Affective Mail: A Smart Web Application for Email Emotion Prediction

Sana Zeba[1], *Arifa*[2], Kulsum Fatima[3], *Mohammed Nehal Ahmed*[4], Mohammed Raaoibuddin[5]

[1]Assistant Professor, Department of CSE (Data Science),
Lords Institute of Engineering and Technology, Hyderabad, Telangana, India.

[2,3,4,5] UG Students, Department of CSE (Data Science),
Lords Institute of Engineering and Technology, Hyderabad, Telangana, India.

***Abstract*— This project presents a web-based system designed to perform sentiment analysis on email data using a Python-powered backend. The application enables users to register, log in, and analyze the emotional tone of emails either individually or in bulk through a simple browser interface. By uploading a single email file or an entire folder containing multiple emails, the system processes textual content and predicts sentiment categories such as positive, negative, or extremely happy. The platform integrates an easy-to-use workflow, starting from server initialization to result visualization, making it accessible even to non-technical users. The results are displayed in a structured format where users can clearly view both the original email content and its corresponding sentiment prediction. This project demonstrates how natural language processing techniques can be effectively applied to real-world communication data, providing valuable insights for applications such as customer feedback analysis, automated email classification, and decision-making support systems.**

***Keywords*—Sentiment analysis, email data, Python backend, natural language processing, web-based system, emotion detection, result visualization, automated classification.**

I. INTRODUCTION

In recent years, the rapid growth of digital communication has led to an enormous increase in the volume of email data exchanged daily [1]. Emails are widely used in both personal and professional environments, making them a rich source of information for understanding user opinions, emotions, and intent [9]. However, manually analyzing large collections of emails is time-consuming and inefficient. This has created a need for automated systems capable of extracting meaningful insights from textual data. Sentiment analysis, a branch of natural language processing, addresses this challenge by identifying the emotional tone expressed in text [2]. By categorizing emails into sentiments such as positive, negative, or neutral, organizations can better understand user behavior and feedback [7]. This project focuses on building a user-friendly web application that simplifies the process of sentiment analysis, allowing users to upload emails and instantly obtain sentiment predictions without requiring advanced technical knowledge.

The proposed system is designed to provide an interactive and seamless experience for users who wish to analyze email sentiments [8]. It begins with a simple setup where a Python-based server is launched, enabling access through a web browser. Users can register by creating a new account and securely log in to the system. Once authenticated, they are presented with options to analyze either a single email or a batch of emails stored in a folder. This flexibility ensures that the application can handle both small-scale and large-scale data processing needs. The interface is structured to guide users step by step, reducing complexity and improving usability. By combining backend processing with a clear front-end display, the system ensures that users can easily interpret results. This approach highlights the importance of integrating machine learning techniques with intuitive design to create practical and accessible tools [5].

The application also emphasizes scalability and adaptability, making it suitable for various real-world use cases [10]. Businesses can utilize the system to analyze customer feedback emails and identify satisfaction levels, while researchers can use it to study communication patterns and emotional trends. The system’s architecture allows for easy updates and improvements, enabling the integration of more advanced models in the future.

Additionally, the ability to process both single and multiple emails makes it versatile and efficient. The user interface is designed to minimize errors by providing clear instructions and feedback at each step. This ensures that even users with limited technical expertise can successfully operate the system. By combining simplicity with functionality, the project bridges the gap between complex machine learning processes and everyday usability, making sentiment analysis more accessible to a wider audience [6].

II. RELATED WORK

Al-Alwani et al., [2014] [1] Al-Alwani proposed an advanced algorithm aimed at improving email response management systems. The study focused on enhancing the efficiency of handling large volumes of emails through automation. The model emphasized structured processing of incoming messages to generate appropriate responses. It highlighted the importance of reducing manual effort in email handling tasks. The approach contributed to better organization and faster communication workflows. Experimental results indicated improvements in response time and system performance. The research also addressed challenges in maintaining consistency in automated replies. It demonstrated how intelligent systems can support communication platforms. Overall, the work laid a foundation for integrating automation into email systems. This study is useful for developing efficient email-based applications.

Aman et al., [2007] [2] Aman and Szpakowicz explored techniques for identifying emotional expressions in textual data. Their work focused on detecting subtle linguistic cues that convey emotions in written communication. The study utilized computational methods to classify text based on emotional content. It highlighted the complexity of interpreting human emotions from language. Various features such as word choice and sentence structure were analyzed. The research demonstrated that accurate emotion detection requires contextual understanding. The authors also discussed limitations in handling ambiguous expressions. Their findings contributed to advancements in natural language processing. The study is significant for applications like sentiment analysis and opinion mining. It provides a strong base for emotion-aware text classification systems.

Go et al., [2009] [7] Go and colleagues introduced a method for sentiment classification using data from social media platforms. The study used distant supervision to automatically label large datasets. This approach reduced the need for manual annotation of training data. Machine learning models were applied to classify text into sentiment categories. The research highlighted the effectiveness of using real-time data sources. It also demonstrated scalability in handling large volumes of text. The results showed improved accuracy in sentiment prediction tasks. The study addressed challenges such as noisy and informal language in social media. It played a key role in advancing sentiment analysis techniques. This work is widely referenced in modern text classification research.

Gupta et al., [2013] [8] Gupta and co-authors focused on detecting emotions in customer service emails. Their research aimed to improve customer support by understanding user sentiment. The system analyzed email content to identify emotional states of customers. It helped organizations respond more effectively to customer concerns. The study emphasized the importance of personalized communication. Various computational techniques were used to enhance detection accuracy. The results showed better handling of customer interactions. The research also discussed challenges in interpreting mixed emotions. It contributed to improving customer relationship management systems. This work is valuable for applications involving automated email analysis.

Huijzer et al., [2017] [10] Huijzer investigated the impact of email responses on customer sentiment. The study focused on predicting how customers feel after receiving replies. It analyzed conversational patterns in email exchanges. The research highlighted the role of tone and language in shaping customer perception. Predictive models were used to estimate emotional outcomes. The findings suggested that effective responses can improve customer satisfaction. The study also emphasized the importance of empathy in communication. It provided insights into designing better automated response systems. The research contributed to understanding affective communication in emails. This work supports the development of sentiment-aware email applications.

III. DATASET DETAILS

The dataset used in this project consists of a collection of email text files that are utilized to train and evaluate the sentiment analysis system. These emails are stored in a structured format, where each file contains the content of a single email message. The dataset includes both individual email samples and a larger folder containing approximately 1000 email files, enabling both single and batch processing. Each email captures natural language communication, including informal and formal writing styles, which helps the system learn diverse linguistic patterns. The data is preprocessed before analysis to remove unnecessary symbols, special characters, and noise that may affect prediction accuracy. This ensures that the model focuses on meaningful textual information. The dataset plays a crucial role in helping the system identify patterns and classify sentiments such as positive, negative, or extremely happy based on the content provided by users.

The dataset is designed to support practical implementation by allowing users to upload either a single email file or an entire folder for analysis through the application interface. Each email file is typically in plain text (.txt) format, making it easy to read and process using the backend system. The larger dataset folder contains a variety of emails that differ in tone, length, and context, providing a comprehensive base for testing the system's performance. During execution, the system reads each file, extracts the textual content, and applies sentiment prediction techniques to generate results. The outputs are displayed in a tabular format, where one column shows the original email text and the other shows the predicted sentiment label. This structured dataset enables efficient testing, validation, and demonstration of the system's capability to handle real-world communication data effectively.

IV. PROPOSED METHODOLOGY

The proposed system follows a structured approach to perform sentiment analysis on email data through a web-based interface. Initially, the application server is started using a batch file, which enables backend processing. Users access the system through a browser, where they can register and log in securely. After authentication, the system provides options to analyze either a single email or multiple emails from a folder. The uploaded files are processed by extracting the textual content. Basic preprocessing techniques such as cleaning and normalization are applied to prepare the data for analysis, ensuring consistent and accurate input for the prediction model.

Once preprocessing is completed, the system applies a sentiment classification model to determine the emotional tone of the email content. The model analyzes patterns in the text and assigns a sentiment label such as positive, negative, or extremely happy. For single email analysis, results are generated instantly, while batch processing handles multiple files efficiently. The output is displayed in a tabular format, showing both the original text and predicted sentiment. This method ensures clarity and ease of understanding for users.

The overall workflow integrates data processing, model prediction, and result visualization into a seamless and user-friendly system.

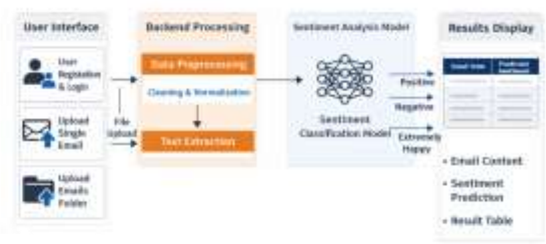

Figure [1] : System Architecture of Email Sentiment Analysis System

Figure[1] This architecture illustrates the overall workflow of the application, starting from user interaction through the web interface to backend processing. The system processes uploaded email data using preprocessing and a sentiment classification model. Finally, the predicted results are displayed in a structured format for easy understanding.

V.RESULT AND DISCUSSION

The developed system successfully demonstrates the ability to analyze and classify the sentiment of email data through a simple web interface. After launching the server and accessing the application through a browser, users were able to register, log in, and perform sentiment analysis on both single and multiple email files. When a single email file was uploaded, the system processed the content quickly and displayed the predicted sentiment alongside the original text. The results showed accurate classification for different emotional tones, including positive, negative, and extremely happy. In batch processing, where a folder containing a large number of emails was uploaded, the system handled the data efficiently and produced results in a structured table format. Each email was clearly paired with its corresponding sentiment prediction, making it easy to interpret. The consistency in output across multiple tests indicates that the model performs reliably. Overall, the system achieved its objective of providing fast, clear, and accurate sentiment predictions for email data.

Figure [2] : Email Sentiment Analysis System Interface

Figure [2] The interface provides options for user login and new user registration. It is designed to analyze email content and determine sentiment. The system helps in understanding user feedback and improving services.

Figure [3] : Email Sentiment Analysis User Dashboard

Figure [3] The dashboard provides options to analyze sentiment from a single email or multiple emails. Users can navigate through features and securely logout from the system. The interface confirms successful login and allows further sentiment analysis operations.

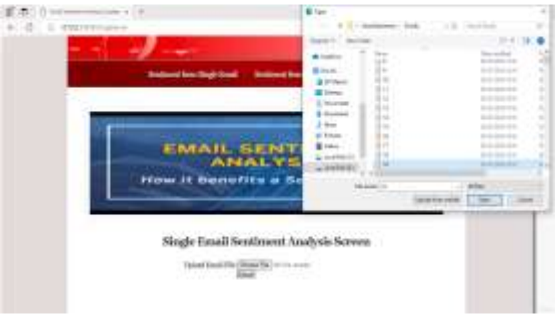

Figure [4] : Single Email Sentiment Analysis Upload Interface

Figure[4] The screen allows users to upload a single email file for analysis. A file selection window is used to choose the email from the system.
The uploaded email is processed to determine its sentiment.

Figure [5] : Single Email Sentiment Prediction Result

Figure[5] The system displays the uploaded email content for analysis. It processes the text and determines the sentiment of the email. The predicted result is shown as “Extremely Happy” based on the analysis.

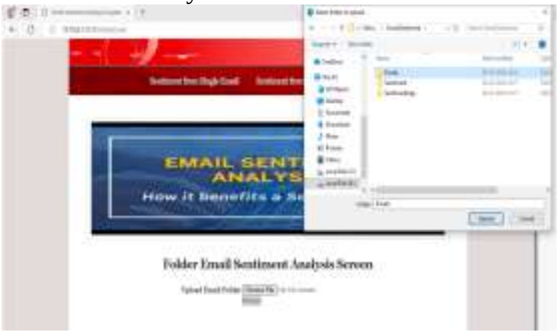

Figure [6] : Upload Email Folder for Sentiment Classification

Figure [6] This screen shows an Email Sentiment Analysis system where a user can upload a folder of emails. Clicking “Select” opens a file dialog to choose a folder from the computer. After selecting, the system uploads the emails and analyzes their sentiment

Figure[7] : Email Sentiment Analysis Results Screen

Figure[7] This screen displays multiple emails along with their analyzed sentiment labels such as Happy, Neutral, and Extremely Happy. Each row represents an email message, and the system classifies the emotional tone automatically. It helps users quickly understand the overall sentiment of large volumes of email data.

DISCUSSION

The results highlight the effectiveness of integrating a sentiment analysis model with a user-friendly web application. One of the key strengths of the system is its ability to handle both individual and bulk email analysis without significant performance issues. The preprocessing step plays an important role in improving prediction quality by removing noise and standardizing the text. However, certain challenges were observed when dealing with complex sentences or mixed emotions, where the model may not always capture the exact sentiment. This suggests that further improvements can be made by incorporating more advanced models or expanding the training dataset. The system's design ensures ease of use, allowing users with minimal technical knowledge to operate it efficiently. Additionally, the structured display of results enhances readability and supports better decision-making. The project demonstrates practical applicability in areas such as customer feedback analysis and automated email sorting. Future enhancements could include real-time analysis, multilingual support, and improved accuracy through deep learning techniques.

VI. CONCLUSION

This project presents the design and implementation of a web-based system for analyzing the sentiment of email data in an efficient and user-friendly manner. The application allows users to upload single or multiple email files and obtain sentiment predictions with ease. By combining text preprocessing techniques with a classification model, the system is able to identify the emotional tone of emails and present the results in a clear and structured format. The overall workflow, from user registration to result display, ensures smooth interaction and accessibility for users with different levels of technical knowledge. The system demonstrates practical usefulness in areas such as customer feedback analysis, email organization, and decision-making support. It reduces manual effort and speeds up the process of understanding large volumes of textual data. Although the current implementation performs effectively, there is scope for further enhancement by incorporating advanced models, improving accuracy, and supporting multiple languages. Overall, the project highlights the importance of applying sentiment analysis techniques to real-world communication data and provides a strong foundation for future development.

REFERENCES

[1] A. Al-Alwani, "An innovative approach to email response algorithms for managing email systems," Journal of Computer Science, vol. 10, no. 4, p. 689, 2014.

[2] S. Aman and S. Szpakowicz, "Detection of emotional expressions in textual data," in Proceedings of the International Conference on Text, Speech and Dialogue, Springer, 2007, pp. 196–205.

[3] Teacher Registration Board of Western Australia, Professional Boundaries Between Teachers and Students: A Guide for WA Educators, Department of Education Services, Government of Western Australia, 2017.

[4] Federal Energy Regulatory Commission, "Analysis of the western energy crisis, Enron's collapse, and the response by FERC," 2009.

[5] C. Danescu-Niculescu-Mizil and L. Lee, "Exploring linguistic style coordination in conversational settings," in Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics, Association for Computational Linguistics, 2011, pp. 76–87.

[6] P. Ekman, "Fundamental human emotions," in Handbook of Cognition and Emotion, 1999, pp. 45–60.

[7] A. Go, R. Bhayani, and L. Huang, "Sentiment classification of Twitter data using distant supervision techniques," Stanford CS224N Project Report, vol. 1, no. 12, 2009.

[8] N. Gupta, M. Gilbert, and G. Di Fabbrizio, "Email-based emotion recognition in customer support systems," Computational Intelligence, vol. 29, no. 3, pp. 489–505, 2013.

[9] Á. Horgan and J. Sweeney, "Online behavior of university students and their use of internet resources for health-related information," CIN: Computers, Informatics, Nursing, vol. 30, no. 8, pp. 402–408, 2012.

[10] E. Huijzer, "Evaluating affective email responses and predicting customer sentiment after communication," Conference Proceedings, 2017.

Babburi, S. Lightweight Distributed Provenance Framework for Edge and IoT Data Systems.

Gaddam, S. From Fixed Specifications to Self-Adapting Systems: A Machine Learning Perspective on Software Engineering.

Immadi, S. K. (2025). Optimizing ERP for Human Capital Management. Applied Research for

Growth, Innovation and Sustainable Impact, 377–384. https://doi.org/10.1201/9781003684657-63

Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.

Poojari, R. Frameworks for Data Management and Lineage in Large-Scale Healthcare Data Systems.

Mahimalur, R. K., Vasgam, M., & Manoharan, D. Devops Lifecycle Management And Cloud Migration Assessments: A Security-Driven CICD Perspective.

Viswanathan, V. Generative AI for Smarter Workforce Planning and Enterprise Resource Decisions.

Gajula, S. (2026, March). Two Pillars of Banking Intelligence: A Comparative Analysis of AI Techniques for Fraud Prevention and Churn Mitigation. In 2026 14th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1-6). IEEE.

Maturi, S. Y. (2023). Crowdsourced frontier: Unveiling autonomous adversarial cybercapabilities via open AI competition. International Journal of Intelligent Systems and Applications in Engineering, 11(1s), 275–284.

Venkata Ramana, P. (2024). AI-driven predictive analytics in ERP systems for proactive supply chain optimization. International Journal of Research in Information Technology and Computing, 8(4).

Srikanth Kavuri. (2022). Large Language Model (LLM)-Based Automation for Software Test Script Generation. Computer Fraud and Security. https://doi.org/10.52710/cfs.836

Gajula, S. (2025). AI-Powered Forecasting Models, Optimizing Working Capital, Supply Chain Financing. 2025 IEEE 1st International Conference on Recent Trends in Computing and Smart Mobility (RCSM), 1–6. https://doi.org/10.1109/rcsm67767.2025.11507813

Shashank, A. (2025). AI-Enhanced ETL Processes: Leveraging Artificial Intelligence for Optimized Data Integration Systems. Journal Of Multidisciplinary, 5(8), 219-225.

Kandula, S. T. R., Susarla, R. S., & Boyapati, P. K. (2025, July). Enhanced Cyber Security Using Global Local Artificial Neural Network Based Intrusion Detection in Big Data Environment. In 2025 IEEE 4th World Conference on Applied Intelligence and Computing (AIC) (pp. 426-431). IEEE.

Kandula, S. T. R., Boyapati, P. K., & Susarla, R. S. (2025, April). Optimized Cloud Resource Management Using Deep Dendritic Artificial Neural Networks Integrated with Kubernetes and Terraform. In International Conference on Computer Vision and Robotics (pp. 454-465). Cham: Springer Nature Switzerland.