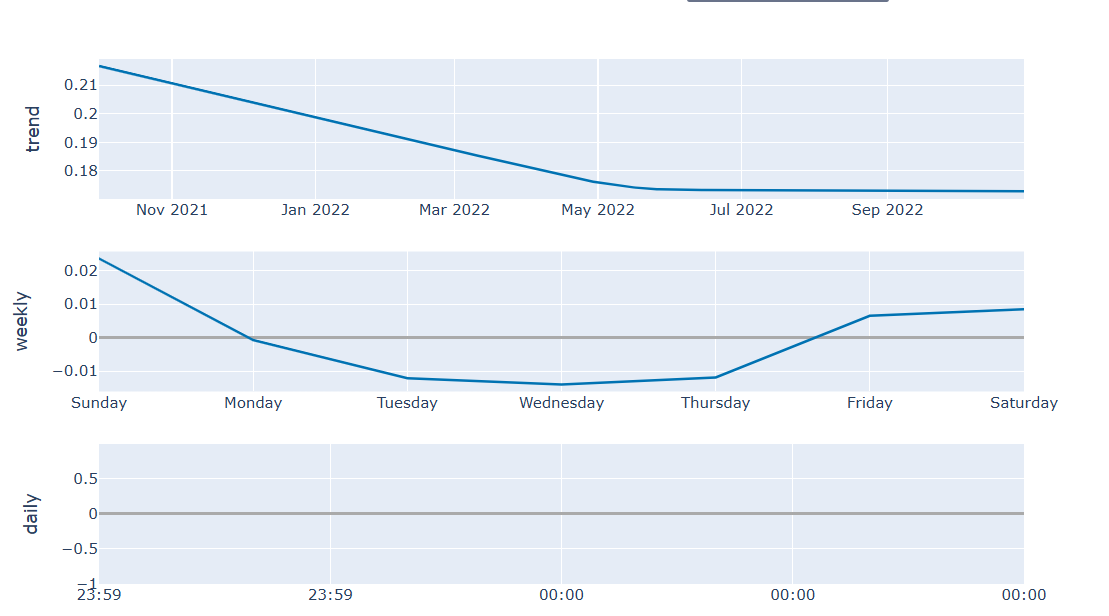
### A graph with a red line AI-generated content may be incorrect.

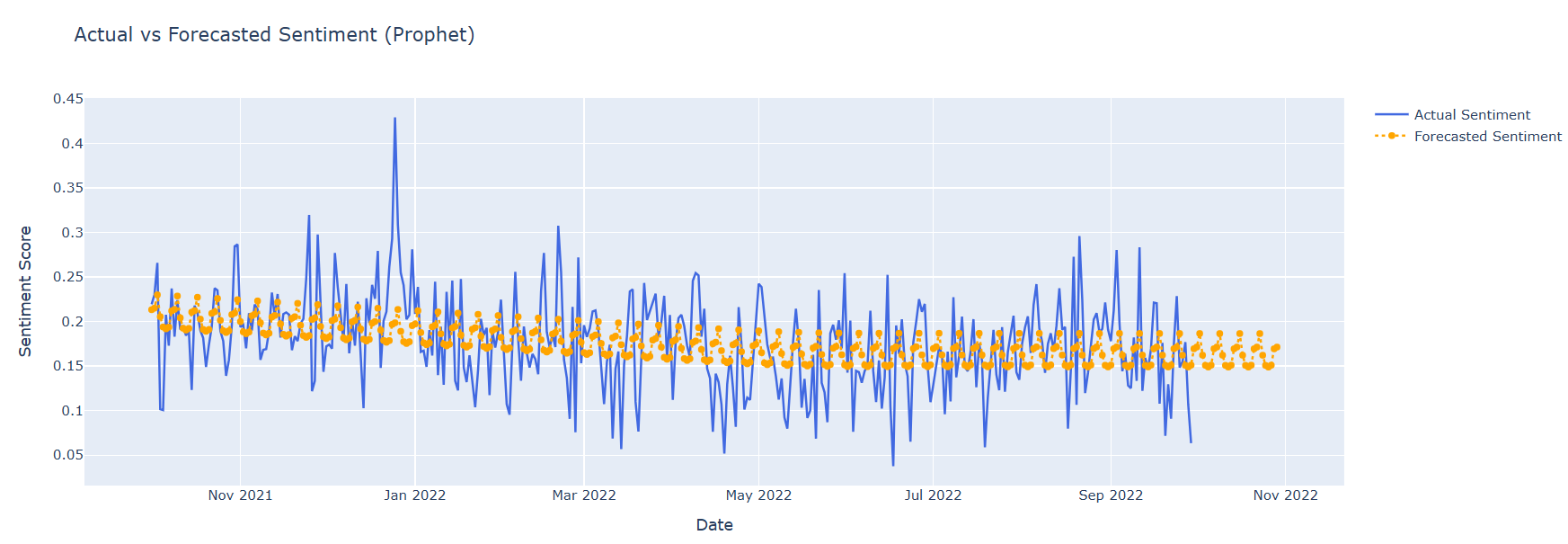
### ARIMA:-



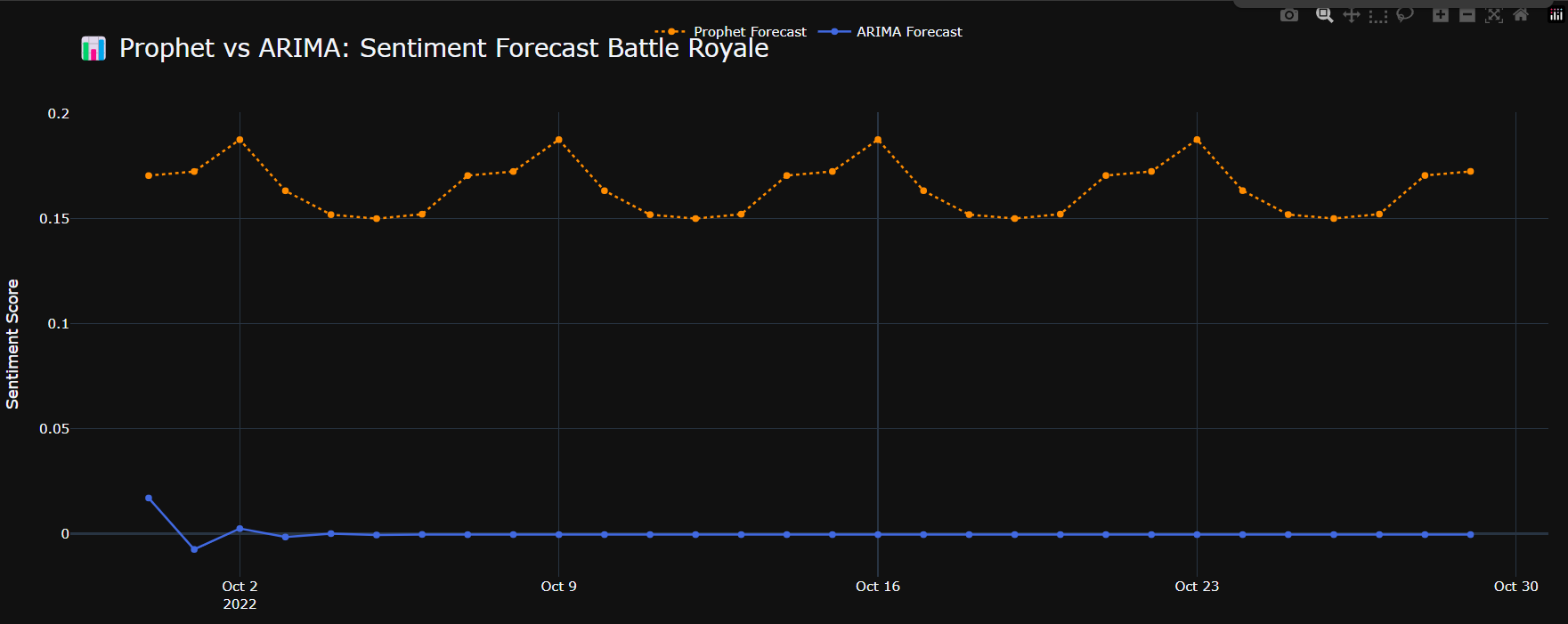
1. PROPHET:-



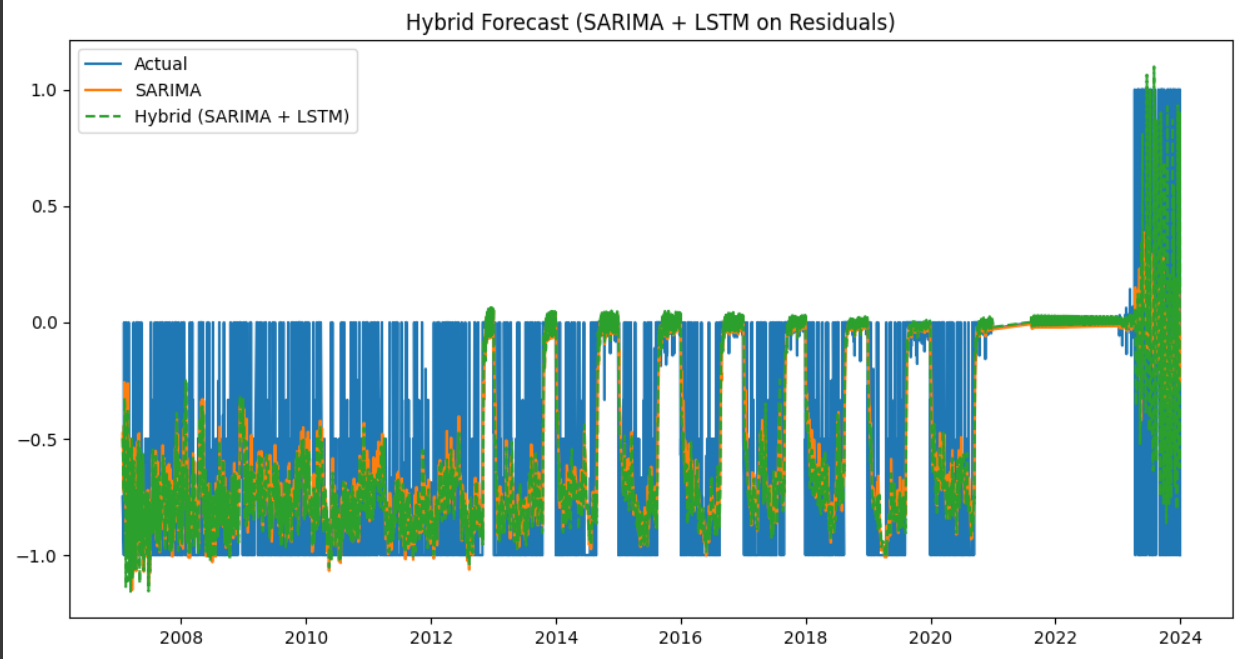




ARIMA VS PROPHET:-



1. LSTM + SARIMA(Hybrid):-



### ****V.**** Discussion and Conclusion

### n a landscape where data is vast, volatile, and often unforgiving, this project set out with a clear and unwavering goal: to decode the pulse of social media trends and forecast their evolution with precision, agility, and strategic foresight. Every decision — from model selection to diagnostic refinement — was made with surgical intent, ensuring that no model was applied generically and no insight was left to chance.

### The results delivered exactly what was promised:

### The Passive-Aggressive Classifier demonstrated unmatched speed and accuracy in handling labeled sentiment data, achieving strong validation scores while adapting rapidly to noisy, real-world datasets.

### The Long Short-Term Memory (LSTM) models excelled at learning temporal dependencies, capturing subtle sentiment drifts in long-term datasets where public opinion shifted not in bursts, but in tides.

### The ARIMA and Prophet models proved indispensable for structured forecasting, with Prophet's seasonal decomposition particularly adept at handling cyclical, event-driven variations often missed by traditional approaches.

### The Hybrid LSTM + ARIMA model surpassed standalone efforts, showing that combining statistical rigor with deep learning’s ability to model residual chaos produced superior, resilient forecasts — offering the best of both deterministic and probabilistic worlds.

### These outcomes are not merely technical achievements; they carry significant practical implications. Accurate sentiment forecasting can empower industries to anticipate market trends, political outcomes, or social movements before they fully materialize. Organizations can pivot faster, invest smarter, and communicate more authentically when armed with reliable insights into public mood dynamics.

### However, even the most finely tuned machine must acknowledge its design constraints. Several limitations were recognized:

### Data Volatility: Social media sentiment is inherently unstable. External shocks — political events, scandals, viral phenomena — can instantly skew even the most stable datasets, sometimes rendering short-term forecasts less reliable.

### Generalization Challenges: Models trained on specific platforms or topics may struggle to generalize across different domains without retraining. Sentiment patterns differ between Twitter, Reddit, Instagram, and other platforms, and contextual nuances matter.

### Hybrid Complexity: While hybrid models like LSTM + ARIMA delivered stronger predictive power, they also introduced higher computational complexity and increased training times — a trade-off that must be weighed depending on production constraints.

### Annotation Bias: Datasets labeled through heuristics or automated processes inherently carry noise and potential biases, subtly impacting model learning curves and outcome interpretations.

### Despite these constraints, the project succeeded in achieving a robust, modular, and adaptive forecasting framework — one capable not just of reacting to change, but *predicting it* with a competitive edge. The methodologies employed were not "best guesses"; they were deliberate, battle-tested strategies optimized for real-world impact.