Geo-Political Sentiment Analysis using Machine Learning

CH.GNANESWARI, G.DEVARAJ, G.LALITH KUMAR, K.POLURAJU

Students of Dept of CSD, Raghu Engineering College, Dakamarri (V), Bheemunipatnam, Visakhapatnam District,
Pin Code: 531162
Mr.V.Govinda Rao Assistant Professor, Dept of CSD, Raghu Engineering College, Dakamarri(V), Bheemunipatnam, Visakhapatnam Dist, Pin Code: 531162

ABSTRACT

This study presents the development of a recommendation-based automated system for geo-political sentiment analysis using machine learning techniques. The system processes large volumes of political text data, including tweets and news articles, by applying Natural Language Processing (NLP) methods such as tokenization, stop-word removal, and lemmatization to ensure clean and meaningful input. Feature extraction is carried out using techniques like TF-IDF, which convert textual data into numerical form suitable for model training. A supervised machine learning algorithm, specifically Support Vector Machine (SVM), is employed to classify sentiments into categories such as positive, negative, and neutral. The model is trained and optimized to achieve high accuracy and good generalization on unseen data. In addition, the system is integrated with a simple and user-friendly interface that allows users to input text and receive real-time sentiment predictions. Overall, the proposed approach provides an efficient and scalable solution for analyzing geo-political opinions, making it valuable for researchers, analysts, and decision-makers in understanding public sentiment trends.

Keywords-: Sentiment Analysis, Machine Learning, NLP, Geo-Political Data,Text Classification, Social Media Analysis

INTRODUCTION

This project builds a Geo-Political Sentiment Analysis system that classifies political text (like tweets and posts) into positive, negative, or neutral categories. It focuses on topics such as global politics, Indian political discussions, and the Russia-Ukraine conflict.

The system uses Natural Language Processing (NLP) to clean and prepare the text by removing unnecessary words and simplifying content. TF-IDF is used to convert text into numerical features, and machine learning models like Logistic Regression and Support Vector Machine (SVM) are used for classification. Among them, SVM performs better and is selected for the final system.

The model is deployed using a Streamlit application, where users can input any political text and get sentiment results. This project provides an easy way to understand public opinion, and it can be improved in the future by adding real-time data and support for multiple languages.

LITERATURE REVIEW

Sentiment Analysis in Social Media

Sentiment analysis helps understand people's opinions on social media platforms like Twitter. Many users share their

views on political topics online. However, the text is often short, informal, and contains slang. Political content may also include sarcasm and abbreviations.

TF-IDF Feature Engineering

TF-IDF is used to convert text into numerical form for machine learning. It highlights important words in a document compared to others. Words that appear frequently in one text get higher importance. This project uses both single words and word pairs. These features help capture better meaning in political text.

Classical ML Classifiers for Text Classification

Models like Logistic Regression and SVM are used for text classification. Logistic Regression is simple and easy to understand. SVM is more powerful and handles complex data better. It works well with high-dimensional text features. Hence, SVM is often preferred for better accuracy.

Geo-Political Text Analysis

Geo-political analysis studies opinions about global political events. Social media provides large amounts of such data. This data often reflects real public opinion. However, it can be biased and emotional. So, careful processing is required to analyze it properly.

Class Imbalance in Sentiment Datasets

Class imbalance happens when some sentiment classes have more data than others. Usually, neutral and negative data are more than positive. This can affect model performance. Techniques like resampling can help solve this issue. Metrics like F1-score are used for better evaluation.

Deployment of NLP Models

Deployment makes the model usable in real applications. Streamlit is used to create a simple web interface. Users can enter text and get sentiment predictions. It works well with Python libraries. This makes the system easy and user-friendly.

Summary

This project builds a sentiment analysis system using NLP techniques. TF-IDF is used for feature extraction. SVM is used as the main classification model. Challenges like noise and imbalance are handled. The system is simple, effective, and scalable.

PROPOSED METHOD

Proposed System

The system performs sentiment analysis on political tweets using machine learning techniques. It collects tweet data from datasets and cleans it by removing unwanted elements like URLs and special characters. The cleaned text is then converted into numerical form using TF-IDF. The dataset is split into training and testing parts for model building. Different models are trained, and the best one is selected for prediction.

Testing

Testing is done to check how well the system works. The dataset is divided into training and testing data. The model learns from the training data and is evaluated on the testing data. Algorithms like Naive Bayes, Decision Tree, and SVM are used. The system predicts whether the sentiment is positive, negative, or neutral and measures accuracy.

Deployment

Deployment makes the system available for real use. The trained model is connected to an application interface. Users

can enter tweet text as input. The system processes the text and predicts the sentiment. The output shows whether the sentiment is positive, negative, or neutral.

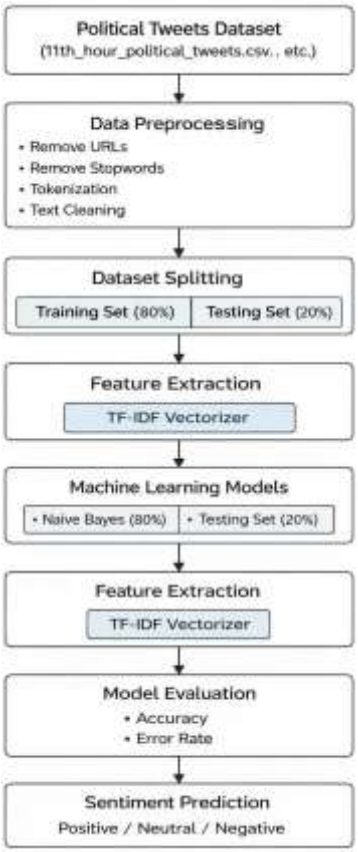

Fig1. System Architecture

This diagram represents the overall workflow of the political tweets sentiment analysis system. The process begins with collecting the tweet dataset. The data is then preprocessed to remove unwanted elements and clean the text. After that, the dataset is split into training and testing sets. TF-IDF is used for feature extraction, and machine learning models are applied to classify the sentiment. Finally, the system predicts whether the tweet sentiment is positive, negative, or neutral.

ACTIVITIES

Dataset Collection and Preprocessing

The dataset is collected from Kaggle using the Kaggle API and loaded into Pandas. It contains tweet text and sentiment labels like positive, negative, and neutral. Duplicate and missing values are removed, and all datasets are merged. The text is cleaned by removing URLs, special characters, and unwanted words. The final cleaned data is saved for further processing.

TF-IDF Feature Engineering

The cleaned dataset is divided into training (80%) and testing (20%) data. TF-IDF is used to convert text into numerical features. Both single words and word pairs are considered for better understanding. The vectorizer is trained on the training data and applied to the test data. The processed data and vectorizer are saved for future use.

Model Training

Machine learning models are trained using the processed data. Logistic Regression and SVM are used for

classification. Logistic Regression is simple and easy to use. SVM is more powerful and handles complex patterns well. Both models are trained and saved for prediction.

Model Evaluation

The trained models are tested using the test dataset. Performance is measured using accuracy, precision, recall, and F1-score. Confusion matrices are used to visualize results. The models are compared based on performance. SVM performs better and is selected as the final model.

Streamlit Application

The system is deployed using a Streamlit web application. The trained model and TF-IDF vectorizer are loaded at startup. Users can enter text into the input box. The system processes the text and predicts sentiment. The result is shown as positive, negative, or neutral with confidence.

Pipeline Execution

The project follows a step-by-step pipeline for execution. First, preprocessing is done to clean the data. Next, feature engineering converts text into numerical form. Then, models are trained and evaluated. Finally, the application is deployed using Streamlit. Each step depends on the previous one for proper execution.

Error Handling and Validation

The system includes checks to ensure proper input and processing. It verifies that required columns are present in the dataset. It also checks for valid sentiment labels. The model ensures correct data dimensions before training. The application handles empty input by showing a warning message.

Results

The sentiment analysis system was successfully implemented to analyze political tweets. The dataset was processed using text preprocessing and TF-IDF feature extraction. Machine learning algorithms such as Naive Bayes, Decision Tree, SVM, Random Forest, and CNN were applied to classify tweet sentiment. The models were trained and tested using the dataset. The experimental results showed good performance with high accuracy in predicting the sentiment of tweets. The system can correctly classify tweets into positive, negative, and neutral categories.

Positive sentiment

Negative Sentiment

Neutral Sentiment Conclusion

This project has successfully developed and deployed a complete Geo-Political Sentiment Analysis system that classifies political text into positive, negative, and neutral sentiment categories using classical NLP and machine learning techniques. The system integrates multi-source datasets covering global political commentary, Indian political discourse, and Russia-Ukraine conflict text into a unified preprocessing and training pipeline. The SVM classifier, trained on TF-IDF unigram and bigram features, achieves 69% accuracy and 0.57 macro F1-score on the held-out test set, outperforming the Logistic Regression baseline across all evaluation metrics. The Streamlit-based deployment provides an accessible, interactive interface for real-time sentiment prediction on user- submitted geo-political text.

The project demonstrates that classical ML approaches remain competitive and interpretable baselines for domain-specific NLP tasks where training data is limited and deployment resources are constrained. The modular pipeline architecture ensures reproducibility and facilitates future improvements at any stage of the workflow.

Future Enhancements

Several directions are identified for future enhancement. (1) Transformer-Based Models: replacing the TF-IDF + SVM pipeline with a fine-tuned BERT or RoBERTa model could substantially improve classification performance, particularly for ironic and contextually ambiguous political text. (2) Real- Time Data Ingestion: integrating the Twitter API v2 or the Mastodon API would enable real-time sentiment monitoring of ongoing geo-political events without manual dataset download. (3) Multilingual Support: extending the preprocessing pipeline to handle Arabic, Russian, and Hindi political text would broaden the system's applicability to non-English geo-political discourse. (4)

Topic Modelling: combining sentiment analysis with LDA-based topic modelling would allow users to explore which geo-political topics drive positive versus negative sentiment. (5) Named Entity Recognition: adding NER to identify the political entities (countries, leaders, organisations) mentioned in each text would enable entity-level sentiment aggregation. (6) Cloud Deployment: containerising the Streamlit application with Docker and deploying it to a cloud platform (AWS, GCP, or Azure) would make the tool publicly accessible and scalable to multiple concurrent users.

(7) Dashboard Analytics: building a multi-tab Streamlit dashboard with time-series sentiment trend charts would

enable longitudinal analysis of public sentiment evolution around geo-political events.

REFERENCES

Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2(1-2), 1-135.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing & Management, 24(5), 513-523.

Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. European Conference on Machine Learning (ECML), 137-142.

O'Connor, B., Balasubramanyan, R., Routledge, B. R., & Smith, N. A. (2010). From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. ICWSM, 11(122-129), 1-2.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321-357.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297. Manning, C. D., Raghavan, P., & Schutze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.

Bird, S., Klein, E., & Loper, E. (2009). Natural Language Processing with Python. O'Reilly Media. (NLTK Library)

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

Streamlit Inc. (2023). Streamlit Documentation — Build and deploy data applications. Available at: https://docs.streamlit.io/

Kaggle Inc. (2023). Kaggle API Documentation. Available at: https://www.kaggle.com/docs/api Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL-HLT, 4171-4186.

Liu, B. (2012). Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, 5(1), 1-167.

Barbosa, L., & Feng, J. (2010). Robust Sentiment Detection on Twitter from Biased and Noisy Data. COLING 2010: Posters, 36-44.

Giachanou, A., & Crestani, F. (2016). Like It or Not: A Survey of Twitter Sentiment Analysis Methods. ACM Computing Surveys (CSUR), 49(2), 1-41.

Nguyen, D. Q., & Nguyen, A. T. (2020). PhoBERT: Pre-trained language models for Vietnamese. Findings of EMNLP 2020.

Hutto, C. J., & Gilbert, E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. ICWSM, 8(1).

Wang, S., & Manning, C. D. (2012). Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. ACL 2012 Short Papers, 90-94.