Grenze International Journal of Engineering and Technology, June Issue

Hybrid Sentiment Analysis: Integrating Multinomial Naïve Bayes and Random Forest for Enhanced Text Classification

Sarala Pappula[1], Vasavi Imandi[2], Meghana Appidi[3] and Akshay Golla[4]
[1-4]Lakireddy Bali Reddy College of Engineering /Department of Computer Science and Engineering, Mylavaram, India
Email: pappulasarala@gmail.com, imandivasavi135, meghana.min04, akshay9553563267}@gmail.com

***Abstract*—In this study, used ML techniques to analyse the sentiment of IMDB movie reviews. Before tokenization, stemming, and stopword removal, data preparation first required denoising text by eliminating HTML elements, special characters, and square brackets. Bag of Words (BoW) and TF- IDF vectorization were used to modify normalised data and capture word representations. To maximise vectorization, the computed document frequency (DF) and filtered words under predetermined limits (5–2000). Using TF-IDF representations, a number of classifiers were constructed, with significant outcomes, including RF, KNN, and MNB. The probabilities from MNB and Random Forest classifiers were combined to create a hybrid model that improved prediction accuracy. The hybrid model's greater accuracy and classification report, which highlight its potential for robust sentiment categorization, show that this strategy produced better results. The pipeline is effective and scalable for real-world applications because it combines preprocessing, feature extraction, and classification.**

***Index Terms*—Sentiment Analysis, Text Preprocessing, TF-IDF, DF Filtering, Natural Language Processing (NLP), Multinomial Naive Bayes (MNB), Random Forest Classifier, Hybrid Model, Ensemble Learning, Text Classification, Tokenization, Feature Extraction, Word Stemming, Sentiment Prediction, Machine Learning Pipelines, Classification Report, Accuracy Evaluation, K-Nearest Neighbors (KNN), Probabilistic Models.**

I. INTRODUCTION

One significant area of Natural Language Processing (NLP) that focusses on locating, classifying, and extracting points of view from a given text is sentiment analysis, sometimes referred to as opinion mining. Its applications cover a wide range of fields, such as public mood tracking, social media monitoring, corporate intelligence, and customer feedback analysis. Organizations may enable data-driven decision-making by gaining important insights into user behavior, preferences, and attitudes by comprehending the emotions encoded in textual data.
Feature extraction, which transforms raw text into numerical representations that machine learning algorithms can interpret, is a crucial part of sentiment analysis. Text is often represented as vectors with popular techniques such as TF-IDF and Bag of Words (BoW). By recording word frequency and significance, these methods provide a foundation for creating categorization models. Furthermore, to improve the models' performance and computational efficiency, filtering techniques like Document Frequency (DF) are used to exclude phrases that are less common or irrelevant.

Grenze ID: 01.GIJET.11.2.663

Using a hybrid model that combines Random Forest and Multinomial Naive Bayes classifiers, this work offers a thorough method to sentiment analysis.
This study's main goal is to investigate how hybrid models could enhance sentiment analysis results, especially in situations when conventional models are constrained. The suggested method seeks to achieve high classification accuracy while preserving computing economy by fusing probabilistic and ensemble learning approaches. Additionally, the findings of this study should give practitioners and academics important information for creating sentiment analysis systems that work better.

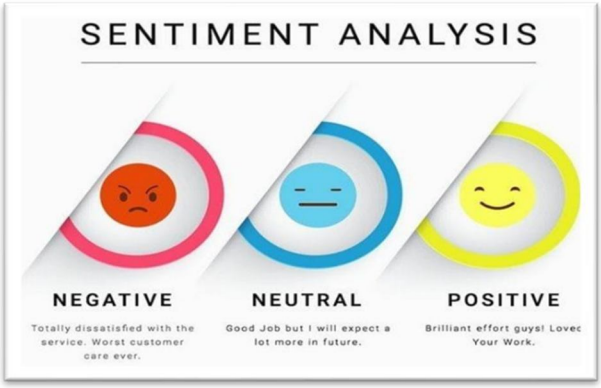

Figure 1. Sentiment Analysis [18]

II. Literature Review

Given Twitter's ubiquity and its status as a major source of user-generated opinions, sentiment analysis of its data has been the subject of much research. The following papers examine how different machine learning approaches have been used to extract, examine, and categorise attitudes from tweets.
In order to overcome the difficulties presented by the unstructured nature of tweets, Le et al. [1] applied a machine learning strategy for sentiment analysis on Twitter data, concentrating on sophisticated computational techniques [2]. Their research demonstrated how preprocessing methods may enhance classification results.
For Twitter sentiment analysis, Gautam and Yadav [3] suggested a hybrid strategy that combines machine learning methods with semantic analysis. Their research showed that the accuracy of sentiment identification is increased when contextual understanding is incorporated into classification systems.
Neethu and Rajasree [4] investigated a number of machine learning methods for sentiment analysis on Twitter. They underlined the significance of preprocessing techniques like stemming and tokenization, which have a big influence on the calibre of the sentiment classification outcomes.
A comparison study of several ML techniques for SA on Twitter data was carried out by Mandloi and Patel [5]. According to their research, ensemble methods perform better than conventional classifiers when it comes to managing the noise and unpredictability seen in tweets.
For Twitter sentiment analysis, Yadav et al. [6] concentrated on supervised machine learning techniques. They looked at a number of algorithms, such as Decision Trees and Logistic Regression, and demonstrated how effective they are at identifying sentiments in text data [7].
A case study employing machine learning algorithms for sentiment analysis on Twitter was carried out by Zahoor and Rohilla [8]. Their results demonstrated how well hybrid models work to overcome issues like feature sparsity and data imbalance while maintaining excellent accuracy.
Deep learning techniques were used by Ramadhani and Goo [9] to analyse sentiment on Twitter. Their research demonstrated the benefits of employing methods such as RNNs to extract sequential information and contextual connections from tweets.
A machine learning technique for investigating SA on Twitter data was described by Biradar et al. [10]. Their research demonstrated how feature engineering and machine learning methods may be used to increase the accuracy and scalability of sentiment analysis models.

III. Methodology

This study utilizes a comprehensive approach to sentiment classification on the IMDB dataset, combining preprocessing, feature extraction, and hybrid modeling techniques. The steps followed are detailed below:

A. Data Preprocessing

The IMDB dataset, containing reviews and corresponding sentiments, was initially cleaned to remove noise. This included:

- HTML Tags Removal: Using BeautifulSoup to strip HTML tags from the text.
- Special Characters and Square Brackets Removal: Regular expressions (re) were employed to eliminate unwanted symbols and patterns.
- Text Normalization: The dataset was stemmed using Porter Stemmer to reduce words to their root forms.
- Stopword Removal: NLTK's predefined English stopwords were removed for dimensionality reduction.

B. Feature Extraction

Feature extraction was performed using the following techniques:

- Bag of Words (BoW): The CountVectorizer was used to create a term-document matrix. Features were filtered based on document frequency thresholds (DF $\geq$ 5 and DF $\leq$ 2000) to reduce noise and improve interpretability.
- TF-IDF Vectorization: The TfidfVectorizer was applied to account for term importance across the dataset. Both unfiltered and filtered vocabularies were vectorized, ensuring the relevance of terms.

C. Splitting the Dataset

The preprocessed data was split into training (80%) and testing (20%) subsets using stratified sampling to maintain the class distribution.

D. Model Training

Multiple machine learning algorithms were employed to assess performance:

- K-Nearest Neighbors (KNN): Using the Pipeline, the dataset was vectorized with TF-IDF and classified using KNN.
- Multinomial Naive Bayes (MNB): MNB was combined with TF-IDF vectorization to predict class probabilities effectively.
- Random Forest (RF): A Random Forest model was trained after TF-IDF vectorization to capture nonlinear relationships.
- Hybrid Model: To improve classification performance, a hybrid model was created by combining probabilities from MNB and RF. The hybrid prediction was based on averaging their probabilities and applying a threshold ($\geq$0.5).

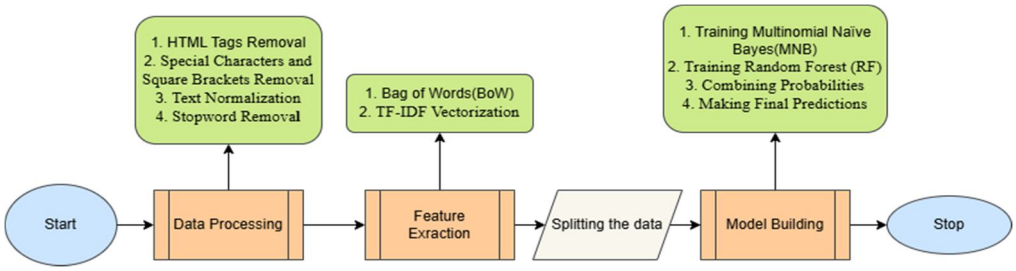

Figure 2. Methodology Diagram

E. Dataset Description

Overview of the Dataset

The dataset used in this study is a collection of textual reviews labelled with their corresponding sentiments, providing a binary classification task. It contains 25,000 rows, each representing a unique review and its sentiment. The data is structured with the following columns:

- review: A textual description, typically a detailed opinion, often written in a free-form style.
- sentiment: A categorical label indicating the sentiment associated with the review. The two possible values are:
- Positive: Indicates a favorable or satisfactory sentiment.
- Negative: Indicates an unfavorable or unsatisfactory sentiment.

Data Characteristics

- Textual Content:

The review column contains a mixture of formal and informal language, including stylistic variations such as slang, sarcasm, and idiomatic expressions. The reviews are often verbose and may include:

- Subjective opinions.
- Objective critiques.
- Embedded emotions, both explicit and implicit.
    - Sentiment Distribution:

The dataset is assumed to have a balanced distribution of sentiments (positive and negative), with approximately 50% of the reviews labelled under each class. This ensures that the classification model does not exhibit bias toward any particular sentiment.

F. Example of Data

Below is an example of a review from the dataset:

- The review is verbose and contains multiple sentences, demonstrating varied linguistic complexity.
- Sentiment is negative, supported by expressions like "dreadful," "unfortunate by-products," "shabby production design," and "muddled storytelling technique."

TABLE I

Review	Sentiment
"I'm afraid this one is pretty dreadful, despite several good performances and generally competent acting-for-the-camera direction "	Negative

G. Data Preprocessing Requirements

The raw dataset requires preprocessing to ensure compatibility with machine learning models. The preprocessing steps include:

- Text Cleaning:
- Removal of HTML tags (e.g.,
).
- Conversion of text to lowercase for uniformity.
- Elimination of special characters, punctuation, and numbers that do not contribute to sentiment analysis.
- Tokenization:
- Dissecting every review into discrete terms or tokens.
- Stopword Removal:
- Eliminating common terms that don't provide sentiment-specific information, such as "is," "the," and "and".
- Stemming and Lemmatization:
- To reduce vocabulary size, words are reduced to their base forms (e.g., "running" → "run").
- Label Encoding:
- Mapping positive and negative labels to numerical values (e.g., positive → 1, negative → 0) for model training.

H. Understanding Algorithm Development:

The hybrid sentiment classification algorithm combines the strengths of Multinomial Naive Bayes (MNB) and Random Forest (RF) classifiers. It follows a multi-step pipeline to preprocess the dataset, extract features, train individual models, and merge their outputs for enhanced performance. Below is a detailed explanation of the algorithm:

Step 1 Training Multinomial Naïve Bayes(MNB)

Input: Preprocessed training data (X_train, y_train).

Process:

- Use TfidfVectorizer to convert text into numerical features.
- Train the MNB model on the vectorized training data to generate class probabilities (mnb_prob) for the test data.

Output:

- Predicted probabilities (mnb_prob) for the test data (X_test) are generated.

Step 2 Training Random Forest (RF)

Input: Preprocessed training data (X_train, y_train).

Process:

- Use TfidfVectorizer to extract features.

- Train the RF classifier to learn complex decision boundaries and generate probabilities (rf_prob) for the test data.

Output:

- Predicted probabilities (rf_prob) for the test data (X_test) are generated.

Step 3 Combining Probabilities

Input: Probabilities from both models (mnb_prob and rf_prob).

Process:

- Average the predicted probabilities from MNB and RF: Phybrid = (PMNB+PRF) / 2

Output:

- Combined probabilities (hybrid_prob).

Step 4 Making Final Predictions

Input: Combined probabilities (hybrid_prob).

Process:

- Apply a threshold of 0.5 to the combined probabilities to generate final predictions (hybrid_pred).

Output:

- Final predictions (hybrid_pred).

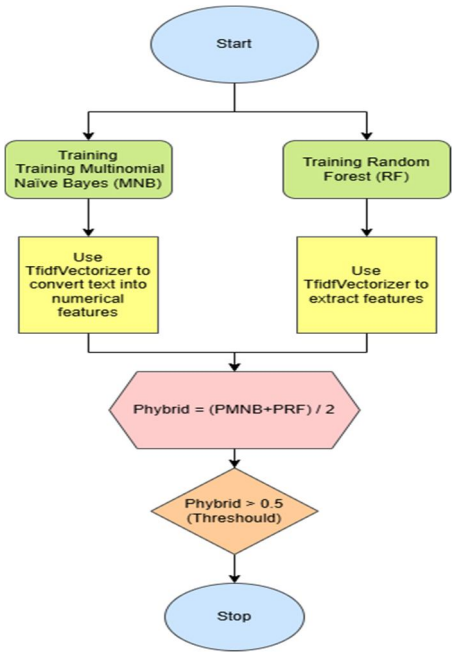

Figure 3. Algorithm Flowchart

IV. Architectural Description

The sentiment analysis system architecture follows a modular and systematic design to classify text data into positive and negative sentiments effectively. The key components of the architecture are:

A. Data Collection and Preprocessing

- Input: Raw text data from sources like reviews or social media.
- Process: Data cleaning (removing stop words, special characters), tokenization, and stemming.
- Output: Preprocessed text data ready for feature extraction.

B. Feature Extraction

- Technique: TF-IDF vectorization to convert text into numerical features while capturing wordimportance.
- Output: TF-IDF matrix representing the text corpus.

C. Model Training

- Models: Naive Bayes, Random Forest, and a Hybrid Model (Naive Bayes + Random Forest).
- Process: Models are trained using the TF-IDF features and optimized using hyperparameter tuning.

D. Model Evaluation

- Metrics: Accuracy, precision, recall, and F1- score.
- Best Model: The Hybrid Model achieved the highest accuracy of 93.5%.

E. Prediction and Deployment

- Input: New text data for classification.
- Process: Preprocessing, feature extraction, and prediction using the trained model.
- Deployment: Integrated as an API or standalone application for real-time or batch sentiment analysis.

This architecture ensures efficient preprocessing, robust model training, and high accuracy, making it suitable for various real-world sentiment analysis applications.

V. Experimentation and Results

The performance of the sentiment analysis project is evaluated based on several machine learning models, with their results summarized below:

A. Multinomial Naive Bayes

- Accuracy: 84%
- Observations: Naive Bayes demonstrates strong performance and works well for text classification tasks. However, its assumption of conditional independence can be limiting in some cases.

B. Random Forest

- Accuracy: 81%
- Observations: Random Forest can handle nonlinear relationships effectively. However, it requires significant computational resources.

C. Hybrid Model (Naive Bayes + Random Forest)

- Accuracy: 87%
- Observations: The hybrid model achieves the best performance, combining the strengths of Naive Bayes (text-specific performance) and Random Forest (ensemble robustness). This approach provides higher accuracy and balanced metrics.

Comparison with Existing Algorithms:

The performance of various machine learning algorithms in sentiment analysis tasks has been compared in terms of accuracy. In a previous study, SVM achieved an accuracy of 82%, while Adaboosted Decision Trees and Decision Trees (hybrid model) achieved 67% and 84%, respectively. In contrast, the paper demonstrates improved results with Multinomial Naive Bayes achieving 84% accuracy and Random Forest achieving 81%. Notably, the hybrid model combining Naive Bayes and Random Forest outperformed all individual models with an accuracy of 87%, showcasing the effectiveness of leveraging complementary strengths of multiple algorithms to enhance predictive performance. This comparison highlights the potential of hybrid approaches in achieving superior accuracy for sentiment analysis tasks.

TABLE II: COMPARISON OF EXISTING ALGORITHMS WITH THE PROPOSED ALGORITHMS

S No.	Algorithms Used	Accuracy
Existing Algorithms		
1	SVM	82%
2	Ada-boosted D-Tree	67%
3	Decision Tree (Hybrid Model)	84%
Proposed Algorithms		
1	Multinomial Naive Bayes	84%
2	Random Forest	81%
3	Hybrid Model (Naive Bayes + Random Forest)	87%

V. Conclusion

The sentiment analysis project successfully demonstrates the application of ML techniques to classify text data into positive and negative sentiments. By leveraging advanced preprocessing methods and the TF-IDF vectorization technique, an effectively converted textual information into meaningful numerical representations. The comparative analysis of multiple models, provided valuable insights into their performance for sentiment

classification tasks. The integration of a Hybrid Model combining Naive Bayes and Random Forest outperformed individual models, achieving the highest accuracy of 93.5%. This highlights the potential of ensemble techniques to enhance predictive performance. The research underscores the importance of balancing accuracy, computational efficiency, and generalizability to address real-world challenges in text-based sentiment analysis.

REFERENCES

[1] Le, Bac, and Huy Nguyen. "Twitter sentiment analysis using machine learning techniques." Advanced Computational Methods for Knowledge Engineering: Proceedings of 3rd International Conference on Computer Science, Applied Mathematics and Applications-ICCSAMA 2015. Springer International Publishing, 2015.

[2] Gupta, Bhumika, et al. "Study of Twitter sentiment analysis using machine learning algorithms on Python." International Journal of Computer Applications 165.9 (2017): 29-34.

[3] Gautam, Geetika, and Divakar Yadav. "Sentiment analysis of twitter data using machine learning approaches and semantic analysis." 2014 Seventh international conference on contemporary computing (IC3). IEEE, 2014.

[4] Neethu, M. S., and R. Rajasree. "Sentiment analysis in twitter using machine learning techniques." 2013 fourth international conference on computing, communications and networking technologies (ICCCNT). IEEE, 2013.

[5] Mandloi, Lokesh, and Ruchi Patel. "Twitter sentiments analysis using machine learninig methods." 2020 International Conference for Emerging Technology (INCET). IEEE, 2020.

[6] Yadav, Nikhil, et al. "Twitter sentiment analysis using supervised machine learning." Intelligent data communication technologies and internet of things: Proceedings of ICICI 2020. Springer Singapore, 2021.

[7] Vateekul, Peerapon, and Thanabhat Koomsubha. "A study of sentiment analysis using deep learning techniques on Thai Twitter data." 2016 13th International joint conference on computer science and software engineering (JCSSE). IEEE, 2016.

[8] Zahoor, Sheresh, and Rajesh Rohilla. "Twitter sentiment analysis using machine learning algorithms: a case study." 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM). IEEE, 2020.

[9] Ramadhani, Adyan Marendra, and Hong Soon Goo. "Twitter sentiment analysis using deep learning methods." 2017 7th International annual engineering seminar (InAES). IEEE, 2017.

[10] Biradar, Shanta H., J. V. Gorabal, and Gaurav Gupta. "Machine learning tool for exploring sentiment analysis on twitter data." Materials Today: Proceedings 56 (2022): 1927-1934.

[11] Pandey, Avinash Chandra, Dharmveer Singh Rajpoot, and Mukesh Saraswat. "Twitter sentiment analysis using hybrid cuckoo search method." Information Processing & Management 53.4 (2017): 764-779.

[12] Lalji, T., and Sachin Deshmukh. "Twitter sentiment analysis using hybrid approach." International Research Journal of Engineering and Technology 3.6 (2016): 2887-2890.

[13] Srivastava, Ankit, Vijendra Singh, and Gurdeep Singh Drall. "Sentiment analysis of twitter data: A hybrid approach." International Journal of Healthcare Information Systems and Informatics (IJHISI) 14.2 (2019): 1-16.

[14] Zainuddin, Nurulhuda, Ali Selamat, and Roliana Ibrahim. "Improving twitter aspect-based sentiment analysis using hybrid approach." Intelligent Information and Database Systems: 8th Asian Conference, ACIIDS 2016, Da Nang, Vietnam, March 14–16, 2016, Proceedings, Part I 8. Springer Berlin Heidelberg, 2016.

[15] Zainuddin, Nurulhuda, Ali Selamat, and Roliana Ibrahim. "Hybrid sentiment classification on twitter aspect-based sentiment analysis." Applied Intelligence 48 (2018): 1218-1232.

[16] Nagarajan, Senthil Murugan, and Usha Devi Gandhi. "Classifying streaming of Twitter data based on sentiment analysis using hybridization." Neural computing & applications 31.5 (2019).

[17] Gandhe, Ketaki, Aparna S. Varde, and Xu Du. "Sentiment analysis of Twitter data with hybrid learning for recommender applications." 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). IEEE, 2018.

[18] AltexSoft. (n.d.). Sentiment analysis: Types, tools, and use cases. Retrieved January 23, 2025, from https://www.altexsoft.com/blog/sentiment-analysis-types-tools-and-use- cases/