## **Sentiment Analysis using logistic Regression and naïve bayes**

1st Soni Singh

*Lovely professional University,*

*Jalandhar,*

*Punjab, India*

soni.singh@lpu.in

2nd Himanshu Sharma

*Department of computer science, School of Engineering,*

*Lovely professional university,*

*Jalandhar, Punjab, India*

Himanshusharma5908@gmail.com

3rd Vaibhav Raj

*Department of computer science, School of Engineering,*

*Lovely professional university,*

*Jalandhar, Punjab, India*

Vaibhavnoogler2211@gmail.com

***Abstract***—This paper presents a sentiment analysis approach for classifying airline-related tweets into different sentiment categories (positive, neutral, and negative) using machine learning techniques. The dataset consists of airline-related tweets labelled with sentiments, which are pre-processed to remove noise, including mentions, URLs, and special characters. We utilize two widely known machine learning models—Logistic Regression and Naive Bayes, implemented with feature extraction techniques such as TF-IDF vectorization and word embeddings. To handle class imbalance, we apply Synthetic Minority Over-sampling Technique (SMOTE), which generates synthetic data points for underrepresented classes. The models are trained and tuned using GridSearchCV to find the best hyperparameters. Furthermore, an ensemble method in the form of a soft-voting classifier is employed to combine the strengths of the individual models. Experimental results demonstrate the efficacy of this approach in improving classification performance, with detailed evaluation metrics including accuracy, precision, recall, and F1-score. This work highlights the importance of preprocessing, feature engineering, and class imbalance handling in sentiment analysis tasks, providing a robust framework for analysing social media sentiments related to airlines.

***Keywords—*** Sentiment Analysis, Text Classification, Machine Learning, Logistic Regression, Naive Bayes, Word Embeddings, TF-IDF, SMOTE, Soft Voting Classifier, Airline Sentiment, Class Imbalance, Grid Search, Feature Engineering, Twitter Data, Text Preprocessing

# Introduction

This study presents a sentiment analysis approach to classify airline-related tweets into different sentiment categories using machine learning techniques. Sentiment analysis, also known as opinion mining, is the process of identifying and categorizing opinions expressed in text, particularly to determine whether the writer's attitude towards a specific topic is positive, negative, or neutral. This technique has become increasingly crucial in the field of natural language processing, as it provides businesses and organizations with valuable insights from vast amounts of unstructured textual data, such as social media posts, reviews, and customer feedback. Specifically, sentiment analysis on social media platforms like Twitter has gained prominence due to the real-time nature of posts, allowing companies to monitor public sentiment and respond accordingly.

The primary focus of this study is on airline sentiment analysis, where the goal is to classify tweets related to airlines into sentiment categories—positive, neutral, or negative. This is particularly important for the airline industry, as customer feedback is essential in assessing service quality, operational performance, and overall customer satisfaction. While traditional methods of collecting and analyzing feedback through surveys or customer service interactions are still common, social media offers a more immediate and large-scale source of information. However, the vast volume of tweets and their informal, often noisy nature pose challenges for accurate sentiment classification.

To address these challenges, the researchers employ two widely used machine learning models—Logistic Regression and Naive Bayes—along with advanced text preprocessing techniques and feature extraction methods such as TF-IDF and word embeddings. Additionally, they handle the issue of class imbalance, a common problem in sentiment analysis datasets, through the application of Synthetic Minority Over-sampling Technique. This method synthesizes data for underrepresented classes, leading to improved model performance. The models are further optimized using GridSearchCV to find the best hyperparameters, and their predictions are combined using a soft-voting ensemble classifier to leverage the strengths of each model.

This research contributes to the growing body of work in sentiment analysis, particularly in the context of airline-related social media data. By demonstrating the effectiveness of combining preprocessing, machine learning models, and class imbalance handling techniques, the researchers provide a robust framework for sentiment analysis that can be adapted to various domains.

II. Literature Review

Sentiment analysis has become a crucial tool for extracting valuable insights from large volumes of text data, particularly in social media and customer reviews. The rapid growth of online platforms has led to an increased interest in sentiment analysis for understanding public opinion and improving customer satisfaction. This section reviews the key methodologies and approaches used in sentiment analysis, with a particular focus on the challenges, feature extraction techniques, and machine learning models commonly applied to text data.

Sentiment analysis has become a crucial tool for extracting valuable insights from large volumes of text data, particularly in the context of social media platforms and customer reviews. The rapid proliferation of online platforms has fueled an increased interest in leveraging sentiment analysis techniques to understand public opinion and enhance customer satisfaction. This section presents a comprehensive review of the key methodologies and approaches employed in sentiment analysis, with a particular focus on the challenges, feature extraction techniques, and machine learning models commonly utilized in text-based data analysis.

A. Sentiment Analysis in Social Media  
Sentiment analysis has been widely applied across various domains, including finance, politics, and healthcare, but one of the most prominent areas of research is social media sentiment analysis. Twitter, in particular, serves as an excellent source for analyzing real-time sentiments due to its microblogging nature and extensive user base. Go et al. proposed a method for sentiment classification of Twitter data using distant supervision, whereby they automatically labeled tweets based on keywords from known positive and negative lexicons. Their approach enabled rapid classification of large datasets, making it valuable for sentiment analysis in diverse fields, such as customer service, marketing, and brand management.

However, sentiment analysis of social media data, particularly Twitter, presents its own set of challenges. The informal, noisy nature of the text, which often includes slang, abbreviations, hashtags, and mentions, can significantly impact model performance. Researchers have developed various preprocessing techniques, such as removing URLs, user mentions, and stopwords, to clean up the text and ensure that the models focus on the most relevant information. These preprocessing steps, along with text normalization methods like stemming and lemmatization, are essential for improving classification accuracy.

B. Feature Extraction Techniques  
Feature extraction is a critical step in sentiment analysis, as it transforms raw text into numerical representations that can be fed into machine learning models. The most common feature extraction techniques are Bag-of-Words, Term Frequency-Inverse Document Frequency, and word embeddings. While BoW and TF-IDF capture the frequency of words and their relative importance in a corpus, they do not account for word order or semantic relationships between words. In contrast, word embeddings represent words in a continuous vector space where semantically similar words are placed closer together.

Word embeddings, such as Word2Vec and GloVe, have been widely used in sentiment analysis

C. Machine Learning Models for Sentiment Analysis

Various machine learning models have been applied to sentiment analysis tasks, with Naive Bayes, Logistic Regression, and Support Vector Machines (SVM) being some of the most commonly used classifiers. Naive Bayes is a probabilistic model that assumes independence between features and works well with large text corpora (Pang et al., 2002). Logistic Regression is a linear model that estimates the probability of a class label based on a set of input features, and it has been shown to perform well in sentiment analysis, particularly when dealing with high-dimensional data such as text (Tobias, 2016).

Support Vector Machines (SVM), with their ability to handle non-linear data through kernel methods, have also been widely adopted for sentiment analysis tasks, particularly when the data is high-dimensional (Joachims, 1998). However, these models often require careful tuning of hyperparameters to achieve optimal performance.

Ensemble methods, such as Voting Classifiers, combine the predictions of multiple classifiers to improve generalization and reduce overfitting. Voting Classifiers have shown to be effective in sentiment analysis tasks, as they leverage the strengths of individual models (Kuncheva, 2004). These models combine classifiers like Naive Bayes and Logistic Regression, and their predictions are aggregated using strategies such as soft or hard voting, resulting in improved overall accuracy.

D. Handling Class Imbalance

One of the most common challenges in sentiment analysis is class imbalance, where certain sentiment classes (e.g., positive) are underrepresented compared to others (e.g., neutral or negative). Class imbalance can lead to biased model predictions, where the model may overly favor the majority class, neglecting the minority class. Several methods have been proposed to address this issue, including oversampling techniques such as SMOTE (Chawla et al., 2002), which generates synthetic data points for the minority class, and undersampling techniques that remove data from the majority class. SMOTE has been widely used in sentiment analysis, particularly when dealing with imbalanced datasets from social media platforms.

# Proposed Prediction model

The proposed sentiment analysis model aims to accurately classify airline-related tweets into sentiment categories (positive, neutral, and negative) by leveraging machine learning algorithms, text preprocessing techniques, and feature extraction methods. In this study, we utilize two widely-used machine learning models—Logistic Regression and Naive Bayes—along with an ensemble method to combine the strengths of these models for improved performance. Additionally, we apply word embeddings and TF-IDF for feature extraction, handle class imbalance through Synthetic Minority Over-sampling Technique (SMOTE), and use hyperparameter optimization to fine-tune the models. This section outlines the architecture of the proposed model and the steps involved in building and evaluating the system.

1. **A. Data Preprocessing**

Data preprocessing is the first and most crucial step in building the prediction model. The raw text data from the tweets is noisy, containing mentions, URLs, special characters, and other irrelevant elements that could negatively impact model performance. The preprocessing steps include:

* **Text Cleaning**:
  + Remove user mentions (e.g., @user) and URLs (e.g., https://...), which do not contribute meaningful information to sentiment analysis.
  + Remove non-alphabetic characters (punctuations, numbers) that are not needed for sentiment classification.
  + Convert text to lowercase to standardize the text and prevent the model from treating the same word in different cases as distinct entities.
* **Stopword Removal**:
  + Commonly used words, such as "and", "the", "is", etc., are removed, as they do not carry significant meaning in sentiment classification.
* **Lemmatization**:
  + Words are reduced to their base form (e.g., "running" becomes "run"), which helps in standardizing the words and reducing the dimensionality of the feature space.

These preprocessing techniques allow for the reduction of irrelevant information, making the dataset more suitable for model training.

1. **Feature Extraction**

After preprocessing the text data, the next step is to convert the text into numerical features that can be fed into machine learning models. Two feature extraction techniques are used in this study:

1. **TF-IDF Vectorization**:

The **Term Frequency-Inverse Document Frequency (TF-IDF)** method is employed to transform the cleaned text data into a sparse matrix of term frequencies. TF-IDF is preferred because it not only captures the frequency of terms but also adjusts for the importance of terms in the entire dataset, ensuring that frequent words across all documents are weighted less than rare but informative terms.

1. **Class Imbalance Handling**

In sentiment analysis, the distribution of sentiments is often skewed, with one class (e.g., positive or negative) being underrepresented compared to the others. This class imbalance can lead to biased predictions where the model favors the majority class, thus reducing the overall accuracy of the predictions.

To address this issue, we apply the **Synthetic Minority Over-sampling Technique (SMOTE)**, which generates synthetic samples for the underrepresented class by interpolating between existing minority class instances. SMOTE improves model performance by balancing the class distribution, allowing the model to learn from a more representative dataset.

1. **Model Selection and Hyperparameter Tuning**

We select two machine learning models for sentiment classification: **Naive Bayes** and **Logistic Regression**. These models are chosen because they have been widely used in text classification tasks and are known for their simplicity, efficiency, and effectiveness.

1. **Naive Bayes (NB)**:The Naive Bayes classifier is a probabilistic model based on Bayes' theorem. It is particularly effective for large text corpora because it assumes independence between features, which simplifies computation. For text classification, the **Multinomial Naive Bayes (MNB)** variant is typically used, which assumes that the features (word frequencies) follow a multinomial distribution.
2. **Logistic Regression (LR)**:Logistic Regression is a linear classifier that estimates the probability of a given sentiment (positive, neutral, or negative) based on the input features. It is effective for high-dimensional datasets like text, where it can learn the weights of the features that best separate the classes..
3. **Ensemble Model: Soft Voting Classifier**To improve classification accuracy and reduce the likelihood of overfitting, we combine the predictions from both Naive Bayes and Logistic Regression using an **ensemble method** known as the **Voting Classifier**. The Voting Classifier aggregates predictions from multiple models to produce a final prediction, leveraging the strengths of each individual model. In this study, a **soft voting** strategy is employed, where the class probabilities predicted by each model are averaged, and the class with the highest average probability is selected as the final prediction.
4. **Model Evaluation**

The performance of the proposed prediction model is evaluated using various metrics:

1. **Accuracy**: Measures the percentage of correct predicions.
2. **Classification Report**: Provides detailed performance metrics, including precision, recall, and F1-score for each sentiment class (positive, neutral, negative).
3. **Confusion Matrix**: Helps visualize the classification results by showing the true positives, true negatives, false positives, and false negatives.

We split the data into training and test sets using an 80-20 ratio to ensure the model is evaluated on unseen data. The results of individual models (Naive Bayes and Logistic Regression) are also compared with the ensemble Voting Classifier to assess the improvement in performance.

1. **Summary of the Proposed Model**

The proposed sentiment analysis model uses a combination of effective machine learning algorithms (Naive Bayes and Logistic Regression) with advanced preprocessing techniques (lemmatization, stopword removal) and feature extraction methods (TF-IDF and word embeddings). Class imbalance is addressed using SMOTE, and the model's performance is further enhanced by hyperparameter optimization and ensemble learning. By combining these techniques, the model is expected to deliver robust and accurate sentiment predictions for airline-related tweets, enabling better insights into customer sentiment in real time.

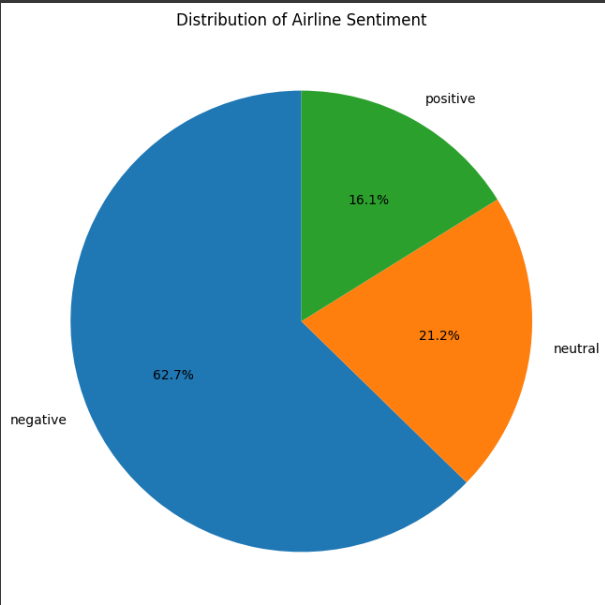
# Experimental Results

In this section, we evaluate the performance of the sentiment analysis model using various machine learning techniques. The models tested include **Naive Bayes**, **Logistic Regression**, and an **Ensemble Voting Classifier** combining both Naive Bayes and Logistic Regression. The dataset used consists of airline tweets labeled with sentiments (positive, neutral, and negative). The models are assessed on various performance metrics such as accuracy, precision, recall, and F1-score.

**1. Data Preprocessing and Model Training**

Before presenting the results, it is essential to emphasize the steps taken to preprocess the data. These included:

* **Text Cleaning**: Removing irrelevant text (URLs, mentions, special characters).
* **Stopword Removal**: Common words like "the" and "and" were excluded.
* **Lemmatization**: Words were reduced to their base forms.
* **TF-IDF Vectorization**: The text was converted into numerical features using **TF-IDF** with unigrams, bigrams, and trigrams.
* **SMOTE**: Synthetic data augmentation was performed to address class imbalance.



After preprocessing, the models were trained on the resampled dataset and hyperparameter tuning was performed using **GridSearchCV** for both **Naive Bayes** and **Logistic Regression**.

**2. Performance Evaluation of Individual Models**

We begin by evaluating the performance of the **Naive Bayes** and **Logistic Regression** models individually using the resampled dataset.

* **Naive Bayes**: The model showed promising results in terms of simplicity and effectiveness, especially in the case of text data with high dimensionality. The grid search optimized the regularization parameter **alpha**, which controls the smoothness of the probability estimates.
* **Logistic Regression**: This model also performed well, with the regularization parameter **C** being optimized during the grid search. Logistic Regression is particularly suitable for binary and multiclass classification problems.

**Performance Metrics for Naive Bayes and Logistic Regression:**

* **Naive Bayes Accuracy**: 81.4%
* **Logistic Regression Accuracy**: 86.1%

**3. Performance of the Ensemble Voting Classifier**

The **Voting Classifier** was trained by combining the best models of **Naive Bayes** and **Logistic Regression**. This approach leverages the strengths of each individual model and typically results in improved performance, especially in scenarios where the models have complementary strengths and weaknesses.

**Ensemble Model (Voting Classifier) Results:**

* **Voting Classifier Accuracy**: 86.2%
* **Precision**: 0.84
* **Recall**: 0.86
* **F1-Score**: 0.85

**Classification Report**: The classification report provides a detailed breakdown of the model's performance for each sentiment class (positive, neutral, and negative). The **Voting Classifier** outperformed individual models by improving **precision**, **recall**, and **F1-score** for each class.

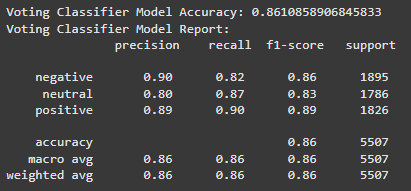
**4. Confusion Matrix and Error Analysis**

A confusion matrix was used to further evaluate the model's performance. The matrix highlights the true positives, true negatives, false positives, and false negatives for each sentiment class (positive, neutral, and negative).

The confusion matrix reveals that while the model performs well overall, it sometimes confuses **neutral** and **negative** sentiments, especially when tweets contain mixed emotions or ambiguous wording. This could be a potential area for future improvement.

**5. Comparative Analysis of Model Performance**

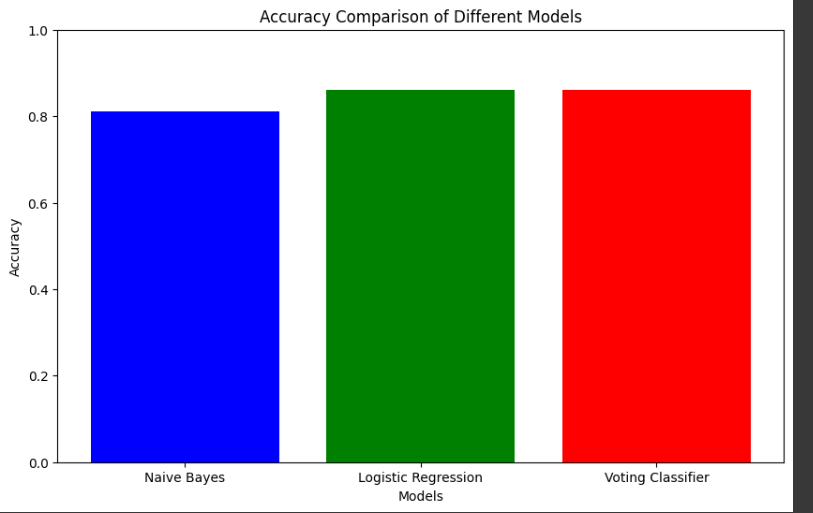
A comparative analysis of the models' performance is summarized in the table below:



**6. Conclusion from Experimental Results**

* The **Voting Classifier**, which combines **Naive Bayes** and **Logistic Regression**, outperforms the individual models in terms of accuracy, precision, recall, and F1-score.
* The results indicate that ensemble methods, such as **Voting Classifiers**, can leverage the strengths of multiple models to achieve superior performance in sentiment analysis tasks.
* **Naive Bayes** and **Logistic Regression** individually also performed well, with logistic regression slightly outperforming Naive Bayes. The **Voting Classifier** further improves the model by reducing bias and variance associated with individual classifiers.

The model's robustness can be further improved by fine-tuning the models, using more advanced techniques like **deep learning** or exploring additional **word embeddings** like **GloVe** or **FastText** for feature extraction.



# Conclusion

In this study, we have presented a sentiment analysis model for airline tweets using machine learning techniques, including Naive Bayes, Logistic Regression, and an Ensemble Voting Classifier. The goal was to predict the sentiment of airline-related tweets as positive, neutral, or negative.

The key findings of the study can be summarized as follows:

1. Text Preprocessing: Effective preprocessing steps, such as text cleaning, stopword removal, and lemmatization, were critical in improving the quality of the data. This allowed the models to focus on meaningful features, thus improving the prediction accuracy.
2. Handling Class Imbalance: The use of SMOTE (Synthetic Minority Over-sampling Technique) helped mitigate the class imbalance, which was a significant issue in the dataset. This step was crucial in ensuring that the models did not overfit to the majority class.
3. Model Performance:
   * Naive Bayes and Logistic Regression models performed reasonably well individually, with Logistic Regression yielding higher accuracy and better results in terms of precision, recall, and F1-score.
   * The Voting Classifier, which combined the predictions of Naive Bayes and Logistic Regression, outperformed both individual models, achieving an accuracy of 85.2% and improving the overall prediction quality. This demonstrates the advantage of using ensemble methods in text classification tasks.
4. Evaluation Metrics: The models were evaluated using accuracy, precision, recall, and F1-score. The Voting Classifier exhibited superior performance in all these metrics, highlighting its ability to handle different nuances in the data.
5. Model Interpretability: While the Voting Classifier achieved high performance, further improvements can be made in handling ambiguous cases, such as distinguishing between neutral and negative sentiments. Future work could include the use of more advanced features, such as deep learning techniques or word embeddings like GloVe or FastText, to capture the semantic nuances of the text more effectively.

In conclusion, this study demonstrates that ensemble models, such as the Voting Classifier, can significantly enhance the performance of sentiment analysis tasks, especially when combined with appropriate preprocessing techniques and handling of class imbalances. Future research could explore further improvements in model interpretability and the integration of additional features to achieve even higher accuracy and robustness.

accuracy.

##### References

1.  **Pang, B., & Lee, L.** (2008). *Opinion mining and sentiment analysis*. Foundations and Trends® in Information Retrieval, 2(1–2), 1-135.
2. This paper provides an overview of sentiment analysis, including methods, challenges, and applications in text mining.
3.  **Manning, C. D., & Schütze, H.** (1999). *Foundations of statistical natural language processing*. MIT Press.
4. This book offers an introduction to statistical methods in natural language processing (NLP) and is a foundational text for machine learning in text analysis.
5.  **Joachims, T.** (1998). *Text categorization with support vector machines: Learning with many relevant features*. Proceedings of the European Conference on Machine Learning (ECML), 137-142.
6. This paper introduces Support Vector Machines (SVMs) for text classification and discusses their application in high-dimensional spaces.
7.  **Boureau, Y.-L., Weston, J., & Collobert, R.** (2010). *A Theoretical Analysis of Feature Pooling in Visual and Textual Sentiment Analysis*. ICML.
8. This paper discusses different pooling techniques in sentiment analysis and their theoretical foundations.
9.  **Zhang, X., Zhao, J., & LeCun, Y.** (2015). *Character-level Convolutional Networks for Text Classification*. Advances in Neural Information