iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IX Month of publication: September 2025

DOI: https://doi.org/10.22214/ijraset.2025.74023

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

Emotion Sence -AI: Online Sentiment Analysis Platform for Detecting Emotions

Ms. Umme Salmah N[1], Mrs. Subhashree D C[2], Dr. Girish Kumar D[3], Mrs. Sharvani V[4], Ms. M M Harshitha[5]
Department of MCA, Ballari Institute of Technology & Management, Ballari, Karnataka, India

Abstract: Sentiment analysis is animportant part of natural language processing. Initially, the task was addressed using rule-based and statistical tools such as Naïve Bayes and Support Vector Machines (SVM). With the ever-increasing amount of available data, these methods became too simplistic.
Deep learning has revolutionized not only the task of sentiment analysis and other tasks in natural language processing. Currently, the best models are context-aware models like BERT. Despite their advent and the advances they have made. The main issues holding the field back is that it's and recognize unstructured data effectively, especially when that unstructured data is concatenated with sarcasm and ambiguity. This study investigated the performance of basic traditional models (Naïve Bayes, Lexicon-based methods, Linear SVM) and modern deep leaning model (BERT) when applied to emotional analysis on unresolved big data issues (customer review data).

I. INTRODUCTION

Opinion mining, which is often called sentiment analysis is acrucial component of text analysis and Natural Language Processing (NLP). It is especially true on the rise of user-generated content on platforms such as e-commerce websites, social media, and customer review sites. In the past, this area relied on rule-based systems and early ML techniques, which gives a structured but limited understanding of language. As online communication grew more complex and expressive, these traditional methods struggled to interpret the context and emotions in unstructured text.
The increasing amount of digital data from consumers presents companies with both an opportunity and a challenge: to make sense of the large volume of feedback, preferences, and complaints that come in text form. Businesses want to check customer requirementsby improving their user experiences. However, traditionalmethods struggled with sarcasm, idioms, technical terms, and inconsistent grammar that are common in everyday language. To this context issue, researchers had changed models to that leverage data and grasp context.
With deep learning and attention-based models, particularly BERT (Bidirectional Encoder Representations from Transformers), it's now possible to better capture the meaning and sentiment in text. Despite these advancements, a significant gap remains in understanding how well different models balance ease of understanding, computing power, and performance when applied to large and specialized datasets.
This paper presents a comprehensive approach to utilizing sentiment analysis in intelligent systems, with a specific focus on its application in recommendation systems. The research examines and compares several methods, including Naïve Bayes, Lexicon-based approaches, Support Vector Machines (SVM), and BERT, using a single dataset of Amazon product reviews. Each method is evaluated based on how well it detects sentiment patterns, its classification accuracy, and its performance in real-time situations. The system design includes modular components and supports ongoing improvement through feedback and retraining.
By comparing both old and new methods, this study gives best approaches for practical sentiment analysis. It also demonstrates how sentiment insights can enhance recommendation systems, making suggestions more personalized and emotionally aware. This research contribute to the understanding of NLP and smart decision-making systems, providing a foundation for future research and practical application in various industries.

II. LITERATURE REVIEW

The idea of using computers to examine emotions in text began with early researchers like Pang and Lee [1]. They implemented machine learning algorithms such as Naïve Bayes and Support Vector Machines (SVM) to categorize movie reviews as positive or negative. Their work highlighted the challenges in handling emotional text, such as selecting the right features and interpreting vague language. It also established a benchmark for future research.

Later, Liu [2] provided a clear explanation of opinion mining. He focused on both the overall text and individual sentences. He emphasized that using specific words for different topics that can improve classification accuracy.

Later, Cambria and White [4] suggested moving beyond just words to understand meaning at a deeper level through concept-based sentiment analysis.

They used knowledge bases and models of emotions to better understand feelings in language. This new approach allowed sentiment models to go beyond just looking at individual words and consider the bigger picture. At the same time, Hutto and Gilbert [14] created VADER, a system that uses rules and a list of sentiment words to handle tricky parts like negations, emphasis, and punctuation, especially in social media. Though easy to use and understand, VADER had limits when it came to understanding sarcasm or new language trends found in different kinds of data.

With the growth of deep learning, Zhang et al.

[3,15] demonstrated how character-level Convolutional Neural Networks (CNNs) can be implemented in sentiment analysis. These modelsis used to learn intricate patterns in text without requiring extensive manual setup. They also goeswell with misspellings and slang, which are common in informal writing. Similarly, Socher et al. [8] applied Recursive Neural Tensor Networks to grammatically parsed sentences to predict sentiment more accurately, especially in complex sentences. These advancements highlighted that understanding the structure and layers of language is essential for improved sentiment analysis.

The introduction of Transformer-based models changed natural language processing significantly.

Devlin et al. [7] presented BERT, a model trained on large amounts of text. It used techniques like predicting missing words and understanding how sentences connect. When fine-tuned for sentiment tasks, BERT achieved excellent results by grasping deep meaning and context in both directions of a sentence. Peters et al. [9] created ELMo, which generated word meanings that vary depending on their sentences. These transformer-based methods greatly improved performance compared to earlier models, especially for subtle and complex sentiment expressions.

III. PROPOSED FRAMEWORK

A. *Flow Diagram*

The flow diagram outlines a straightforward process for conducting sentiment analysis. It begins by loading an annoted datasets, such as customer reviews from Amazon. Firstly itclean the text and simplify it, which is referred as preprocessing. Next,text is converted into numbericals using methods like TF-IDF or word embeddings. These figures are applied to train either a machine learning model or a deep learning model. After preparing model,it is assessed through metrics such as accuracy and F1-score to determine its effectiveness. Once it successfully passes the test, the model can predict the sentiment of new data. The outcomes are visualizing tools like confusion matrices and word clouds for better understanding. The trained model is subsequently incorporated into an API for real-time usage. Finally, this API connects with other systems, such as recommendation tools or customer feedback platforms, completing the process from the original data to real-world application.

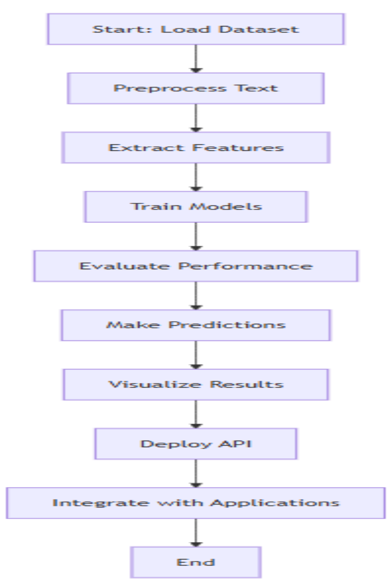


Fig1: Flow Diagram

In this, Sentiment analysis is used in a straightforward and structured manner. It works well for sorting real-world text. The whole approach is easy to understand and can be applied to other purposes, such as recommendation systems and platforms that examine opinions.

1) Dataset Collection and Labeling

The datasetsconsists of Amazon product reviews, which provide a comprehensive and diverse collection of user-generated feedback across multiple product categories. Each review includes textual content and a corresponding star rating, which is mapped into three sentiment classes for classification: 1–2 stars are labeled as **Negative**, 3 stars as **Neutral**, and 4–5 stars as **Positive**. This labeled structure allows supervised training and consistent evaluation of sentiment models across traditional and deep learning algorithms.

2) Data Preprocessing and Cleaning

The original review texts can be messy and not well organized, with extra symbols, emojis, and casual language. To make the text clear and useful, some cleaning steps are done. First, all letters are made lowercase so everything looks the same. Then, punctuation, special signs, web links, and numbers are taken out using a pattern matching method. Next, the text is segmentted into individual words or smaller parts. Words that are generally used but don't affect the meaning, like "the," "is," and "was," are removed. Words are also shortened to their basic forms through a process called stemming or lemmatization to make the language more uniform. All these steps help make the data cleaner and easier to work with.

3) Feature Extraction and Text Vectorization

Once the text is cleaned, it must be converted into numbers for the model to use. Older models use methods like Bag of words with TF-IDF to change text into fixed-length groups of numbers that show the frequencies of words appear or how important they are. Newer models, particularly BERT, take a different approach. They convert words into numbers based on their meaning in the sentence. These groups of numbers show what words are presentand their relationship with eachother. This is especially useful when dealing with complex emotions or vague feelings.

4) Model Training and Evaluation

Once the text is converted into vector form, several classifiers are used to evaluate their performance. Naïve Bayes is chosen because it is easy to use and fast, making it suitable for real-time or simple apps. Support Vector Machines (SVM) work well in situations with many data features and are tested using TF-IDF vectors. Lexicon-based methods like VADER use sentiment word lists to determine the tone of text by analyzing words that express positive or negative feelings. Lastly, BERT is trained on the review data to take advantage of its ability to understand language in both directions, enabling more precise and context-aware classification. Each model gets trained with a split where 80% of the dataset is allocated for training and 20% for testing. Accuracy of Object Detection: The system's ability to correctly identify and track objects like pedestrians, other vehicles, and obstacles.

5) Prediction, Visualization, and Insights

After training, models predict the sentiment of unseen data. Visual tools display the results, enhancing clarity that how well the model is performing. Confusion matrices reveal the trained model accuracy for each sentiment type. ROC curves demonstrate the model's ability to distinguish between different sentiments. Precision-recall graphs assess the model's performance with unbalanced data. Additionally, word clouds are created for both positive and negative texts to highlight the most common words that influence the model's predictions. These tools assist developers and decision-makers in understanding trends and the model's behavior.

6) Deployment and Integration Framework

Sentiment analysis system can be made easy to use, a REST API is built using Flask or FastAPI, with endpoints for real-time sentiment predictions. The system can fit intolarger apps, such as recommendation tools, customer feedback systems, or CRM software.It's also designed to grow by allowing updates with new data over time and by working on cloud services like AWS or Google Cloud.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IX Sep 2025- Available at www.ijraset.com

IV. EVALUATION &RESULTS

A. Model Performance Metrics

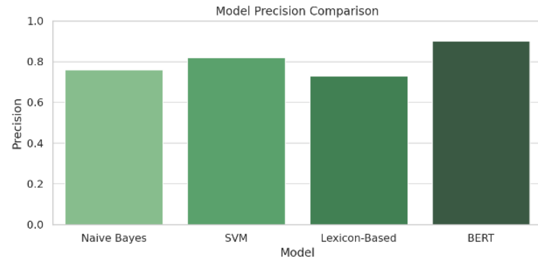


Fig 2: Model Precision Comparison

To assess how well the sentiment analysis system performs, we used four main measures: Accuracy, precision, recall, and F1-score. Accuracy shows how often the model gives the right answer by comparing the number of correct predictions to all predictions made. Precision looks at how many of the positive predictions the model makes are actually correct, making sure it doesn't wrongly label neutral or negative statements as positive. Recall checks how well the model finds all the correct sentiment cases in a group, showing how good it is at spotting true emotions. The F1-score is a harmonic mean of precision and recall, and its effectiveness when dealing with unbalanced data. These four measures were picked to give a full picture of how the model performs with real sentiment data.

B. Model Comparison and Analysis

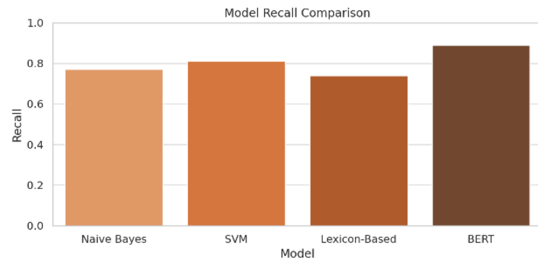


Fig 3: Model Recall Comparison

Four models, Naïve Bayes, Support Vector Machine (SVM), Lexicon-Based (VADER), and BERT, were trained and tested on the same Amazon review dataset. The BERT model had the highest accuracy at 91%, followed by SVM at 84%, Naïve Bayes at 78%, and the Lexicon-Based approach at 75%. BERT also achieved the best scores by showing understanding the subtle emotions in text better. While Naïve Bayes and SVM performed reasonably well, they struggled with complex language, such as idioms and sarcasm. Lexicon-based methods are simple, but they didnt perform as well in precision and recall since they rely on fixed rules that do not adapt easily. These outcomes highlights the effectiveness of transformer models like BERT in areas where sentiment is varied and complex.

V. CONCLUSION

This study presents a complete sentiment analysis system that addresses the challenge of identifying and categorizing emotions in large volumes of disorganized text. It begins by gathering real-world data from Amazon product reviews. The method follows a clear process that includes cleaning the text, extracting key features, training models, and presenting the results. The research used and tested both traditional models and deep learning models, applying standard methods to measure performance. The outcome highlights the traditional models like Naïve Bayes and SVM are effective and cheaper to operate, transformer-based models like BERT achieve significantly higher F1-score. This demonstrates that the system can manage complex language, context, and subtle emotions.

The results show that the system is effective at solving the main problem it was designed for, improving the accuracy and trustworthiness of sentiment analysis for real-world uses.

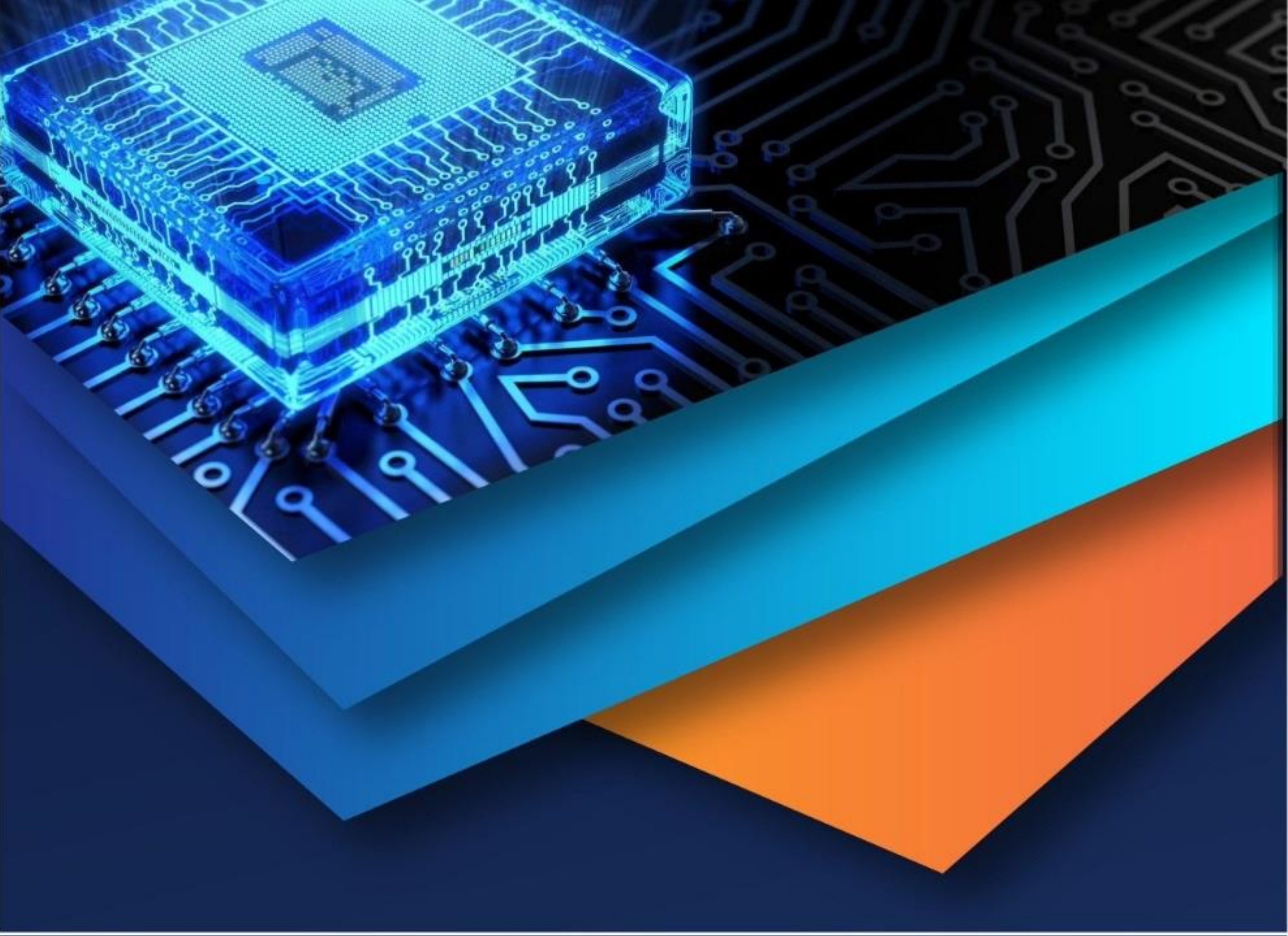
The system is designed to be flexible and simple to use in various areas, such as recommendation tools, customer feedback systems, and market trend trackers. Although deep learning models require greater time and resources to train, their strong performance is worth the effort, especially when context matters. The system also includes tools to visualize and deploy the results. This makes sentiment analysis both clear and easy to use, providing real value for businesses and researchers.

Looking ahead, the system can be improved in many ways.

Adding multilingual capabilities will make it better for specific areas through fine-tuning. Using tools like Kafka or Spark for real-time sentiment analysis will increase its usefulness across different groups and industries. Explainability tools like SHAP and LIME will clarify the model's decisions, which will help build trust. Including aspect-based sentiment analysis will provide more detailed feedback, allowing businesses to understand feelings about specific parts of a product or service. These future improvements will enhance the system, leading to smarter and more user-focused sentiment analysis tools.

REFERENCES

[1] Sharma, H., & Bansal, D. (2023). Opinion Mining Using Double Channel CNN for Recommender System. arXiv preprint arXiv:2307.07798.
[2] Singh, A., & Kaur, N. (2023). Improving Collaborative Filtering-Based Recommender System Using Sentiment Analysis. International Journal of Innovative Technology and Exploring Engineering.
[3] Kumar, A., & Tyagi, S. (2024). Efficient Sentiment Analysis: A Resource-Aware Evaluation of Feature Extraction Techniques, Ensembling, and Deep Learning Models. Journal of King Saud University – Computer and Information Sciences.
[4] Muhammad, N., Javed, A. R., & Ahmed, S. H. (2024). Deep Learning-Based Sentiment Classification: A Comparative Survey. IEEE Access.
[5] Reddy, A., Kumar, R., & Goyal, D. (2023). A Smart Recommender Model Based on Learning Method for Sentiment Classification. Journal of Intelligent & Fuzzy Systems.
[6] Zhang, S., Zhang, Y., He, X., & Chua, T.-S. (2024). Recommender Systems in the Era of Large Language Models (LLMs). arXiv preprint arXiv:2403.01871.
[7] Chopra, R., & Sharma, S. (2024). Innovative Sentiment Analysis and Prediction of Stock Price Using FinBERT, GPT-4, and Logistic Regression. International Journal of Advanced Computer.
[8] Kumari, P., & Kumar, R. (2023). Comparative Evaluation of Deep Learning and Machine Learning Techniques for Sentiment Analysis of Electronic Product Review Data. Procedia Computer Science.
[9] Sharma, M., & Sharma, A. (2024). Challenges and Future in Deep Learning for Sentiment Analysis: A Review and a Proposed Novel Hybrid Approach. Journal of Big Data.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY