

ADVANCED APPROACHES IN SENTIMENT ANALYSIS: FROM FEATURE SELECTION TO EMOTION DETECTION

Mrs. D. Kavitha¹, Mohammed Jawad², K.Nandhan Kumar³, B.NISHANTH⁴, V.PAVAN KUMAR⁵

¹Assistant Professor, Department of IT, TKR College of Engineering and Technology, Telangana, India

^{2,3,4,5}B.Tech Students, Department of IT, TKR College of Engineering and Technology, Telangana, India

Abstract - Sentiment analysis and emotion detection play a vital role in modern Natural Language Processing (NLP) by enabling organizations to understand user opinions, feedback, and emotional responses from textual data. Most traditional sentiment analysis approaches depend on labelled datasets and supervised machine learning models, which reduces flexibility and requires frequent retraining when applied to new domains. To overcome these limitations, this project proposes a web-based Sentiment and Emotion Classification System using the Gemini Large Language Model (LLM) for zero-shot text analysis.

The system accepts unlabelled CSV datasets and performs automated preprocessing such as text cleaning, normalization, and noise removal. Each text instance is analysed using Gemini to predict sentiment polarity (Positive, Negative, Neutral) and emotion categories (joy, sadness, anger, fear, surprise, disgust, neutral) without any prior training. The platform also supports real-time single-text classification through an interactive web interface.

To improve interpretability and insight, the system generates multiple visualizations including sentiment distribution charts, emotion frequency plots, confidence density graphs, word clouds, scatter plots, and aspect-based sentiment analysis. Additionally, a pseudo-SHAP based explanation mechanism is integrated for single-text analysis to highlight word-level importance and improve transparency. Developed using Django and modern visualization libraries, the proposed system provides a scalable, domain-independent, and user-friendly solution for sentiment and emotion analysis in real-world applications.

KEYWORDS- Sentiment Analysis, Emotion Detection, Gemini LLM, Zero-Shot Classification, Django, NLP, Visualization, Explainable AI, Pseudo-SHAP

1. INTRODUCTION

Electric Sentiment analysis is one of the most important research areas in Natural Language Processing (NLP), focused on identifying human opinions, attitudes, and emotional responses expressed in textual form. It is widely applied in customer feedback analysis, product review mining, social media monitoring, and business intelligence

systems. Early studies proved that machine learning methods can effectively classify sentiments from text using supervised approaches [1]. Later, opinion mining became a major research domain and was recognized as an essential technique for extracting valuable insights from large volumes of unstructured data [2].

1.1 Need for Sentiment and Emotion Analysis

In modern digital platforms, users continuously generate text data in the form of reviews, comments, tweets, and feedback. Analysing such large-scale text manually is impractical. Sentiment analysis helps in understanding overall public opinion, while emotion detection provides deeper insight by identifying emotional states such as joy, anger, sadness, fear, and surprise. Traditional systems mainly focus on polarity classification and often fail to capture fine-grained emotions, reducing the depth of analysis [2].

1.2 Limitations of Traditional Approaches

Most conventional sentiment analysis systems rely on rule-based techniques or supervised machine learning models such as Naive Bayes, SVM, and Logistic Regression [1], [2]. These approaches require large labelled datasets, extensive feature engineering, and frequent retraining when applied to new domains. Even modern transformer-based deep learning models such as BERT improve contextual understanding [3], but still require domain-specific fine-tuning and labelled training data, which increases computational and maintenance cost.

1.3 Emergence of Large Language Models

Recent advancements in Large Language Models (LLMs) have significantly improved NLP capabilities by enabling zero-shot and few-shot learning. Models such as GPT demonstrated that language models can perform classification tasks without explicit training on labelled datasets [4]. Similarly, Google's Gemini model provides highly capable LLM-based reasoning and text understanding, making it suitable for sentiment and emotion analysis in a domain-independent manner [5]. This shift enables systems to analyse new datasets without retraining and improves adaptability to different text styles and domains.

1.4 Importance of Explainability and Visualization

Along with prediction accuracy, interpretability is critical for sentiment and emotion classification systems. Explainable AI (XAI) methods such as SHAP provide a reliable way to interpret model predictions by highlighting the most influential input features [6]. This is essential for building user trust and transparency in AI systems. Additionally, visualization plays a major role in understanding large-scale sentiment trends. Tools such as Pandas and Matplotlib support effective visualization and statistical analysis of dataset-level results [9], [10]. Modern web frameworks such as Django help in building scalable, interactive, and user-friendly sentiment analysis platforms [11].

2. PROPOSED SYSTEM

The proposed system is an intelligent web-based Sentiment and Emotion Classification Platform designed to analyse textual data and generate meaningful insights in an automated and scalable manner. Unlike traditional sentiment analysis methods that depend on labelled datasets and supervised training, the proposed approach leverages the Gemini Large Language Model (LLM) to perform zero-shot sentiment and emotion prediction. This makes the system domain-independent and suitable for analysing datasets from multiple sources such as customer reviews, product feedback, social media posts, and survey responses without requiring retraining or manual annotation.

The system begins with a dataset preprocessing stage in which the user uploads an unlabelled CSV file containing text records. The system automatically identifies the relevant text column and performs preprocessing operations such as text normalization, lowercasing, removal of URLs, email addresses, special characters, and unnecessary symbols. This cleaning process ensures that noisy and unstructured real-world text is converted into a standardized format, improving the reliability of sentiment and emotion prediction.

After preprocessing, each cleaned text entry is processed through the Gemini LLM, which returns both sentiment polarity and emotion category for the given text. Sentiment is classified into Positive, Negative, or Neutral, while emotion is identified as one of the fine-grained categories such as joy, sadness, anger, fear, surprise, disgust, or neutral. Along with the predicted labels, the system also generates confidence values to represent the strength of each prediction. Since Gemini operates in a zero-shot manner, the system can generalize across different writing styles and domains without requiring traditional feature engineering or supervised learning pipelines.

To provide deeper understanding of the dataset, the proposed system includes a visualization and analytical engine that produces multiple graphical outputs. These visualizations include sentiment distribution charts, emotion

frequency graphs, confidence density plots, scatter plots comparing sentiment and emotion confidence, and word clouds for positive and negative texts. The system also supports aspect-based sentiment analysis by extracting common aspects such as camera, battery, display, performance, and price, and identifying the sentiment associated with each aspect. This allows users to understand which features or topics are discussed positively or negatively in the dataset.

In addition to dataset-level analysis, the proposed system supports real-time single-text classification through an interactive web interface. Users can enter any text input and instantly receive sentiment and emotion predictions along with confidence scores. To enhance transparency and trust, the system integrates a pseudo-SHAP inspired explanation mechanism for single-text predictions. This explanation method highlights the most influential words contributing to the predicted sentiment by measuring the change in prediction when individual words are removed from the input. The results are presented in the form of a word-importance visualization, providing an interpretable and explainable AI layer.

Overall, the proposed system delivers an end-to-end sentiment and emotion analysis solution by combining automated preprocessing, Gemini-based zero-shot classification, rich visualization, aspect-level insights, and explainable AI support. The platform is implemented using the Django framework, making it scalable, user-friendly, and suitable for real-world opinion mining and sentiment monitoring applications..

2.1 System Architecture

The architecture of the proposed Sentiment and Emotion Analysis System is designed as a web-based pipeline that connects the user interface, processing modules, AI model, and storage components in an integrated workflow. The user interacts with the system through a Django-based web application, where they can upload a CSV dataset or enter single text input for real-time analysis. Once the input is submitted, it is forwarded to the processing module, which performs text preprocessing operations such as cleaning, normalization, and noise removal to standardize the input data.

After preprocessing, the cleaned text is passed to the Sentiment and Emotion Analysis module, which communicates with the Gemini AI Model (Gemini API) to perform zero-shot classification. The Gemini model returns sentiment polarity and emotion category along with confidence values. The predicted results are then stored in the database or storage layer for further analysis and record management. Finally, the processed outputs are returned back to the web application, where results are displayed to the user in the form of predictions and visual analytics. This architecture ensures scalability, modularity, and domain-

independent sentiment and emotion classification through the integration of LLM-based intelligence.

contributing to the predicted sentiment, enabling explainable AI support.

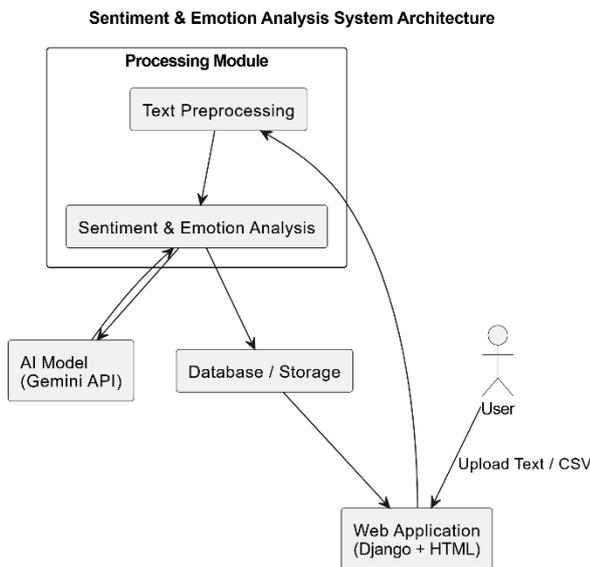


Fig -3: System Architecture

2.2 Module Description

The proposed system is divided into three major functional modules: Dataset Preprocessing, Visualization and Dataset Analysis, and Real-Time Single Text Classification. In the preprocessing module, the system accepts CSV datasets containing raw text and automatically performs cleaning operations such as removing URLs, email addresses, special characters, and unnecessary symbols. The processed text is normalized into a standardized format to improve the reliability of further analysis.

In the visualization and dataset analysis module, the system classifies each text instance into sentiment polarity and emotion category using the Gemini LLM. The predicted results are then aggregated and analysed statistically. Multiple visual outputs such as sentiment distribution charts, emotion frequency graphs, confidence density plots, scatter plots, and word clouds are generated to provide deeper insights into the dataset. Additionally, the system performs aspect-based sentiment analysis by extracting common aspects such as battery, camera, display, performance, and price, and identifying sentiment for each aspect.

The real-time single text classification module allows users to enter a single sentence or paragraph through the web interface. The system instantly predicts sentiment and emotion along with confidence scores. To improve interpretability, a pseudo-SHAP based explanation mechanism highlights the most influential words

2.3 Workflow of proposed system

At The workflow of the proposed system begins when the user uploads a CSV dataset or enters a text input through the Django web application. The system first preprocesses the input by performing text cleaning and normalization, ensuring that noisy real-world data is transformed into machine-readable format. After preprocessing, each cleaned text entry is forwarded to the Gemini LLM for zero-shot classification. The model generates sentiment polarity and emotion category predictions along with confidence scores. For dataset-level analysis, the predictions are aggregated and stored, and the system generates multiple visualizations such as sentiment distribution charts, emotion frequency plots, confidence density graphs, scatter plots, and word clouds. Aspect-based sentiment analysis is also performed to identify sentiment trends for specific aspects discussed in the dataset. For single-text analysis, the system displays real-time sentiment and emotion results and generates a pseudo-SHAP word-importance explanation plot. Finally, all outputs are displayed through the web interface, enabling users to obtain both predictions and interpretable insights in an efficient and user-friendly manner.

3. IMPLEMENTATION DETAILS

The proposed Sentiment and Emotion Classification System is implemented as a web-based application using the Django framework. The complete system integrates text preprocessing, Gemini LLM-based zero-shot classification, visualization generation, and explainable AI support. The implementation focuses on modular design so that each stage of processing can be executed independently while maintaining smooth integration across the platform.

3.1 Web Application Development Using Django

The web application is developed using Django, which provides a structured and scalable framework for handling user requests, file uploads, session management, and dynamic result rendering. The system provides separate interfaces for dataset preprocessing, visualization-based dataset analysis, and real-time single text classification. Django views are used to manage the entire pipeline, including reading CSV files, storing processed results, generating charts, and rendering outputs through HTML templates. The user interface enables easy interaction for both technical and non-technical users by providing simple forms for uploading datasets and entering text for real-time analysis.

3.2 Gemini LLM Integration for Zero-Shot Classification

The core classification functionality is implemented using the Gemini API through the Google GenAI client. For every cleaned text input, a structured prompt is generated and sent to the Gemini model to obtain sentiment polarity and emotion category predictions. The system enforces output formatting by requesting responses strictly in JSON format, ensuring reliable parsing of predictions. The model generates sentiment labels as Positive, Negative, or Neutral, and emotion labels as joy, sadness, anger, fear, surprise, disgust, or neutral. Confidence scores are also extracted and stored for both sentiment and emotion predictions. To improve robustness, a fallback rule-based mechanism is implemented to handle cases where API access is unavailable or responses fail, ensuring the system continues to function even under limited connectivity.

3.3 Visualization Generation and Explainable AI Implementation

To provide analytical insights, the system generates multiple visualizations using Pandas, Matplotlib, Seaborn, and WordCloud libraries. After processing a dataset, sentiment and emotion predictions are aggregated to create sentiment distribution charts and emotion frequency plots. Confidence density graphs and scatter plots are generated to represent prediction confidence behavior. Positive and negative word clouds are created to highlight dominant keywords associated with each sentiment class. Additionally, aspect-based sentiment analysis is implemented by extracting common aspects from text and mapping them to sentiment categories. For single-text classification, a pseudo-SHAP explanation mechanism is implemented by evaluating word importance through word-removal impact analysis. The most influential words are plotted as a word-importance bar chart, improving transparency and interpretability of predictions. All generated charts and annotated CSV outputs are stored in the media directory and dynamically displayed in the results page through the Django interface.

4. RESULTS AND PERFORMANCE ANALYSIS

The proposed system was tested using sample text datasets containing user reviews and feedback collected in CSV format. After uploading the dataset, the system successfully performed preprocessing, removed noise such as special characters and unwanted symbols, and generated cleaned text suitable for classification. Each cleaned record was then analyzed using the Gemini LLM, which produced sentiment polarity and emotion category predictions along with confidence values. The dataset-level results were visualized through multiple charts to provide an intuitive understanding of sentiment and emotion patterns.

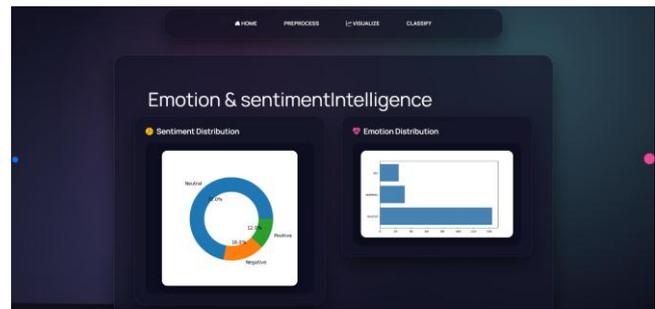


Fig 4.1: Sentiment and Emotion Distribution Visualization

Figure 4.1 shows the overall sentiment distribution and emotion distribution obtained from the uploaded dataset. The sentiment distribution chart provides a clear summary of the percentage of Positive, Negative, and Neutral opinions present in the dataset. Similarly, the emotion distribution graph highlights the frequency of emotions such as joy, sadness, and neutral, enabling deeper understanding beyond polarity classification. These results demonstrate that the proposed system can effectively perform large-scale sentiment and emotion analysis without requiring labelled training.

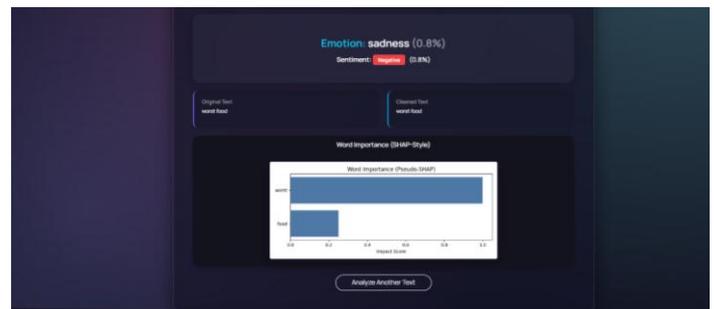


Figure 4.2: Real-Time Single Text Sentiment and Emotion Prediction with Pseudo-SHAP Explanation

In addition to dataset analysis, the system also supports real-time classification for single text input. Figure 4.2 shows the output of the real-time classification module, where the system predicts both sentiment and emotion for an entered text along with confidence scores. To improve transparency, the system generates a pseudo-SHAP word importance explanation plot that highlights the most influential words contributing to the final prediction. This feature enhances interpretability and helps users understand why the model produced a specific sentiment or emotion label. Overall, the results confirm that the proposed system delivers accurate, explainable, and visually interpretable sentiment and emotion analysis for both dataset-level and single-text scenarios.

5. CONCLUSIONS

This project presented a web-based Sentiment and Emotion Classification System that provides an end-to-end solution for analyzing textual data using the Gemini Large Language Model. Unlike traditional sentiment analysis methods that require labeled datasets and model training, the proposed system performs zero-shot sentiment and emotion prediction, making it scalable and domain-independent. The system successfully supports CSV dataset preprocessing, automated sentiment and emotion classification, and rich visualization outputs such as sentiment distribution charts, emotion frequency plots, confidence density graphs, word clouds, and aspect-based sentiment analysis. In addition, the real-time single text classification module with pseudo-SHAP based word importance explanation improves interpretability and enhances user trust in model predictions. Overall, the system demonstrates that LLM-powered sentiment and emotion analysis can effectively replace conventional supervised approaches and provide a flexible, explainable, and user-friendly platform for real-world opinion mining applications.

6. FUTURE WORK

In future, the proposed system can be extended to support multilingual sentiment and emotion analysis so that users can analyze text in multiple languages such as Hindi, Telugu, Tamil, and other regional languages. The emotion classification module can also be enhanced by adding more fine-grained emotional categories and supporting mixed-emotion detection for complex sentences. To improve performance on large datasets, the system can be optimized using batch processing, caching, and asynchronous task queues such as Celery. Future versions can integrate advanced explainability techniques such as true SHAP, LIME, or attention-based explanations for more accurate interpretation. Additionally, the platform can be expanded with live social media data extraction and real-time streaming dashboards, enabling continuous sentiment monitoring for brands, products, and public opinions.

ACKNOWLEDGEMENT

There are many individuals who have contributed directly or indirectly to the successful completion of our project, and we take this opportunity to express our sincere gratitude to all of them. We are extremely thankful and indebted to our project guide, Mrs. D. Kavitha, Assistant Professor, for her valuable guidance, continuous encouragement, and unwavering support throughout the development of this project. Her insightful suggestions and technical expertise played a vital role in shaping the outcome of our work.

We extend our heartfelt thanks to Dr. N. Satyanarayana, Head of the Department, for his constant motivation, support, and encouragement during the course of the project. We are also grateful to all the faculty members and

staff of the department for their cooperation and assistance whenever required.

Finally, we would like to express our sincere appreciation to our parents and friends for their constant support, motivation, and encouragement, which helped us successfully complete this project.

REFERENCES

- [1] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proceedings of the ACL Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.
- [2] B. Liu, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012.
- [3] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019.
- [4] T. B. Brown et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [5] Google Research, "Gemini: A family of highly capable multimodal models," *Google AI Technical Report*, 2024.
- [6] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [7] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media, 2009.
- [8] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [9] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.
- [10] W. McKinney, "Data structures for statistical computing in Python," in *Proceedings of the 9th Python in Science Conference*, 2010.
- [11] Django Software Foundation, "Django Documentation," 2024. [Online]. Available: <https://docs.djangoproject.com/>
- [12] F. Pedregosa and G. Varoquaux, "Explainable Artificial Intelligence (XAI): Concepts, applications, and challenges," *IEEE Intelligent Systems*, 2020.