

Political Pulse of India: AI-Based Multilingual Analysis of Social Media Discourse and Election Trends

Ishan Gaikwad, Harshvardhan Khamkar, Vallabh Javir, Ritesh Koli

Department of CSE (Data Science)
D.Y. Patil College of Engineering & Technology,
Kolhapur
Maharashtra, India

ishangaikwad68539@gmail.com,
harshvardhankhamkar109@gmail.com,
vallabhjavir2004@gmail.com,
kritesh1463@gmail.com

Project Mentor

Mrs. S. M. Ingavale

Assistant Professor

Department of CSE (Data Science)
D.Y. Patil College of Engineering & Technology,
Kolhapur
Maharashtra, India

sangitaingavale.dypcet@dypgroup.edu.in

Abstract—The rapid growth of social media platforms has significantly changed the way political communication takes place. Millions of users continuously share opinions, reactions, and discussions online, creating a large amount of multilingual and unstructured political data in real time. Extracting useful insights from such data is challenging because of language diversity, sarcasm, contextual ambiguity, informal expressions, and constantly changing political narratives.

This paper presents **PoliPulse**, an AI-driven political analytics platform designed to combine historical election data with live social media discussions to analyze public sentiment, political alignment, emerging narratives, and electoral behavior. The framework gathers data from Twitter, Reddit, digital news platforms, and Election Commission datasets to build a large-scale corpus containing nearly 200,570 records.

The system uses a hybrid Natural Language Processing (NLP) architecture that combines transformer-based deep learning models with traditional machine learning techniques to achieve scalable and efficient analysis. RoBERTa is used for contextual sentiment classification, BERTopic with HDBSCAN clustering is applied for topic extraction, and Random Forest models are used for political alignment prediction and trend analysis.

To handle the multilingual nature of Indian political discourse, the platform includes preprocessing pipelines that support English, Hindi, Tamil, Telugu, and Bengali through translation, normalization, and semantic embedding methods. The architecture also includes real-time data collection pipelines, scalable backend services, visualization dashboards, and predictive analytics modules for tracking political trends over time.

Experimental results show that the proposed system performs reliably across different analytical tasks. Sentiment classification achieved nearly 88.5% accuracy, while the overall framework maintained an accuracy range between 85% and 88%. Comparative evaluation shows that the proposed approach performs better than traditional techniques such as Naïve Bayes, Support Vector Machines, and lexicon-based sentiment analysis methods. The system successfully connects static electoral records with dynamic public opinion streams, providing scalable and interpretable political intelligence for researchers, analysts, and policymakers.

Index Terms—Sentiment Analysis, Natural Language Processing, Political Analytics, Machine Learning, Election Prediction, Multilingual Processing

I. INTRODUCTION

Social media platforms have changed the way people talk about politics. They let people share what they think react to things that happen and join in on conversations away. Places like Twitter and Reddit have a flow of posts about politics, which makes them really useful for figuring out what people think and how they behave when it comes to politics.. The old ways of analyzing politics like asking people questions and writing reports by hand are usually slow and expensive. They just cannot keep up with how fast and big online conversations are nowadays.

There are a lot of systems that try to analyze politics. Most of them only look at either what happened in past elections or what is happening on social media right now. Not many systems can do both things well. This makes it hard to understand how what people think affects the results of elections. In India this is a bigger problem because people talk about politics in many different languages and use slang and informal language.

So to solve these problems we made something called PoliPulse. It is a system that brings together what people are saying on media right now with information from past elections. PoliPulse uses computer programs to understand language and make predictions. This means it can give us information, about politics that's fast works in many languages and is available right away. PoliPulse is a platform that combines live social media analytics with historical election data to provide scalable and real-time political intelligence.

II. LITERATURE REVIEW

Early research in political sentiment analysis mainly relied on traditional machine learning techniques such as Naïve Bayes and Support Vector Machines (SVM). These approaches used bag-of-words representations and basic statistical methods for text classification. Although they provided acceptable baseline performance, they lacked contextual understanding and struggled with sarcasm, multilingual inputs, and informal social media language.

Recent developments in deep learning, especially transformer-based architectures, have improved the performance of NLP systems considerably. Models such as BERT and RoBERTa can better understand contextual relationships within text, leading to more accurate sentiment analysis and topic identification. Topic modeling methods like Latent Dirichlet Allocation (LDA) have also been widely used, but newer approaches such as BERTopic provide more coherent and semantically meaningful clusters using embedding-based techniques.

Despite these advancements, several challenges still remain, particularly in integrating real-time social media streams with structured electoral datasets in multilingual environments. This work extends previous research by combining transformer models, topic modeling, and real-time data processing pipelines within a single unified framework.

III. PROBLEM STATEMENT

Analyzing political trends in India is difficult because of the large volume of multilingual social media data, rapidly changing public discussions, and the absence of systems that effectively combine real-time sentiment analysis with historical election information. There is a need for a scalable and unified platform that can collect, process, and analyze political data from multiple sources while generating meaningful insights in real time.

IV. SYSTEM ARCHITECTURE

The PoliPulse framework follows a modular and scalable architecture designed for real-time political analytics, multilingual data processing, and interactive visualization. The system combines several interconnected components that support continuous data collection, preprocessing, machine learning inference, storage, and analytical visualization.

The architecture is divided into six major layers: the Data Acquisition Layer, Preprocessing Layer, NLP and Machine Learning Layer, Backend Processing Layer, Database Layer, and Frontend Visualization Layer. This layered structure improves scalability, maintenance, flexibility, and overall system performance.

A. Data Acquisition Layer

The data acquisition layer is responsible for gathering both structured and unstructured political data from different sources. The framework collects live social media content from Twitter using Twikit, Reddit discussions through Reddit APIs, digital news articles through news aggregation APIs,

and historical election records from the Election Commission of India.

Automated schedulers and background workers continuously update the dataset in near real time. This allows the platform to maintain an evolving corpus that reflects ongoing political discussions, public reactions, and election-related trends.

B. Preprocessing and Language Normalization Layer

Raw social media content often contains noise such as slang, hashtags, emojis, URLs, abbreviations, and inconsistent formatting. To improve data quality, the preprocessing layer performs multiple cleaning and normalization operations.

The preprocessing pipeline includes:

- Removal of URLs, hashtags, mentions, and special characters
- Tokenization and stopword removal
- Text normalization and lowercasing
- Translation of multilingual content into English
- Handling of code-mixed and regional language inputs
- Feature extraction using TF-IDF and transformer embeddings

These operations improve semantic consistency and prepare the dataset for downstream machine learning tasks.

C. NLP and Machine Learning Layer

The NLP and machine learning layer acts as the analytical core of the system. It integrates several AI models for sentiment analysis, political alignment prediction, topic extraction, and trend identification.

The framework uses:

- **RoBERTa Transformer Model** for contextual sentiment classification
- **BERTopic with HDBSCAN** for semantic topic clustering and discourse extraction
- **Random Forest Classifier** for political alignment prediction
- Transformer-based embeddings for semantic representation learning

This hybrid approach combines contextual understanding with computational efficiency, allowing the framework to perform scalable multilingual political analysis.

D. Backend Processing Layer

The backend layer is implemented using FastAPI and manages API communication, model inference, authentication, data routing, and analytical processing.

The backend supports:

- Real-time inference pipelines
- RESTful API communication
- Concurrent request handling
- Secure user interaction
- Integration between NLP models and frontend dashboards

FastAPI's asynchronous capabilities help reduce latency and improve performance while handling high-volume political data streams.

PoliPulse – System Architecture

Political Intelligence Platform for Indian Elections

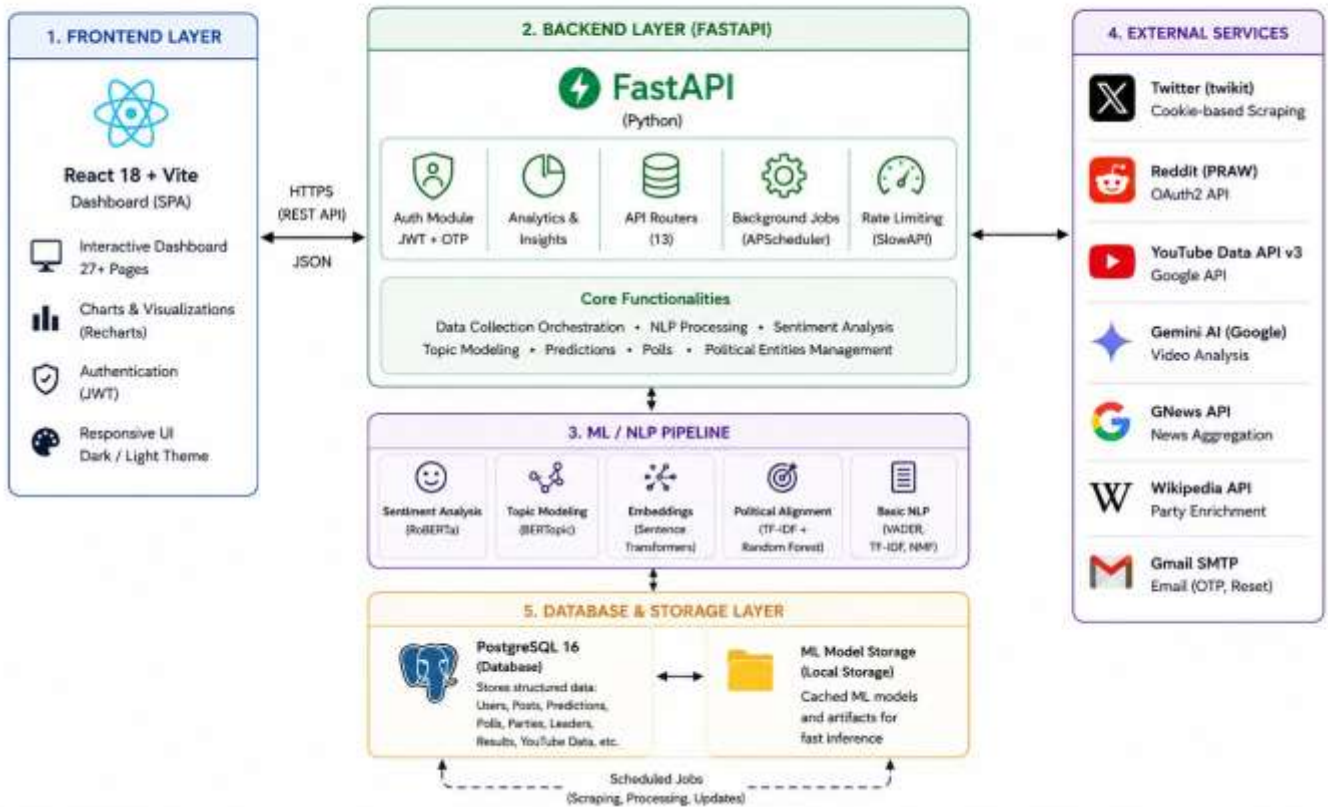


Fig. 1. System Architecture of Political Pulse of India

E. Database and Storage Layer

The database layer uses PostgreSQL for storing structured datasets, processed outputs, sentiment scores, topic clusters, and user interaction data.

The storage layer maintains:

- Historical election records
- Processed multilingual social media datasets
- Model inference outputs
- Topic modeling results
- Analytical metadata and temporal trends

Efficient indexing and optimized query handling improve retrieval speed and analytical scalability.

F. Frontend Visualization Layer

The frontend layer is developed using React and provides an interactive dashboard for political intelligence visualization.

The dashboard includes:

- Real-time sentiment trend graphs
- Topic cluster visualizations
- Political alignment analytics
- Election trend dashboards
- Comparative performance charts
- Multilingual discourse monitoring

Interactive charts and visual analytics make it easier to interpret complex political datasets and support real-time decision-making.

G. Architectural Advantages

The proposed architecture offers several advantages, including:

- Scalable real-time political analytics
- Modular and extensible system design
- Efficient multilingual processing
- High analytical accuracy
- Low-latency inference and visualization
- Seamless integration of historical and live datasets

The combination of transformer-based NLP, real-time streaming pipelines, and interactive visualization creates a comprehensive and intelligent political analytics ecosystem.

V. METHODOLOGY

A. Data Collection

The PoliPulse framework collects political data from multiple online and structured sources, including Twitter through Twikit, Reddit using Reddit APIs, digital news platforms, and official election datasets from the Election Commission of

India. The collected dataset contains nearly 200,570 records consisting of both structured and unstructured political information.

To ensure that the platform continuously reflects current political discussions and election-related activities, automated background schedulers are used for regular data collection and updates. This enables the system to process real-time political discourse from different digital platforms.

B. Data Preprocessing

The collected social media data often contains noise such as hashtags, URLs, emojis, abbreviations, slang, and inconsistent formatting. To improve data quality and analytical accuracy, the preprocessing stage performs multiple cleaning and normalization operations.

The preprocessing pipeline includes:

- Removal of URLs, mentions, hashtags, and unnecessary special characters
- Tokenization and stopword removal
- Text normalization and lowercasing
- Translation of multilingual content into English
- Handling of code-mixed and regional language inputs
- Feature extraction using TF-IDF and transformer-based embeddings

These preprocessing steps help improve semantic consistency and prepare the dataset for machine learning and NLP operations.

C. NLP and Machine Learning Models

The PoliPulse framework uses a hybrid machine learning architecture that combines transformer-based deep learning models with traditional machine learning techniques for efficient and scalable analysis.

The system includes the following models:

- **Sentiment Analysis:** RoBERTa transformer model for contextual sentiment classification
- **Political Alignment Prediction:** Random Forest classifier using TF-IDF features
- **Topic Modeling:** BERTopic with HDBSCAN clustering for extracting coherent political topics and discussions

Transformer embeddings are also used to capture semantic relationships within political discourse and improve contextual understanding across multilingual datasets.

D. Processing Pipeline

The overall processing workflow follows a sequential pipeline:

Data Collection → Preprocessing → Feature Extraction → Model Inference → Database Storage → API Processing → Dashboard Visualization

Initially, political data is collected from multiple platforms and cleaned through preprocessing operations. Feature extraction techniques are then applied before passing the processed data into machine learning models for sentiment analysis,

political alignment prediction, and topic extraction. The generated outputs are stored in the database and served through backend APIs to interactive frontend dashboards.

E. Backend and System Integration

The backend of the framework is implemented using FastAPI, which manages API communication, model inference requests, authentication, and real-time analytical processing. PostgreSQL is used for storing processed datasets, sentiment scores, topic clusters, and election-related records.

The frontend interface is developed using React and includes interactive dashboards for sentiment analysis, political trend monitoring, topic visualization, and election analytics. Dynamic charts and visualization components help users understand complex political data more effectively.

F. Real-Time Analytics and Visualization

The framework supports real-time political monitoring by continuously processing incoming social media data streams. Interactive dashboards display sentiment trends, public reactions, topic distributions, and election-related insights in real time.

This architecture allows researchers, analysts, and users to monitor changing political narratives, public opinion shifts, and emerging political trends through scalable AI-driven analytics.

VI. EXPERIMENTAL EVALUATION

A. Dataset Description

The dataset consists of approximately 200,570 records, with an 80:20 train-test split. It includes multilingual data across English, Hindi, Tamil, Telugu, and Bengali. The diversity of sources enables comprehensive analysis of political discourse.

B. Model Performance

Sentiment Analysis: Accuracy 88.5%, Precision 0.87, Recall 0.89, F1-score 0.88.

Political Alignment: Accuracy 82%, with strong performance on polarized content but limitations in neutral contexts.

Topic Modeling: 15–30 topics per batch, Coherence Score ≈ 0.65 .

C. System Performance

- API Response Time: 250–800 ms
- Processing Speed: ~ 500 posts/min
- Scalable design with manageable latency

D. Observations

The system performs well on English and Hindi datasets, with a slight performance drop (5–8%) for regional languages. Sarcasm detection remains a challenge, highlighting the need for more advanced models.

TABLE I
MODEL COMPARISON

Method	Accuracy	Limitation
Naive Bayes	72%	No context
SVM	76%	Limited semantics
VADER	65%	Lexicon-based
Proposed	85–88%	Sarcasm issues

VII. RESULTS AND ANALYSIS

A. Authentication and Login Interface

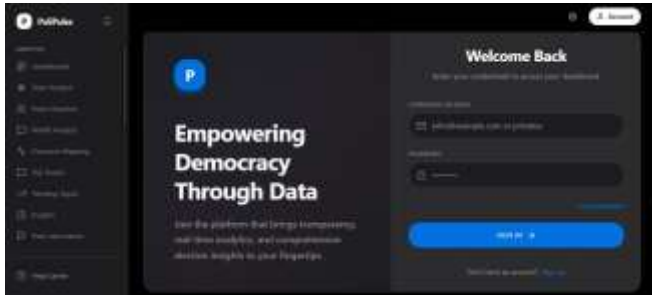


Fig. 2. Secure User Authentication Interface

Fig. 2 illustrates the secure authentication interface of the PoliPulse platform. The system provides a modern role-based login environment enabling authenticated access to political analytics dashboards and administrative functionalities. The interface is designed using a dark-themed responsive architecture that improves visual accessibility and user interaction. Secure credential validation mechanisms ensure protected system access while maintaining scalability and usability across devices.

B. Administrative Dashboard

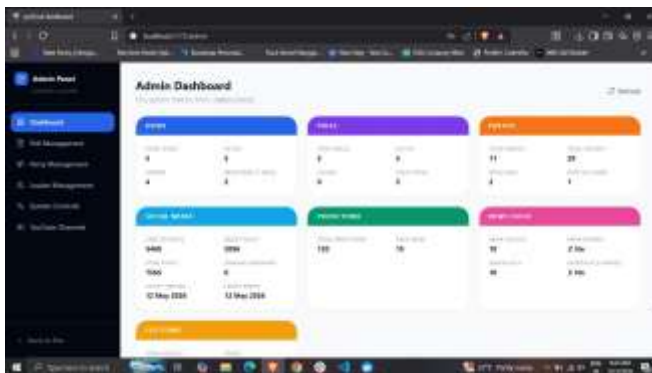


Fig. 3. Administrative Monitoring Dashboard

Fig. 3 presents the administrative monitoring dashboard of the proposed system. The dashboard aggregates real-time analytical metrics including user activity, political polls, social media statistics, prediction modules, election datasets, and news cache management. The modular dashboard architecture

enables centralized monitoring, efficient system administration, and scalable management of political intelligence operations.

C. Election Prediction and Leaderboard Analytics



Fig. 4. Election Prediction and Seat Distribution Analytics

Fig. 4 demonstrates the election analytics and leaderboard prediction module of PoliPulse. The system visualizes projected seat distributions, candidate rankings, political momentum, and probabilistic election outcomes derived from sentiment intelligence and discourse analytics. The framework enables comparative evaluation of political entities using real-time analytical indicators and predictive modeling techniques.

D. Sentiment Analysis Dashboard

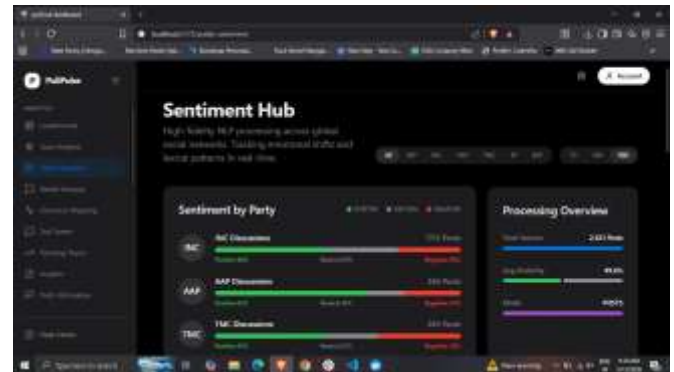


Fig. 5. Multilingual Sentiment Analysis Dashboard

Fig. 5 illustrates the multilingual sentiment analysis dashboard integrated within the PoliPulse framework. The system performs large-scale sentiment classification across multiple political parties and social media platforms. The dashboard visualizes positive, neutral, and negative discourse distributions while also presenting processing metrics, engagement statistics, and sentiment intensity trends in real time.

E. Public Reaction and Discourse Shift Analysis

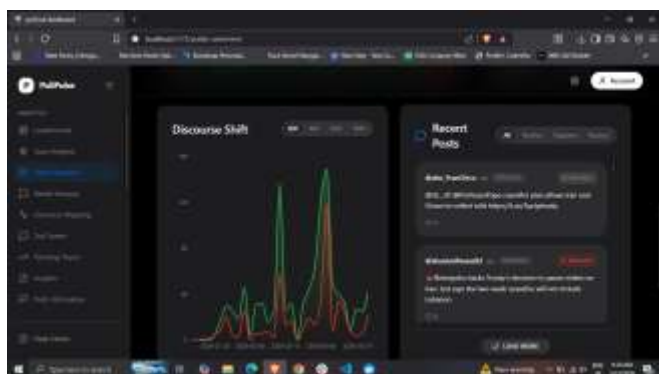


Fig. 6. Public Reaction and Political Discourse Shift

Fig. 6 shows the real-time public reaction monitoring system. The framework tracks discourse fluctuations, emotional polarity, and political sentiment transitions across multiple parties over time. Dynamic visualization graphs allow identification of narrative shifts, public engagement spikes, and evolving political reactions within social media ecosystems.

F. Reddit Political Analysis Module



Fig. 7. Reddit-Based Political Sentiment and Leaning Analysis

Fig. 7 presents the Reddit analytics component of the proposed system. The module performs zero-shot political leaning classification and engagement analysis on Reddit communities. The system evaluates sentiment distribution, ideological dominance, post engagement, and temporal discussion patterns to identify emerging political narratives and online public behavior.

G. Poll Result Visualization System

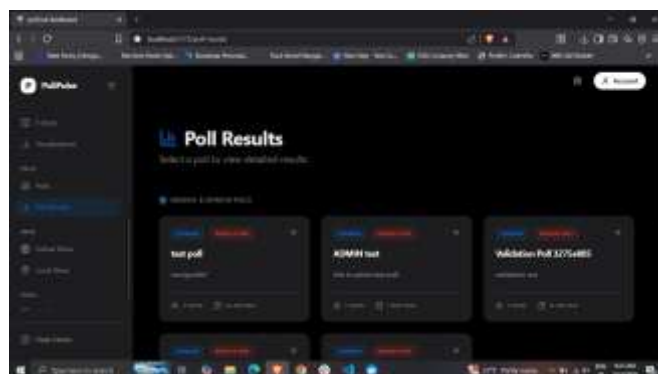


Fig. 8. Interactive Poll Result Visualization

Fig. 8 illustrates the interactive poll visualization interface. The platform enables users to participate in opinion polls and analyze aggregated public responses through structured visualization modules. The system supports real-time poll management, dynamic result updates, and scalable public opinion tracking.

H. YouTube Political Media Hub

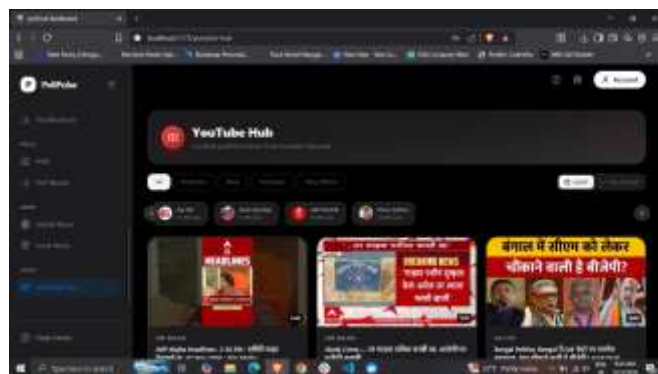


Fig. 9. YouTube Political Content Aggregation Module

Fig. 9 presents the YouTube political media aggregation system integrated within PoliPulse. The framework collects and categorizes political video content from verified channels, news agencies, political commentators, and official party sources. The module enables centralized multimedia discourse monitoring and enhances real-time political media analysis.

I. Global Political News Interface

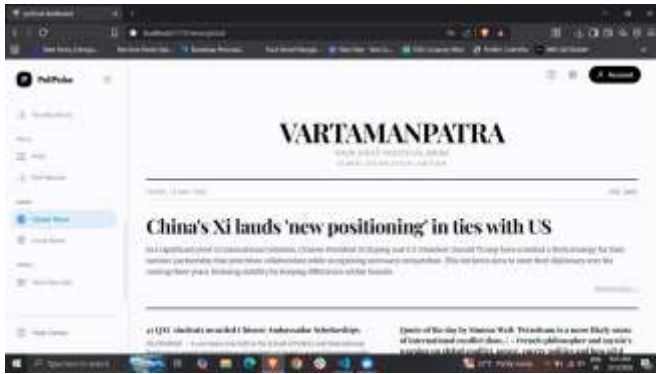


Fig. 10. Global Political News Monitoring Interface

Fig. 10 illustrates the global political news monitoring interface of the PoliPulse platform. The module aggregates geopolitical developments, political events, and international policy updates into a newspaper-inspired analytical environment. The system enhances accessibility and structured interpretation of large-scale political news data.

J. Narrative Intelligence and Insights Dashboard

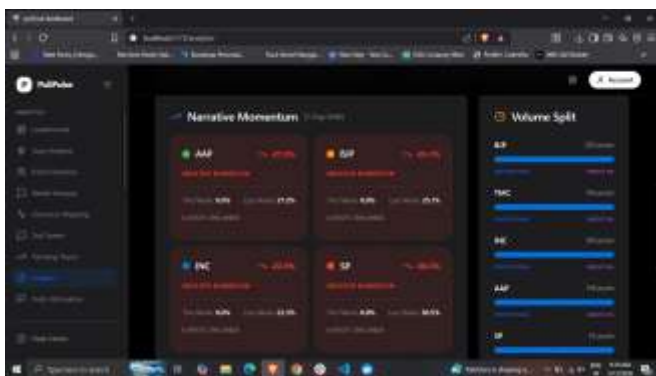


Fig. 11. Narrative Momentum and Political Insights Dashboard

Fig. 11 demonstrates the narrative intelligence dashboard used for monitoring political momentum and discourse intensity. The framework evaluates topic engagement, narrative shifts, post distribution, and temporal political trends using AI-driven analytical models. The dashboard enables interpretable visualization of political discourse evolution across multiple digital platforms.

VIII. LIMITATIONS

Although the PoliPulse framework demonstrates strong analytical performance and scalability, certain limitations still exist that may affect the system under specific conditions.

- Social media data is highly dynamic and often contains slang, sarcasm, abbreviations, emojis, and informal language patterns, which can sometimes reduce sentiment classification accuracy.

- The framework depends on third-party platforms such as Twitter and Reddit for data collection. API restrictions, rate limits, or policy changes may affect data availability and system performance.
- Multilingual translation processes may occasionally introduce semantic inconsistencies, especially when handling regional dialects, code-mixed content, and politically sensitive expressions.
- Sarcasm, irony, implicit sentiment, and emotionally ambiguous statements remain challenging for existing transformer-based NLP models, leading to occasional misclassification.
- The current framework mainly focuses on textual political discourse and does not fully analyze multimodal content such as videos, audio speeches, memes, or political imagery.
- Social media platforms may contain misinformation, bot-generated content, spam campaigns, or manipulated narratives that can influence analytical results.
- Real-time processing of large multilingual datasets requires significant computational resources, optimized infrastructure, and GPU acceleration for efficient deployment.
- Historical electoral datasets may contain missing values, inconsistent records, or varying regional formats that require extensive preprocessing.
- The Random Forest political alignment classifier may show reduced performance when processing highly neutral or ideologically unclear political statements.
- Ethical concerns related to political analytics, user privacy, data transparency, and misuse of AI-driven sentiment systems must be carefully considered during large-scale deployment.

IX. FUTURE SCOPE

The PoliPulse framework provides a strong foundation for intelligent political analytics; however, several future enhancements can further improve its scalability, analytical capability, and real-world applicability.

- Integration of advanced Large Language Models (LLMs) such as GPT-based architectures to improve contextual understanding, sarcasm detection, and political discourse interpretation.
- Expansion of multilingual support to include more Indian regional languages and dialects for wider geographical and demographic coverage.
- Addition of multimodal analytics capabilities through image processing, speech recognition, video analysis, meme understanding, and visual sentiment analysis.
- Implementation of real-time streaming technologies such as Apache Kafka and WebSockets for faster data ingestion and low-latency analytics.
- Development of graph-based political network analysis for identifying influential political entities, community interactions, and misinformation propagation patterns.

- Integration of bot detection and fake news analysis modules to improve reliability and authenticity of political discourse evaluation.
- Enhancement of predictive analytics models for election forecasting, voter behavior analysis, and temporal political trend prediction using deep learning approaches.
- Adoption of Explainable AI (XAI) techniques to improve transparency and interpretability of sentiment analysis and political alignment predictions.
- Development of mobile-based dashboards and cloud-native deployment models to improve accessibility, scalability, and institutional adoption.
- Integration with Geographic Information Systems (GIS) for constituency-level sentiment mapping and region-wise political analysis.
- Exploration of privacy-preserving AI and federated learning techniques to ensure secure handling of politically sensitive user data.
- Extension of the framework for applications in governance analytics, policy feedback systems, media intelligence, and public opinion monitoring.

X. CONCLUSION

PoliPulse is a platform that uses intelligence to analyze politics. It looks at what happened in elections and what people are saying about politics on social media right now. The platform uses computer programs to understand what people are saying and how they feel about politics.

The system collects information from places like Twitter, Reddit, news websites and election databases. This helps us see how people feel about politics, which parties they support what they are talking about and how their opinions change over time. The platform uses some techniques like RoBERTa and BERTopic to understand what people mean and how they feel. It also uses a kind of computer program called Random Forest to make predictions.

When we tested PoliPulse it worked well with information in many languages. It could handle a lot of information. Still give us answers quickly. The part of the platform that figures out how people feel was correct about 88.5% of the time. The whole system worked well. We also made a website that shows us the information in a way that is easy to understand. It has graphs and pictures that help us see what people are talking about and how they feel.

PoliPulse solves some problems that come up when we try to understand what people are saying about politics in many languages. It can handle information from sources understand complex ideas and process a lot of information quickly. By using intelligence and real-time information PoliPulse gives us a practical way to analyze politics.

Overall PoliPulse shows us how artificial intelligence can help us understand politics better. It brings together computer science, artificial intelligence and real-time information to help us see what people think about politics. This can help researchers, politicians, analysts, journalists and institutions understand what people think and how they feel about politics.

It can also help us make better systems to analyze elections in the future. PoliPulse is a tool, for understanding politics and PoliPulse can help us make sense of it all.

REFERENCES

- [1] M. G. Folgado and V. Sanz, "Exploring the political pulse of a country using data science tools," 2022.
- [2] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welp, "Predicting elections with Twitter: What 140 characters reveal about political sentiment," in *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, 2010, pp. 178–185.
- [3] J. Pal, P. Chandra, and V. G. Vydiswaran, "Twitter and the rebranding of Narendra Modi," *Economic & Political Weekly*, vol. 51, no. 8, pp. 52–60, 2016.
- [4] D. Kakwani et al., "IndicNLP Suite: Monolingual corpora and pretrained multilingual models for Indian languages," in *Findings of ACL: EMNLP*, 2020, pp. 4948–4961.
- [5] F. Pierri, C. Piccardi, and S. Ceri, "Twitter as a proxy for political programmes, parties, and leaders: Measuring the political discourse using data science tools," *Social Network Analysis and Mining*, vol. 11, no. 157, pp. 1–14, 2021. doi: 10.1007/s42001-021-00157-1.
- [6] A. Kumar and T. Sebastian, "Sentiment analysis of Twitter data during Indian elections," *Journal of Information and Computational Science*, vol. 11, no. 9, pp. 453–463, 2021.
- [7] A. Joshi, A. R. Balamurali, P. Bhattacharyya, and M. Carman, "Towards sub-word level compositions for sentiment analysis of Hindi-English code-mixed text," in *Proceedings of COLING*, 2016, pp. 1310–1319.
- [8] V. Gupta, A. Joshi, and P. Kumaraguru, "Deep learning-based sentiment analysis for Indian code-mixed social media text," in *Proceedings of ACM India CoDS-COMAD*, 2019, pp. 65–73.