Journal of Intelligent Computing

DLINE JOURNALS

Print ISSN: 0976-9005
Online ISSN: 0976-9013

JIC 2025: 16 (3)
https://doi.org/10.6025/jic/2025/16/3/103-113

An Approach for Sentiment Analysis using Balanced Learning

Phuong Nguyen[1], Van-Huu Tran[1], The-Bao Nguyen[1], Hung Ho-Dac[1]
[1]Thu Dau Mot University, Binh Duong. Vietnam
{phuongnch, huutv, baont, hunghd}@tdmu.edu.vn

ABSTRACT

Sentiment analysis is a field of study in natural language processing (NLP). This study proposes an approach to data processing, feature extraction, data balancing, and training using four machine learning models: Multinomial Naïve Bayes, Random Forest, Support Vector Machine, and Decision Tree. Firstly, the dataset selected in the paper comprises the Internet Movie Database (IMDb), Twitter US Airline Sentiment (US Airline), and SemEval 2017. Second, data processing, feature extraction, and data balancing are employed to improve the accuracy of the training dataset. Specifically, data balancing is performed using the K-means SMOTE method, which has been proven effective for classification. Finally, the standard feature sets are applied to four machine learning models for training. The experimental results indicate that the SVM model achieves the highest accuracies of 89%, 96%, and 75% on the IMDb, US Airline, and SemEval 2017 datasets, respectively, compared to other state-of-the-art models.

Keywords: Feature Extraction, Sentiment Analysis, Balanced Learning, Support Machine Learning

Received: 13 March 2025, Revised 26 May 2025, Accepted 4 June 2025

1. Introduction

Sentiment analysis is a research field within the discipline of data mining [1] [2]. In [3], the authors suggested that feature selection plays a crucial role in developing effective and efficient sentiment analysis applications by choosing features that are pertinent and insightful. This not only enhances classifier performance but also decreases the dimension of the feature set. For the skewed dataset problem, [4] suggests integrating the k-means clustering algorithm with SMOTE (Synthetic Minority Over-sampling Technique) (K-mean SMOTE), which tackles certain limitations of other oversampling methods through a simple approach. The use of clustering allows the proposed oversampling to pinpoint and focus on regions of the input space where artificial data generation is most beneficial. This strategy seeks to correct both inter-class and intra-class imbalances while minimizing the risk of generating noisy samples. Sentiment analysis is complex mainly because of the long-range dependencies and the varied vocabulary found in texts. Many machine learning methods were proposed

for sentiment analysis, especially the models that are used parallel computing models consume a lot of computing resources, requiring high-performance computers, and large data set [2]. While Multinomial Naïve Bayes (Multi NB), Random Forest (RF), Support vector machine (SVM), Decision tree (DT) can train better on small datasets, and need fewer computing resources. This not only helps businesses reduce costs and training time but also ensures improved accuracy. The contributions of this paper are fourfold:

1) Discuss effective methods of data processing and feature extraction to improve accuracy, and focus the work.

2) Using K-means SMOTE technique on the imbalanced dataset to effectively perform feature extraction;

3) Experimenting with machine learning algorithms: Multi NB, Random Forest, SVM, Decision tree;

4) Contrast the experimental results with the latest algorithms to assess the proposed model's effectiveness.

The rest of the paper is organized as follows: Section 2 is related works encompassing nearly all aspects of data preprocessing, feature extraction, data augmentation, and machine learning algorithms. Section 3: Introduction to the proposal sentiment analysis model. Section 4 provides an analysis of three datasets, highlighting several issues. Section 5: Experimental Results. Finally, Section 6 concludes the paper.

2. Related Works

Text mining is a field of Natural Language Processing (NLP). It focuses on extracting useful information from unstructured or semi-structured text. However, unstructured text has the disadvantage of being diverse, complex, and extensive in storage, requiring a significant amount of computational resources. Therefore, data normalization to reduce the dimensionality of vectors and feature extraction becomes more difficult. Furthermore, to improve the efficiency of computational resource utilization and accuracy of machine learning algorithms, reducing the vector dimensionality and feature extraction quality is essential. [3] processed a dataset of 1959 tweets crawled on twitter with 977 negative tweets, and 982 positive tweets, removed sentiment labels, account names, links, word repetitions, and replaced misspelled words with correct words, then the data was trained with Support Vector Machine (SVM) which gave accuracy 83%, however the authors did not share further data loss after preprocessing, and the dimensionality of the feature vector before and after the data was processed. While [5] processed the Twitter US Airline Sentiment (US Airline) dataset with 14,640 tweets, including 3099 neutral tweets, 9178 negative tweets, and 2363 positive tweets, preprocessed the data by removing stop words, and punctuation marks; converted to lowercase, and original words using the Stemming algorithm; then the data was trained using SVM with an accuracy of 83%, demonstrating the effectiveness of this proposal, however, the authors have not compared to the case of the training dataset in a balanced data state. In addition, [6] also used the Twitter US Airline Sentiment (US Airline) dataset, the authors not only performed data preprocessing steps such as: removing punctuation, hash tags, stop words, HTML tags, mentions (e.g. @james), and emoticons; but also used the SMOTE algorithm to enhance the data to achieve balance; helping the training on SVM to increase the accuracy to 91%, however, the authors did not state the number of feature vector dimensions that were reduced, describe how to balance the dataset, and compare the accuracy between machine learning model.

3. Proposal Sentiment Analysis Model

The proposal model architecture is presented in Fig. 1.

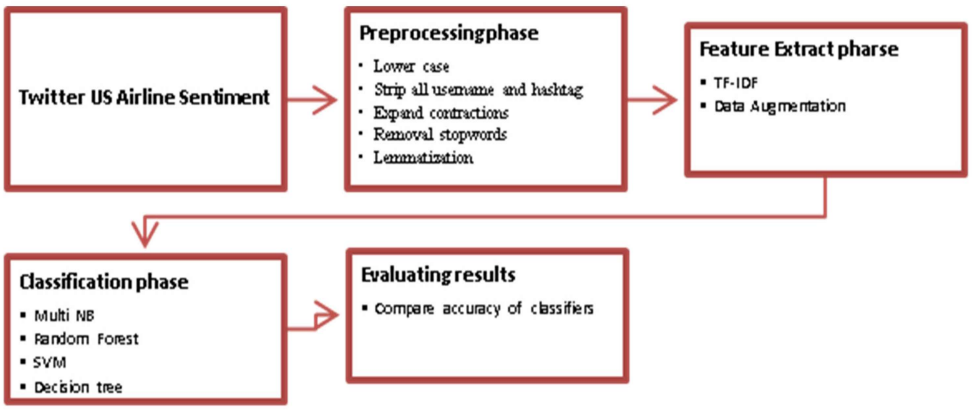


Figure 1. Proposed architecture for sentiment analysis

The system architecture is composed of four main modules, namely: (1) "input review"; (2) "pre-processing"; (3) "feature extraction"; (4) "classification models".

3.1 Data Preprocessing

The data needs to be subjected to certain refinements, such as lowercase, stripping all usernames and hash tags, sentence segmentation, expanding contractions, removing stop words, and lemmatisation [3]. The first step was converting all letters into lowercase to unify all words, since the machine is case sensitive. Dates, special characters like # and @, URLs, and meaningless words (such as b+, C", etc.) are identified and removed. Words such as 'not', 'wasn't', 'isn't',... have not been removed from the review because they expanded contractions. In the next step, we address stop words—insignificant words that can create noise when used as features in text classification. We removed common words such as 'a, 'an, 'are', 'as', 'at', etc. Additionally, lemmatization was applied to convert tokens into their base forms, reducing the number of word types or classes in the data. For instance, the words 'Running,' 'Ran,' and 'Runner' are all reduced to 'run.' We use lemmatization to enhance classification speed and efficiency, employing the Lemmatizer from the Python Natural Language Toolkit (NLTK) for its precision.

3.2 Feature Extractions

A significant challenge in text categorization is managing high-dimensional data. The sheer volume of terms, words, and phrases in documents leads to a substantial computational burden during the learning process. There are two methods that are described in the following:

Count Vectorizer (CV): Count vectorizer creates a sparse matrix, and in this sparse matrix, we store the count of all the words in our corpus. This is a straightforward yet effective method for converting text to numerical data.

Term frequency-inverse document frequency (TF-IDF): TF-IDF is a weighting metric commonly employed in information retrieval and NLP [7].

3.3 Data Augmentation

In the field of text analytics, several data augmentation techniques exist, including the use of Thesauri [8], text generation [9], and word embedding [10], among others. While combining the k-means clustering algorithm

with SMOTE addresses some of the drawbacks of other oversampling techniques using a straightforward method, [4] By utilising clustering, this new oversampling can identify and concentrate on areas in the input space where creating data is most advantageous. In this paper, the K-means SMOTE model is used to augment data in imbalanced datasets.

```
Algorithm: Kmeans SMOTE
Input: X (set feature), y (set labels)
Output: X_balanced (set feature), y_balanced (set labels)
num_labels = Counter(y)
K = len(num_labels)
clusters = MiniBathKmeas(X, k)
Step 1: cluster the set feature
Minority_class = ⦰
num_majority = max(num_labels.values)
for k, v∈ num_labels.items:
ratio = y/num_majority
if ratio < 1 then:
minority_class.append(cluster(k))
Step 2: identify minority class clusters
for m ∈ minority_class do:
avgminorityDistance(m) = mean(euclidean(m))
densityFactor = count(m) / avgminorityDistance(m)
sparsityFactor = 1 / densityFactor (m)
sparsity = sum(sparsityFactor (m))
samplingWeight(m) = sparsityFactor(m)/ sparsity
Step 3: apply SMOTE within each minority class cluster
samples = ⦰
for m ∈ minority_class do:
num_samples = (num_majority - count(m))*samplingWeight(m)
samples = samples + SMOTE(m, num_samples, knn)
for i ∈ range(num_samples) do:
y.append(m.labels)
Step 4: combine the original feature with the synthetic samples, and return the
balanced feature
X_balanced = combine(X, samples)
y_balanced = y
return X_balanced, y_balanced
```

3.4 Classification Sentiment Model

After using the K-means SMOTE model to augment data in imbalanced datasets, we utilise the following models: Multinomial Naïve Bayes (MultiNB), Random Forest (RF), Support Vector Machine (SVM), and Decision Tree (DT). Four machine learning models are used based on the scikit-learn library in Python.

4. Dataset

In this work, three datasets are used, namely Internet Movie Database, Twitter US Airline Sentiment, and *SemEval* 2017.

Dataset	Number of reviews			
	Negative	Neutral	Positive	Total
IMDb	25000	0	2500	50000
US Airline	9178	3099	2363	14640
SemEval 2017	3972	5937	2375	12284

Table 1. Summary of datasets

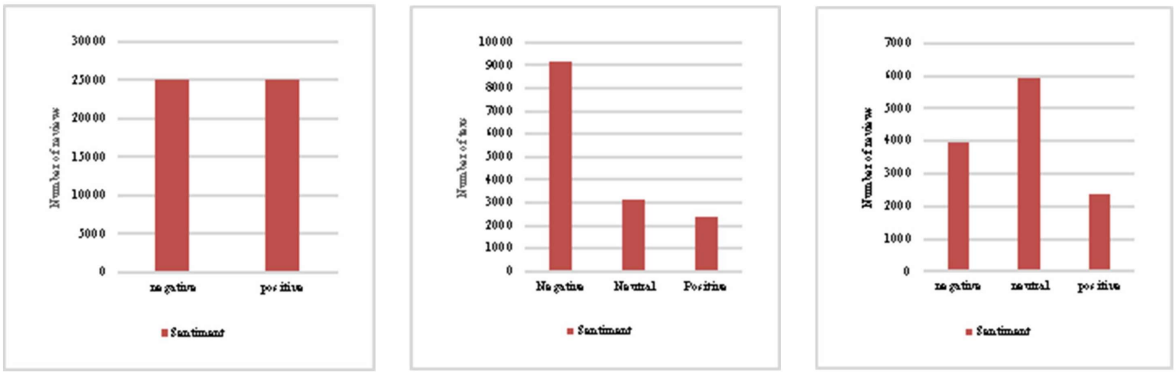
Stanford's Large Movie Review Dataset serves as the data for the sentiment analysis task, published by [12]. This dataset is commonly referred to as the Internet Movie Database (IMDb) dataset. The IMDb dataset is a binary sentiment analysis dataset comprising 50,000 reviews from IMDb. In this dataset, the number of reviews labeled "positive" and "negative" is equal.

A sentiment analysis project focused on issues related to major U.S. airlines. Twitter data, collected in February 2015, by CrowdFlower[1]. This dataset is known as the Twitter US Airline Sentiment (US Airline) dataset. Contributors were tasked with classifying tweets as positive, negative, or neutral. They also identified specific negative reasons, such as "late flight" or "rude service". The detailed information is in Table 2.

The *SemEval* 2017 datasets are extensively utilized in NLP research to benchmark the performance of different algorithms on tasks such as sentiment analysis and textual similarity. SemEval-2017 Task 4[2] involves the primary task of classifying tweets on a scale from 2-point to 5-point. This main task is further divided into five subtasks. In this context, subtask A involves message polarity classification, which means that given a message, the task is to determine whether it expresses positive, negative, or neutral sentiment. For this work, subtask A is performed with messages in English.

However, two of the three datasets are imbalanced, as illustrated in Figure 2. To begin with, the US Airline dataset is notably imbalanced, with the Negative class containing the highest number of tweets, at 9,178. In contrast, the other courses have even fewer tweets—60% for Neutral and 68% for Positive. The second dataset is *SemEval* 2017, with the highest count of 5,937 tweets classified into the Neutral class, while the remaining classes have lower proportions—33% for the Negative class and 60% for the Positive class. For these imbalanced datasets, machine learning models may over fit, even though Multi NB, SVM, DT, and RF algorithms can be adjusted to address this issue. However, the model's accuracy may still be compromised.

[1] https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment
[2] https://alt.qcri.org/semeval2017/task4/

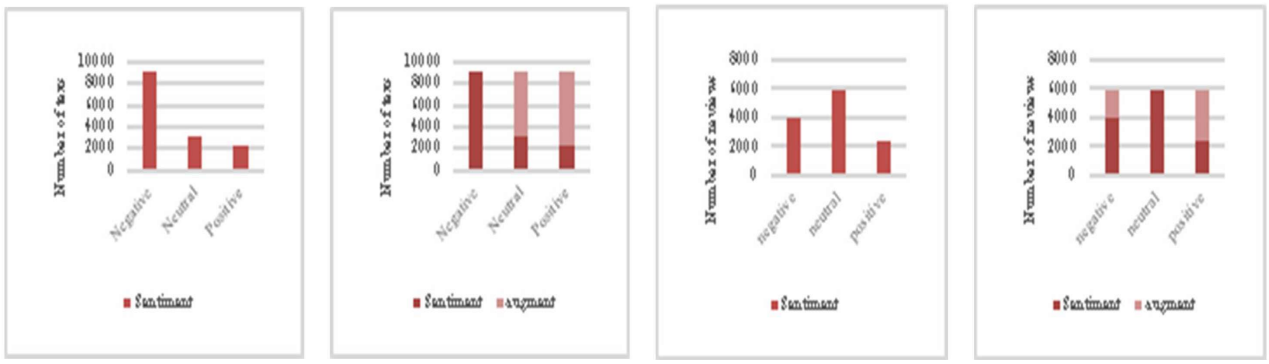


(a) (b) (c)

Figure 2. The information of a) IMDb, b) US Airline, and c) *SemEval* 2017 dataset

5. Experimental Results

As discussed in Section 4, the IMDb dataset is balanced, whereas the US Airline and SemEval 2017 datasets are imbalanced. Therefore, these datasets were initially affected by noise and had an imbalance between training data classes, which impacted the accuracy of machine learning models. The proposed K-means SMOTE algorithm was applied to balance the minority classes in these two datasets, as shown in Figure 3.



(a) After (b) Before (c) After (d) Before

Figure 3. The data augment for a), b) US Airline, and c), d) SemEval 2017 dataset

Figure 3 shows that for the US Airline dataset, if the Negative class is taken as the standard, the Neutral class is augmented by about 66%, and the Positive class by about 74%, Figure 3.b. Similarly, for the *SemEval* 2017 dataset, if the Neutral class is taken as the standard, the Negative class is augmented by about 33% and the Positive class by about 60%, as shown in Figure 3 d. The primary objective of this study is to evaluate the effectiveness of classification performance after balancing the dataset. The evaluation is conducted using performance metrics such as accuracy and F1-score [2]. To validate the model, the dataset is divided into training and testing sets with an 80/20 split.

Dataset	US Airline				SemEval 2017			
	F1-score		Accuracy		F1-score		Accuracy	
Model	After aug.	Before aug.	After aug.	Before aug.	Before aug.	After aug.	After aug.	Before aug.
Multi NB	60.22	84.84	68.92	84.89	71.05	51.81	58.12	72.13
RF	73.84	89.04	76.57	88.97	75.85	57.53	59.55	75.77
SVM	79.98	88.76	80.73	88.70	76.66	62.27	62.65	76.14
DT	68.86	85.20	69.90	85.18	68.10	49.70	49.70	68.39

Table 2. The difference in accuracy and F1-score before and after data augmentation for the US Airline and *SemEval* 2017 datasets

Model	F1-score	Accuracy
Multi NB	86.37	85.91
RF	84.75	84.75
SVM	90.65	90.66
DT	71.82	71.82

Table 3. The accuracy and F1-score of the IMDb dataset

Tables 3 and 4 present the accuracy and F1-scores of the machine learning models across the datasets. The SVM model recorded the highest accuracy, achieving 76.14% on the SemEval 2017 dataset and 90.66% on the IMDb dataset. In contrast, the RF model attained the top accuracy of 88.97% on the US Airline dataset. In the F1-score column, the RF and SVM models exhibited similar performance on the US Airline, IMDb, and SemEval 2017 datasets. The RF model recorded scores of 89.04%, 84.75%, and 75.85%, while the SVM model achieved 88.76%, 90.65%, and 76.66%, respectively. This suggests that both RF and SVM remain effective for text datasets.

On the other hand, these results were achieved by increasing the sample size for the minority classes, thereby balancing the training dataset across classes [9]. This approach proved effective across all the models evaluated, resulting in an average accuracy increase of 13% and 16% (Figures 7 and 5) and an F1-score boost of 16% and 18% on the US Airline and *SemEval* 2017 datasets (Figures 4 and 6), respectively. Although the IMDb dataset is already balanced between classes, applying noise removal [12] further enhanced the accuracy of the SVM model, which reached 90.66%.

To further clarify the effectiveness of the method proposed in this paper, we consider some related approaches, as shown in Tables 5 and 6. In these evaluations, machine learning and deep learning models are assessed for their performance on the US Airline, IMDb, and *SemEval* 2017 datasets.

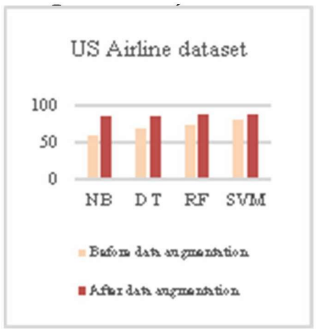


Figure 4.
The difference in F1-score before and after data augmentation for the US Airline dataset

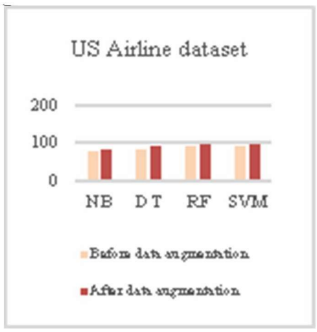


Figure 5.
The difference in Accuracy before and after data augmentation for the US Airline dataset

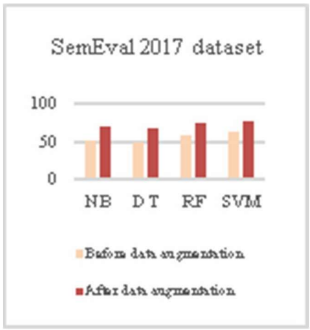


Figure 6.
The difference in F1-score before and after data augmentation for the SemEval 2017 dataset

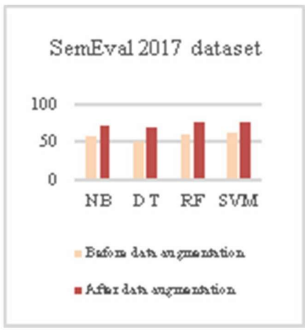


Figure 7.
The difference in Accuracy before and after data augmentation for the SemEval 2017 dataset

Model	US Airline		IMDb	
	F1-score	Accuracy	F1-score	Accuracy
[2] Logistic Regression	72	80.5	90	87
[2] AdaBoost	65	74.59	77	77
[2] K-nearest neighbors	60	68.41	83	83
[13] LSTM	69	77.56	88	88
[13] GRU	72	78	85	85
[14] BiLSTM	70	77	86	86
[1] CNN-LSTM	69	75	86	88
[2] RoBERTa-BiLSTM	80.73	80.74	92	92
Random Forest	89.04	88.97	84.75	84.75
SVM	88.76	88.70	90.65	90.66

Table 4. US Airline, IMDb dataset comparison with other approaches

Table 5 presents the performance of machine learning (Logistic Regression, AdaBoost, K-nearest neighbors), and deep learning models (LSTM, GRU, BiLSTM, and RoBERTa-BiLSTM) alongside RF and SVM models on the US Airline and IMDb datasets. On the US Airline dataset, the RoBERTa-BiLSTM model [2] achieved an accuracy of 80.74% and an F1-score of 80.73% when trained on the imbalanced dataset with Neutral and Positive classes. In comparison, the Logistic Regression model [2] achieved 80.5% accuracy and a 72% F1-score. However, with the application of our method and data augmentation, the RF and SVM models see an approximate 9% boost in

both accuracy and F1-score. Similarly, on the IMDb dataset, the RoBERTa-BiLSTM model [2] achieves the highest accuracy and F1-score, both at 92%. However, training this model demands a high-performance computing system. In comparison, the LR model [2] delivers a respectable accuracy of 87% and an F1-score of 90%, while the proposed SVM model achieves 90.66% accuracy and a 90.65% F1-score, which is only slightly, about 1.4%, lower than the RoBERTa-BiLSTM model [2].

Model	**F1-score**	**Accuracy**
[15] LSTMs-CNNs	67.7	65.1
[15] LSTM-Attention	68.5	65.8
[16] BERT	59.31	63.37
[17] BERTweet	72.8	72
[18] SVM	68.4	68.4
[18] RoBERT retrain on Twitter	74.2	74.2
Random Forest	75.85	75.77
SVM	76.66	76.14

Table 5. SemEval 2017 comparison between other approaches

Table 6 shows that the *SemEval* 2017 Task 4 dataset poses a challenge for training machine learning models. [18] used the SVM model to achieve an accuracy and F1-score of 68.4%. In contrast, deep learning models performed better, with the highest results achieved by the RoBERT a model, which was retrained on Twitter data [18], yielding an accuracy and F1-score of 74.2%. However, these studies have not employed data balancing methods like ours, which makes the proposed model in this paper effective. The RF and SVM models achieved accuracies of 75.77% and 76.14%, respectively, along with F1-scores of 75.85% and 76.66%.

6. Conclusion

This paper proposes an approach to data processing, feature extraction, data balancing, and training using four machine learning models. The method begins with data augmentation via K-means SMOTE, which helps generate more lexically similar samples and oversample minority classes, making it suitable for small datasets.The experimental results show that the Random Forest and SVM model not only saves computational resources but also deliver superior performance in sentiment analysis across three datasets, outperforming other advanced techniques. Looking ahead, these datasets are planned to undergo testing using the Long ShortTerm Memory (LSTM) model.

References

[1] Jain, P. K., Saravanan, V., Pamula, R. (2021). A hybrid CNN-LSTM: A deep learning approach for consumer sentiment analysis using qualitative user-generated contents. *ACM Transactions on Asian and Low-Resource Language Information Processing*, *20*(5), 1–15.

[2] Rahman, M. M., Shiplu, A. I., Watanobe, Y., Alam, M. A. (2024). RoBERTa-BiLSTM: A context-aware hybrid model for sentiment analysis. *arXiv preprint arXiv:2406.00367*, 1–18. , arXiv:2406.00367

[3] Go, A., Bhayani, R., Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford University*, 1–12.

[4] Last, F., Douzas, G., Bacao, F. (2017). Oversampling for imbalanced learning based on K-means and SMOTE, 1–19. arXiv:1711.00837

[5] Saad, A. I. (2020). Opinion mining on US airline Twitter data using machine learning techniques. In: *2020 16th International Computer Engineering Conference (ICENCO)* (pp. 59–63).

[6] Shitole, A. S., Vaidya, A. S. (2023). Machine learning based airlines tweets sentiment classification. *International Journal of Computer Applications*, *185*(20), 32–35.

[7] Baeza-Yates, R., Ribeiro-Neto, B. (2011). *Modern information retrieval* (2nd ed., pp. 68–74). Addison Wesley.

[8] Zhang, X., Zhao, J., LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28 (NeurIPS 2015)* (pp. 1–9).

[9] Douzas, G., Bacao, F., Last, F. (2018). Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Information Sciences*, *465*, 1–20.

[10] Wang, W. Y., Yang, D. (2015). That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 2557–2563).

[11] Madhuri, D. K. (2019). A machine learning based framework for sentiment classification: Indian Railways case study. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, *8*(4), 441–445.

[12] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., Potts, C. (2022). Sentimental analysis for e-commerce website. In *10th IEEE International Conference on Emerging Trends in Engineering & Technology - Signal and Information Processing (ICETET-SIP-22)* (pp. 1–4). IEEE.

[13] Hossen, M. S., Jony, A. H., Tabassum, T., Islam, M. T., Rahman, M. M., Khatun, T. (2021). Hotel review analysis for the prediction of business using deep learning approach. In *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)* (pp. 1489–1494). IEEE.

[14] Garg, A., Kaliyar, R. K. (2020). PSent20: An effective political sentiment analysis with deep learning using real-time social media tweets. *In: 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)* (pp. 1–5). IEEE.

[15] Rosenthal, S., Farra, N., Nakov, P. (2017). SemEval-2017 Task 4: Sentiment analysis in Twitter. *arXiv preprint arXiv:1912.00741*, 1–17.

[16] Das, R. K., Pedersen, D. T. (2024). SemEval-2017 Task 4: Sentiment analysis in Twitter using BERT. *arXiv preprint arXiv:2401.07944*, 1–5.

[17] Nguyen, D. Q., Vu, T., Nguyen, A. T. (2020). BERTweet: A pre-trained language model for English tweets. *arXiv preprint arXiv:2005.10200*, 1–6.

[18] Barbieri, F., Camacho-Collados, J., Neves, L., Espinosa-Anke, L. (2020). TweetEval: Unified benchmark and comparative evaluation for tweet classification. *arXiv preprint arXiv:2010.12421*, 1–7.