Turkish Journal of Mathematics and Computer Science

TR index

Aim & Scope

The aim of Turkish Journal of Mathematics and Computer Science (TJMCS) is to bring together research papers in different areas of applied and pure mathematics as well as applications of computer science and various areas of science and technology. The journal will also publish a limited number of proceedings of conferences. These proceedings will be fully refereed and adhere to the normal standards of the journal.

The Turkish Journal of Mathematics and Computer Science is a peer-reviewed, open-access journal that publishes original research and review articles in all areas of mathematics and computer sciences. Subject matters cover pure and applied mathematics, fuzzy theory, theoretical computer science, algorithms, scientific computing, artificial intelligence, use and application of mathematics and mathematical knowledge in natural science, engineering, medicine and the social sciences.

Coverage includes (Mathematics)

· Algebra

· Algebraic Geometry

· Category Theory

· Complex Analysis

· Control Theory and Optimization

· Differential Equations

· Differential Geometry

· Discrete Mathematics

· Dynamical Systems and Ergodic Theory

· Functional Analysis

· Geometry

· Mathematical Logic and Foundations

· Mathematical Physics

· Number Theory

· Numerical Analysis

· Operator Theory

· Probability Theory and Statistics

· Real Analysis

· Topology

Coverage includes (Computer Science)

· Algorithm

· Artificial Intelligence

· Bioinformatics

· Computational Complexity

· Computer Architecture

· Computer Graphics

· Computer Network and Internet

· Database Systems

· Dependable Computing

· Distributed Computing

· Formal Methods

· Grid Computing

· High-Performance Computing

· Human-Computer Interaction

· Image Processing

· Information Security

· Knowledge-Based Systems

· Natural Language Processing

· New Computational Models

· Pattern Recognition

· Software Engineering

Turkish Journal of Mathematics and Computer Science

CURRENT ISSUE

Volume: 15 Issue: 2, 12/31/23

YEAR: 2023

Articles

Research Article

1. Some New Identities for Fuzzy Fibonacci Number

Merve GÜNEY DUMAN *

Page : 212-217 PDF

Research Article

2. On Double Laplace-Shehu Transform and its Properties with Applications

Mona HUNAIBER Alı AL-AATI *

Page : 218-226 PDF

Research Article

3. On a Topological Operator via Local Closure Function

Ayşe Nur TUNÇ * Sena ÖZEN YILDIRIM

Page : 227-236 PDF

Research Article

4. New Results on a Partial Differential Equation with General Piecewise Constant Argument

Marat AKHMET Duygu ARUŠASLAN ÇİNÇİN Zekeriya ÖZKAN *

Page : 237-246 PDF

Research Article

5. Existence for a Nonlocal Porous Medium Equations of Kirchhoff Type with Logarithmic Nonlinearity

Uğur SERT *

Page : 247-257 PDF

Research Article

6. Spectral Properties of Discontinuous Boundary Value Problem with Transmission Conditions and Manypoint Singularities

Tevhide BALTÜRK Mustafa KANDEMİR *

Page : 258-269 PDF

Research Article

7. Ricci Solitons of Three-Dimensional Lorentzian Bianchi-Cartan-Vranceanu Spaces

Murat ALTUNBAŞ *

Page : 270-276 PDF

Research Article

8. New Conservative Schemes for Zakharov Equation

Ayhan AYDIN * Bahaa Ahmed Khalaf SABAWE

Page : 277-293 PDF

Research Article

9. Four New Sequence Spaces Obtained from the Domain of Quadruple Band Matrix Operator

Mustafa Cemil BİŞGİN *

Page : 294-311 PDF

Research Article

10. On the Semi-Analytical Solutions for the Kudryashov-Sinelshchikov Dynamical Equation Arising in Mixtures of Liquid and Gas Bubbles Without Neglecting of Heat Transfer and Viscosity

Emre AYDIN İnci ÇİLİNGİR SÜNGÜ *

Page : 312-325 PDF

Research Article

11. Investigating the Laplace Transform Method's Efficiency to a Simple Engineering Problem

Hüseyin DEMİR İnci ÇİLİNGİR SÜNGÜ * İbrahim KELES

Page : 326-333 PDF

Research Article

12. Revisiting Probabilistic Relation Analysis: Using Probabilistic Relation Graphs for Relational Similarity Analysis of Words in Short Texts

Dima ALNAHAS * Abdullah ATEŞ Ahmet Arif AYDIN Barış Baykant ALAGÖZ

Page : 334-354 PDF

Research Article

13. On Tangent Bundles of Submanifolds of a Riemannian Manifold Endowed with a Quarter-Symmetric Non-metric Connection

Mohammad Nazrul Islam KHAN * Lovejoy DAS

Page : 355-364 PDF

Research Article

14. Skin Lesion Classification Using Convolutional Neural Network and ABCD Rule

Ezgi KESTEK Mehmet Emin AKTAN * Erhan AKDOŠAN

Page : 365-374 PDF

Research Article

15. Some Curvature Conditions on 3-Dimensional Quasi-Sasakian Manifolds Admitting Conformal Ricci Soliton

Müslüm Aykut AKGÜN * Bilal Eftal ACET

Page : 375-381 PDF

Research Article

16. Pseudoparallel Invariant Submanifolds of a Para-Sasakian Manifold

Tuğba MERT * Mehmet ATÇEKEN

Page : 382-390 PDF

Research Article

17. Quasi-Concircular Curvature Tensor on Generalized Sasakian Space-Forms

Rana Pratap Singh YADAV * Bhagwat PRASAD

Page : 391-402 PDF

Research Article

18. Solution of Integral of the Fourth Power of a Finite-Length Exponential Fourier Series

Hesham SHARKAS *

Page : 403-406 PDF

Research Article

19. A Modified Quadratic Lorenz Attractor in Geometric Multiplicative Calculus

Bugce EMİNAGA TATLİCİOGLU *

Page : 407-413 PDF

Research Article

20. On Polynomial Space Curves with Flc-frame

Mustafa DEDE *

Page : 414-422 PDF

Research Article

21. A Note on the Composition of a Positive Integer whose Parts are Odd Integers

Busra AL * Mustafa ALKAN

Page : 423-432 PDF

Research Article

22. Tubular Surfaces According to a Focal Curve in E^3

Abdullah YILDIRIM *

Page : 433-442 PDF

Research Article

23. Face-Counting Identities and Derivatives of Face Polynomials

Hossein TEİMOORİ FAAL *

Page : 443-448 PDF

Research Article

24. Social Media User Opinion Analysis Using Deep Learning and Machine Learning Methods: A Case Study on Airlines

Ömer Ayberk ŞENCAN * İsmail ATACAK

Page : 449-463 PDF

Research Article

25. On Curvatures of Semi-invariant Submanifolds of Lorentzian Para-Sasakian Manifolds

Ramazan SARI İnan ÜNAL *

Page : 464-469 PDF

Turkish Journal of Mathematics and Computer Science

Journal Boards

Editorial Board

Editor in Chief

Ömer Akın, TOBB University of Economics & Technology, Turkey

Editor

Mustafa Özkan, Gazi University, Turkey

Hasan Hüseyin SAYAN, Gazi University, Turkey

Editor Assistant

Abdulhamit KÜÇÜKASLAN, Pamukkale University, Turkey

Fatih YILMAZ, Ankara Hacı Bayram Veli University, Turkey

Nurettin DOĞAN, Selçuk University, Turkey

Editorial Board

Abdon Atangana, University of the Free State, South Africa

Adnan Tercan, Hacettepe University, Turkey

A. Sinan Sertöz, Bilkent University, Turkey

Alireza Khastan, Institute for Advanced Studies in Basic Sciences, Iran

Anjan Biswas, Delaware State University, USA

Bayram Şahin, Ege University, Turkey

Elena Navarro, Embry-Riddle Aeronautical University, Germany

Emrah Kılıç, TOBB University of Economics & Technology, Turkey

Erkan Afacan, Gazi University, Turkey

Gary F. Birkenmeier, University of Louisiana at Lafayette, USA

Hüseyin Demir, Ondokuz Mayıs University, Turkey

Ian McAndrew, Embry-Riddle Aeronautical University, UK

I. Burhan Turksen, TOBB University of Economics & Technology, Turkey

İshak Altun, Kırıkkale University, Turkey

İsmail Kömbe, Istanbul Ticaret University, Turkey

Mohammad Mehdi Rashidi, Tongji University, China

Necdet Bildik, Celal Bayar University, Turkey

Ramiz M. Aliguliyev, Azerbaijan National Academy of Science, Azerbaijan

Şahin Emrah, Ankara University, Turkey

Süleyman Tosun, Hacettepe University, Turkey

Tahir Khaniyev, TOBB University of Economics & Technology, Turkey

Turkish Journal of Mathematics and Computer Science

Abstracting/Indexing

- Google Scholar
- TUBITAK ULAKBIM TR Dizin
- Asos Index

Journal Home Page

About

Aim & Scope

Submit a Manuscript

Journal Boards

Statistics

Writing Rules

Ethical Principles and Publication Policy

Price Policy

Indexes

Archive

Contact

Abstracting/Indexing

Turk. J. Math. Comput. Sci.
15(2)(2023) 449–463
© MatDer
DOI : 10.47000/tjmcs.1368430

Social Media User Opinion Analysis Using Deep Learning and Machine Learning Methods: A Case Study on Airlines

Ömer Ayberk Şencan[1,*], İsmail Atacak[1]

[1]*Department of Computer Engineering, Faculty of Technology , Gazi University, 06560, Ankara, Türkiye.*

Received: 29-09-2023 • Accepted: 31-10-2023

Abstract. The rapid surge in social media usage has augmented the significance and value of data available on these platforms. As a result, analyzing community sentiment and opinions related to various topics and events using social media data has become increasingly crucial. However, the sheer volume of data produced on social media platforms surpasses human processing capabilities. Consequently, artificial intelligence-based models became frequently employed in social media analysis. In this study, deep learning (DL) and machine learning (ML) methods are applied to assess user opinions regarding airlines, and the effectiveness of these methods in social media analysis is comparatively discussed based on the performance results obtained. Due to the imbalanced nature of the dataset, synthetic data is produced using the Synthetic Minority Over-Sampling Technique (SMOTE) to enhance model performance. Before the SMOTE process, the dataset containing 14640 data points expanded to 27534 data points after the SMOTE process. The experimental results demonstrate that Support Vector Machines (SVM) achieved the highest performance among all methods with accuracy, precision, recall, and F-score values of 0.79 in the pre-SMOTE (imbalanced dataset). In contrast, Random Forest (RF) obtained the best performance among all methods, with accuracy, precision, recall, and F-score values of 0.88 in the post-SMOTE (balanced data set). Moreover, experimental findings demonstrate that SMOTE led to performance improvements in ML and DL models, ranging from a minimum of 3% to a maximum of 24% increase in F-Score metric.

2020 AMS Classification: 68T07, 68T50

Keywords: Social media, sentiment analysis, deep learning, machine learning.

1. Introduction

The airline industry has evolved into a substantial global sector, pivotal in facilitating travel and connecting people worldwide. With the increasing demand for air travel in the past two decades, airline companies have expanded their operations, networks, and services, resulting in significant growth in the industry [11]. Regarding the competitiveness of this industry, airlines, like many other organizations, service providers, and companies, also consider customer feedback as valuable information to improve their services [8]. Understanding the importance of customer satisfaction, airlines have established various mechanisms to obtain passenger feedback. They use various techniques, such as online surveys and social media platforms to gather feedback on different aspects of their services. The data collected using these techniques is often used for sentiment analysis to improve their understanding of customers' opinions about the service they get. Most of this data is collected from social media platforms like Twitter, Instagram, and Facebook.

*Corresponding Author
Email addresses: oayberksencan@gazi.edu.tr (Ö. A. Şencan), iatacak@gazi.edu.tr (İ. Atacak)

In today's interconnected world, social media platforms have become an integral part of our daily lives by changing and affecting the way we communicate, share information about our lives, and express our opinions on specific topics. Recent research and industry projections suggest that active social media users worldwide will reach approximately 5.17 billion by 2024 [23]. These statistics emphasize the importance of social media platforms in our daily lives. Among these platforms, Twitter stands out as a popular microblogging platform that enables its users to post messages, also known as tweets. Furthermore, Twitter's capability for users to share visual content in pictures and videos enhances the richness of the platform, making it one of the most commonly used social media platforms. With the increasing popularity of social media platforms, data produced by users has also grown dramatically. This data can be used in a wide range of application fields, including security, politics, and marketing, for various tasks such as recommendation [20], sentiment analysis [22], topic detection [7], community detection [21] and popularity detection [3]. This enormous volume of user-generated content on Twitter has created a valuable resource for researchers and organizations to gain information about the public's sentiments on a given event, trends, and opinions on a wide range of topics. Considering that this emerging data has many people's opinions, the value of making this content manageable and understandable to understand what the public thinks about specific topics naturally increased.

Given the magnitude of the produced data, processing such data on social media platforms using solely human power is not possible. Therefore, researchers have developed methods to process this data and achieve meaningful results within an acceptable period. It is now possible to categorize each document and its contents, extract their characteristics, and summarize the material due to these "Document Classification" procedures [14]. Text Sentiment Classification, which can be seen as a sub-branch of Document Classification techniques, is one of the most widely used Natural Language Processing methods [29]. Indeed, sentiment refers to expressing a positive or negative opinion, thought, emotion, or feeling conveyed by the sentiment holder, which can be an individual user or a collective entity [4]. In order to comprehend the user's sentiments about the subject through the evaluated text, Sentiment Analysis is a crucial technique. In Sentiment Analysis, users' opinion on the topic is examined under three categories: positive, negative, or neutral [17, 27]. Various methods have been employed for sentiment analysis on social media platforms. These methods include a wide range of techniques, such as natural language processing, ML methods, and lexicon-based approaches [25]. These methodologies enable researchers to extract information from social media content classify and quantify the sentiment from the vast amount of text data generated on social media. This enables a deeper understanding of public opinion and significantly contributes to enhanced insights regarding public perception of a specific topic.

In this study, a comprehensive analysis is conducted utilizing widely employed ML and DL methods to perform sentiment analysis on tweets about six major airline companies in the U.S.A. This study aims to evaluate and compare the performance results of these methods to identify the most effective approach in accurately determining the sentiment expressed by social media users towards the airlines. By leveraging a diverse set of ML techniques such as Naïve Bayes (NB), SVM, RF, and Logistic Regression (LR), and DL techniques such as Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), this study aims to assess the effectiveness of given methods in sentiment analysis. We used accuracy and F1 measure performance metrics to evaluate the methods presented here. The findings of the study contribute to advancing the understanding of sentiment analysis in the context of social media. The rest of the paper is organized as follows: Section 2 broadly explains the techniques utilized to find the sentiment analysis from social media content. The materials and methods used in this study to conduct a sentiment analysis are given in Section 3. In Section 4, the results of the experimental studies are presented. Finally, Section 5 concludes the research.

2. Sentiment Analysis Techniques

Sentiment analysis is a valuable approach that can reveal a particular text's emotional context. For this reason, a large number of studies have been conducted in this field in the last decade, and many innovative approaches have been introduced. This section provides a broad overview of the previously mentioned methods.

ML methods are the main approaches used to perform the sentiment analysis task, and most of them implement supervised or semi-supervised learning to accomplish this task [18]. The process of sentiment analysis using relevant methods relies on training models with labeled data and testing the training models on unseen data. Here, the labeled data consists of text samples previously annotated by human operators with sentiment labels such as Positive, Neutral, or Negative. The annotations reflect the sentiment or emotional orientation of the given text. Researchers have utilized

different methods to extract the sentiment from the given text using ML such as NB, Maximum Entropy, SVM, and K-means [1, 13].

In the study conducted by Khairnar and Kinikar [13], they used various ML methods such as NB, Maximum Entropy, and SVM. Regarding the experimental results that they obtained, SVM was found to be the most effective method for text categorization, outperforming the other methods. Heba et al. [12] analyzed sentiments expressed in travelers' feedback using that airline company. The research study concludes that better performance results can be achieved by employing appropriate features and utilizing data oversampling techniques. Furthermore, this study highlights that the common problem of imbalanced datasets can be addressed by utilizing these techniques, and over-fitting can be prevented. The researchers have utilized AdaBoost, Decision Tree (DT), Linear SVM, NB, RF, K-Nearest Neighbors (K-NN), and Kernel SVM on the WEKA tool to extract the sentiment polarity of the sentences using the "Twitter Airline Sentiment" dataset. Using the DT algorithm, the researchers achieved an accuracy rate of over 85%. Using the IMDB Movie Review dataset, the study made by Pang et al. [19] was employed unigrams and SVM to extract the sentiment from the textual data. Through their approach, they achieved an accuracy rate of 82.9%, indicating that SVM has shown better performance than the other methods used in the same study, namely Maximum Entropy and NB. This study has established the baseline for many other authors after it in terms of the methodology. In another study on sentiment analysis, Liu et al. [16] developed a multi-class sentiment classification method. They used four popular feature selection methods to find the crucial parts of the text; after, they applied ML techniques such as DT, NB, SVM, Radial Basis Function Neural Network (RBFNN), and K-NN. They used three different datasets and validated their models using 10-fold cross-validation. According to the experimental results, they achieved over 80% accuracy rate with the Support Vector Machine algorithm.

Recently, we have seen DL methods being used successfully in sentiment analysis. In this context, in Qahtani and Abdulrahman's study on sentiment analysis [2], they used not only two ML methods including NB and LR, but also four DL methods containing CNN, Bidirectional Encoder Representations from Transformers (BERT), A Lite BERT (ALBERT) and XLNET. They used the "Twitter U.S. Airline Sentiment" dataset obtained from the Kaggle platform, which contains 14,640 tweets in 15 columns [6]. During the data preprocessing, the researchers removed the columns that mostly contained null values from the dataset and removed the "tweet_id" column, which did not affect the result. The authors of this paper attained an F1-Score of 0.9827 for binary-class classification and an F1-Score of 0.8943 for multi-class classification using ALBERT as the model. Furthermore, the researchers achieved an F1-Score of 0.8283 using the NB algorithm in combination with Spacy and Trigram for ML.

Table 1. Performance results of previous research

Study	Model	Performance Metric	Performance Result
[1]	LSTM-GRU	F-Score	0.96
		Accuracy	0.97
[12]	SVM	Accuracy	0.82
[16]	SVM	Accuracy	> 0.80
[2]	ALBERT	F-Score	0.89
[6]	NB	F-Score	0.82

3. Materials and Methods

This section elaborates on the dataset used in the study, providing detailed information and an in-depth analysis of the ML and DL methods employed in the research.

3.1. **Proposed Method.** The proposed method in this study contains five steps, namely: Data Collection, Preprocessing of the data, Data Splitting, Application of ML and DL methods and Evaluations. Figure 1 depicts the flowchart outlining the steps involved in this study's proposed approach. As understood from the flowchart in the figure, the Twitter airline sentiment dataset used in this study is obtained through the Kaggle Platform. In order to be able to feed the obtained text data into ML and DL models as input, this text data first needs to be cleaned. In this context, URLs, mentions (used for tagging other users), hashtags ('#'), punctuation marks, numerical data, and stop words representing

words that do not impact the sentence's sentiment in English language are removed from the text data. Subsequently, the cleaned data is divided into 75% for training and 25% for testing. The prepared data is used to train various ML and DL models. Finally, the performance values obtained by the trained models are evaluated using performance evaluation metrics, including Accuracy, Precision, Recall, and F-Score. Based on these evaluation results, a detailed comparative analysis is conducted among the trained models.

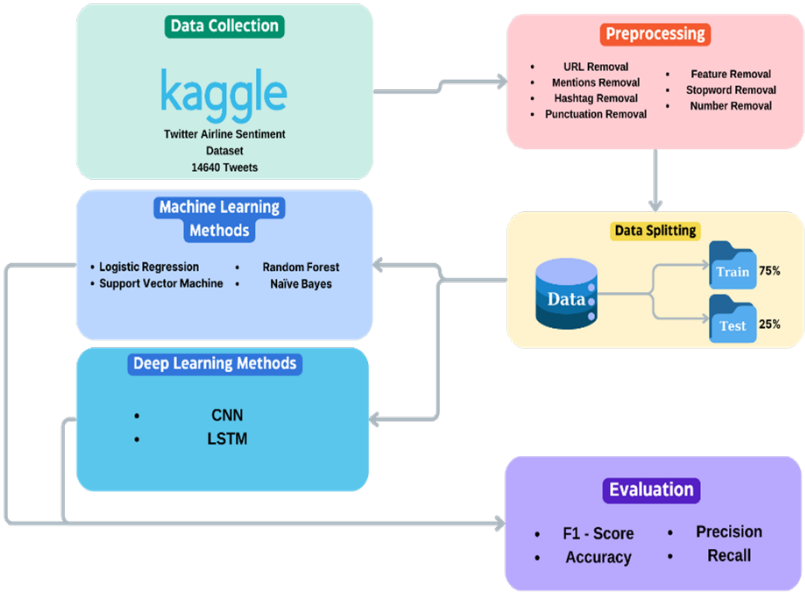

FIGURE 1. Flowchart of the proposed method

3.2. **Data Collection.** In the data collection phase, which is the initial step of the application, the "Twitter Airline Sentiment" dataset is obtained from the Kaggle platform. This dataset encompasses tweets shared by airline customers regarding six major airline companies operating in the United States. The emotional polarity of the given text is also included in the dataset, categorized as Positive, Negative, and Neutral. The dataset comprises 14,640 instances, representing a comprehensive repository of sentiments expressed by air transportation passengers. The distribution of instances in the dataset for each airline company is provided below in Figure 2.

After analyzing the numerical distribution of these airline companies in the dataset, a significant data imbalance becomes apparent. However, given that the airline company is not considered a factor influencing the sentiment of the given text in the dataset, no measures have been taken to address the situation. The number of instances for each sentiment category is shown in Figure 3 below. Based on the value counts of the polarity classes, the number of negative instances significantly outweighs the number of positive instances. Approximately 62% of the dataset comprises negative reviews. As a result, it can be concluded that the polarity data exhibits an imbalanced distribution.

Furthermore, Figure 3 provides insights into the distribution of positive, negative, and neutral reviews across different airlines. Upon analyzing the data provided in Figure 4, it becomes evident that an imbalance exists for United Airlines, U.S. Airways, and American Airlines. Specifically, these airlines' negative reviews are nearly three times higher than the combined count of positive and neutral reviews. On the other hand, when considering Virgin America, Southwest, and Delta, the distribution of reviews appears to be more balanced than the airlines mentioned earlier. The number of reviews for these airlines shows a relatively equal distribution across positive, negative, and neutral sentiments.

3.3. **Preprocessing.** During the preprocessing phase, certain features from the dataset are excluded based on two criteria: high prevalence of null values and their lack of impact on the final result. Table 2 presents an overview of the features excluded from the dataset with an explanation of these features.

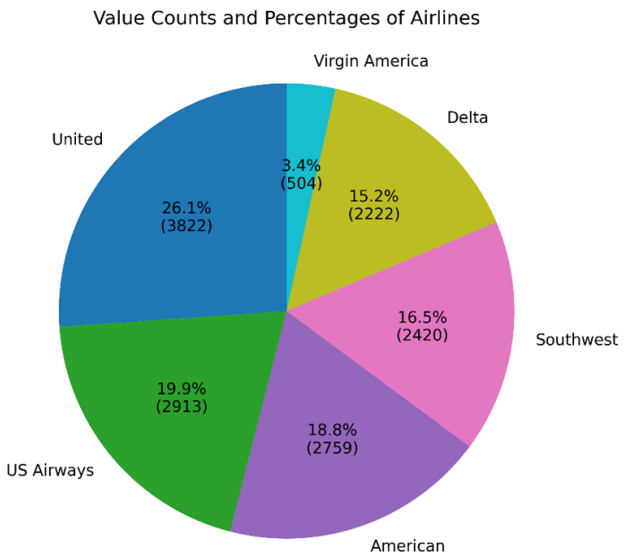

FIGURE 2. The distribution of instances in the dataset for each airline company

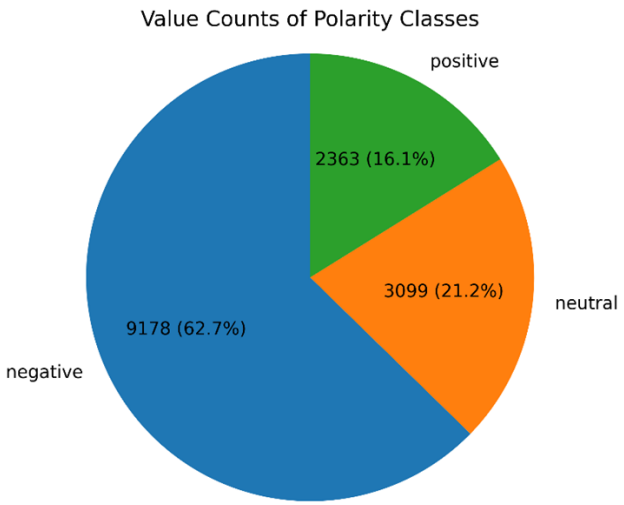

FIGURE 3. The number of instances for each sentiment category

Following the exclusion of irrelevant features, the raw version of the dataset has been obtained. Table 3 provides an example of the dataset in its raw form before undergoing any preprocessing steps.

As a part of the preprocessing step, the Natural Language Toolkit (NLTK) library, available in Python, has been employed to clean the text data in the dataset. The following processes have been sequentially applied using the NLTK toolkit:

- Removal of URLs from the text.
- Removal of mentions (references to other users) from the text.
- Removal of hashtags (only the symbol, not the accompanying text) from the text.
- Elimination of punctuation marks from the text.

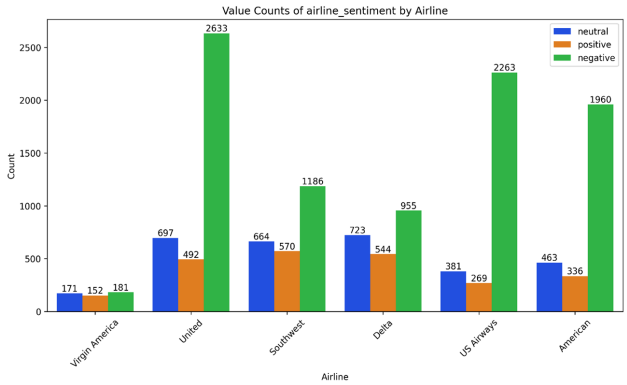

FIGURE 4. The distribution of sentiments by airline company

TABLE 2. The removed features and their explanations

Feature	Explanation
tweet_id	A unique identifier for each tweet.
airline_sentiment_confidence	The confidence level or certainty associated with the assigned sentiment label.
negativereason	The reason cited for a negative sentiment if applicable, such as customer service, flight delays, etc.
negativereason_confidence	The confidence level associated with the assigned negative reason.
airline	The airline company mentioned in the tweet.
airline_sentiment_gold	The manually assigned sentiment label for the tweet (used for the gold standard evaluation).
name	The Twitter username of the person who posted the tweet.
negativereason_gold	Manually assigned gold standard label for the specific negative sentiment reason.
retweet_count	The count or number of times the tweet has been retweeted by other Twitter users.
tweet_coord	The coordinates (latitude and longitude) associated with the location from where the tweet was posted.
tweet_created	The timestamp or date and time when the tweet was created or posted.
tweet_location	The user-provided location information mentioned in their Twitter profile or specified for that particular tweet.
user_timezone	The time zone of the user who posted the tweet, representing the local time zone at their location.

- Removal of numerical values from the text.
- Elimination of stop words (commonly used words with little semantic significance) from the text.
- Conversion of all text to lowercase.

These steps ensure that text data is cleaned and standardized, making modeling and analysis easier.

The steps mentioned above are followed as they aim to mitigate the potential negative impact of unprocessed raw data on the learning process of models. By cleaning the text, we eliminate irrelevant elements that do not contribute to the sentiment analysis task. Starting with the removal of URLs, we exclude web addresses as they lack emotional

Table 3. Sample data with the corresponding label

Text	Sentiment
@VirginAmerica What @dhepburn said.	Neutral
@VirginAmerica plus you've added commercials to the experience... tacky.	Positive
@VirginAmerica I didn't today... Must mean I need to take another trip!	Neutral
@VirginAmerica it's really aggressive to blast obnoxious "entertainment" in your guests' faces & they have little recourse	Negative
@VirginAmerica and it's a really big bad thing about it	Negative

value and do not provide meaningful sentiment information. Similarly, mentions that refer to other users are removed since they do not carry sentiment value alone. Regarding hashtags, we clean them by removing only the '#' sign while preserving the accompanying text. This is done because hashtags can contain sentiment-related information and may significantly influence the overall sentiment of the text. Punctuation marks, although aiding readability, do not possess inherent sentiment value. Hence, their removal ensures that the focus is solely on the textual content with sentiment implications.

Similarly, numerical values in the text are excluded as they do not contribute to sentiment analysis. By removing numbers, we ensure the focus remains on the textual content that carries sentiment-related information. Furthermore, stop words, commonly used words with little or no semantic value, are also removed. These words, such as "the," "is," and "and" do not provide significant insight into the sentiment of the text, and their removal helps to refine the dataset. After applying these steps, the text data is effectively cleaned, resulting in a dataset that is free from unnecessary elements and optimized for sentiment analysis. This cleaned version of the dataset serves as the foundation for the present study.

Feature Extraction. In order to utilize the raw text data as input for ML algorithms, it is necessary to convert the text into a numerical representation. This is because ML algorithms typically operate on numerical data. This conversion process is frequently accomplished using popular methodologies such as Bag of Word (BoW) and TF-IDF [1]. This study employed the TF-IDF method to transform the text data into a numerical representation that ML algorithms could utilize effectively.

Synthetic Minority Over-Sampling Technique (SMOTE). In classification problems, one of the most commonly encountered issues is the imbalanced distribution of classes within the dataset. This problem can be described as significantly more or fewer examples for one or more classes than for others [9]. Classes with limited examples can challenge for ML or DL models as recognizing these instances becomes more difficult. Consequently, this imbalance can adversely affect the performance of the models. In such cases, the accurate classification of examples belonging to classes with a limited number of instances tends to be far less frequent than for classes with a more significant number of data points.

One of the solutions developed to address this problem is using resampling methods. Resampling is a frequently employed and effective technique in problems involving imbalanced datasets. Its primary objective is to reduce the disparity in the number of examples between classes, thereby enhancing the performance of the proposed model. The SMOTE, as introduced by Chawla et al. in 2002, serves as a widely employed oversampling approach. It creates novel instances for the minority class by interpolating between each minority class instance and its k-nearest neighbors [5]. The data counts based on label classes before and after the SMOTE application are as follows:

Following the implementation of the SMOTE process, a significant transformation has occurred within the dataset, improving its balance. Specifically, the dataset, which initially consisted of 14,640 data points, has undergone substantial augmentation with the synthetic data, resulting in a significant increase in its total size, now comprising a more extensive set of 27,534 data points.

TF-IDF (Term Frequency-inverse document frequency). Term Frequency-Inverse Document Frequency (TF-IDF) is an important technique which is widely used in text mining applications to evaluate the importance of a given word in the

TABLE 4. Data Counts Based on Label Class Before and After SMOTE Application

Before SMOTE		After SMOTE	
Label Class	Data Count	Label Class	Data Count
Positive	9178	Positive	9178
Negative	3099	Negative	9178
Neutral	2363	Neutral	9178

entire corpus. In this technique, the text data is converted to numerical data by assigning a weight to the words in the corpus. The computation in the TF-IDF technique can be examined under two stages, namely, TF and IDF.

$$\mathrm{TF}(t,d) = \frac{(\text{Number of times term } t \text{ appears in document } d)}{(\text{Total number of terms in document } d)}. \tag{3.1}$$

In Equation (3.1), d represents the document and t represents a unique term in the document d.

$$\mathrm{IDF} = \log\left(\frac{\text{Number of documents}}{\text{Number of documents containing the term} t}\right). \tag{3.2}$$

$$\mathrm{TF-IDF\ Score}(t,d) = \mathrm{TF}(t,d) * \mathrm{IDF}(t). \tag{3.3}$$

Equation (3.2) shows the calculation for IDF and Equation (3.3) represents the TF-IDF score calculation using TF and IDF.

Data Splitting. After completing the preprocessing steps, the dataset is divided into training and test sets in preparation for the subsequent training of the ML and DL models. The data is split using the "train_test_split()" method from the sklearn library in Python, with a ratio of 75% allocated to the training set and 25% to the test set.

Application of Machine Learning and Deep Learning Methods. In this study, several artificial intelligence models based on ML and DL methods have been developed to accurately evaluate user opinions about airlines, and a performance comparison has been conducted to determine the model that produces the best evaluation results. In this context, this section provides detailed information about the ML and DL models, including NB, RF, LR, SVM, LSTM and CNN.

Naïve Bayes (NB). It is a popular ML algorithm based on Bayesian Network, widely employed in various tasks, particularly for classification problems, and acclaimed for its high performance in this domain [10]. The fundamental basis for classification in this algorithm is established through the probability calculations when addressing classification problems using this algorithm. It calculates the probability of an item belonging to each category and subsequently identifies the category with the highest probability as the outcome of the classification process. Within the NB model, attributes are considered entirely independent, each carrying equal weight.

$$P(\mathrm{Class}_k|x_1,x_2,\ldots,x_n) = \frac{P(x_1|\mathrm{Class}_k) * P(x_2|\mathrm{Class}_k) * \cdots * P(x_n|\mathrm{Class}_k)}{P(x_1) * P(x_2) * \cdots * P(x_n)}. \tag{3.4}$$

Equation (3.4) applies to the Bayes' theorem and estimates the likelihood of a class. The class with the highest probability is predicted using the equation. In this equation:

- $P(\mathrm{Class}_k|x_1,x_2,\ldots,x_n)$ represents the probability of the instance belonging to class k given the observed features such as $x_1,x_2,\ldots,x_n$.
- $P(x_1|\mathrm{Class}_k)$ represents the probability of observing the feature x_i given class k.
- $P(\mathrm{Class}_k)$ is the prior probability of class k.
- $P(x_i)$ represents the overall probability of observing feature x_i.

Using the equation, we can calculate the probability of an instance belonging to a particular class (Class_k) given the observed features $(x_1,x_2,\ldots,x_n)$. The equation combines prior probabilities, which represent the likelihood of each class independently, with likelihoods of observing each feature given a specific class. By comparing the posterior

probabilities for different classes, the algorithm assigns the instance to the class with the highest probability.

Random Forest (RF). This method, commonly used in classification problems, obtains its output by applying the majority voting criteria to the scores of various decision trees [24]. In other words, the prediction result is achieved by aggregating the outputs of decision trees through voting [1]. RF is an ensemble architecture that gains significance for handling diverse data by utilizing the bagging method to train each decision tree. This approach enables it to capture various patterns and relationships within the data, making it highly adaptable for different data types.

Logistic Regression (LR). It is a generalized linear model producing a continuous-valued output and employs the sigmoid function to convert the values obtained from linear regression into binary values [15]. By the application of the sigmoid function, the continuous output is transformed into a probability score ranging from 0 to 1. This probability score is then used to perform the classification, making LR a valuable method for solving the classification problems. The mathematical representation of LR can be given as follows:

$$P(Y = 1) = \frac{1}{1 + e^{-(b_0+b_1*x_1+b_2*x_2+\cdots+b_n*x_n)}}.$$

In this equation:

- $P(Y = 1)$ represents the probability of the dependent variable Y being in class 1.
- e is the base of the natural algorithm.
- $b_0, b_1, b_2, \ldots, b_n$ are coefficients to be estimated.
- $x_0, x_1, x_2, \ldots, x_n$ are the independent variables.

The logistic function $f(x) = \dfrac{1}{1 + e^{-x}}$ maps any real-valued number to the range $[0, 1]$, making it suitable for modeling probabilities. In LR, these coefficients are estimated using techniques like Maximum Likelihood Estimation (MLE). The model then makes predictions by applying the logistic function to the linear combination of the coefficients and the independent variables.

Support Vector Machines (SVM). This method, proposed by Vladimir Vapnik and et al. in 1963 [26], is a binary classification approach that aims to find results by defining the maximum-margin hyperplane for the given dataset. SVM's reliability stems from its optimization approach, which considers the minimization of both empirical risk and structural risk simultaneously [28]. Additionally, the hinge loss function employed by SVM contributes to its sparsity, further enhancing its generalization capability. These two key characteristics, collectively contribute to SVM's generalization ability. The equation of the decision function of SVM can be defined as:

$$f(x) = \text{sign}(w^t x + b).$$

In this equation:

- The sign is the sign function which outputs:

$$\begin{cases} +1 & \text{if } \ w^t x + b > 0 \\ -1 & \text{if } \ w^t x + b < 0. \end{cases}$$

- w represents the weight vector.
- x represents the feature vector.
- b is the bias term.

Long Short-Term Memory (LSTM). LSTM can be defined as a variant of Recurrent Neural Networks (RNN). RNN and RNN-based models have been widely used academically in recent years. The reason for this widespread use can be shown as the success of RNN-based models' high performance in natural language processing. Although very successful results have been obtained in natural language processing using the RNN, it has some shortcomings. The LSTM model, on the other hand, as a variant of RNN, solves the gradient loss (the loss of the previous information) problem in RNN [30]. In the LSTM model, the structure contains three different gates: the forget gate, the input gate, and the output gate. The forget gate decides which information in the given data is valid and which is useless. The useless information in the given data is ignored after this step. Then, the information enters the input gate, where the model

parameters are updated according to the new data. Lastly, the information enters the output gate, and the model output is decided in this gate using the information passed from the previous gates. Thanks to this multi-gate structure, the LSTM model can use the information from the previous steps while processing the data of the current step. This feature makes the LSTM model an effective tool that can be used in natural language processing problems.

Convolutional Neural Network (CNN). This method contains a multi-layered structure in which the layers are arranged sequentially. These layers are named the input layer, the hidden layer, and the output layer. The hidden layer has several convolutional layers, pooling layers, and a fully connected layer, which connects the hidden layer to the output layer. The tasks of the layers mentioned above are given in Table 5.

Table 5. The tasks of the layers in the CNN-based models

Layer	Task
Input Layer	The data enters the model via this layer.
Convolutional Layer	Using the filters to extract features from the input data.
Pooling Layer	In order to reduce the computational complexity, the spatial dimensions of the features of the data are reduced.
Fully Connected Layer	Connects the hidden layer with the output layer which allows the features extracted in the previous steps to pass to the output layer.
Output Layer	The features from the previous steps are used in this layer to either make predictions or to perform classification.

3.4. **Performance Metrics.** In classification problems, performance measurement metrics are employed to assess the proposed model's performance and objectively compare it to previously existing models. These metrics can characterize the relationship between the proposed model and the dataset used directly. Furthermore, they can be calculated based on a confusion matrix, which compares the predictions generated by the proposed model using the dataset against the real data. These metrics play a crucial role in evaluating the effectiveness and accuracy of the proposed model in solving the classification problem at hand. Nevertheless, previous research has demonstrated that performance measurement metrics solely quantifying the relationship between the dataset and the proposed model are insufficient in fully reflecting the model's performance from all perspectives. At this point, metrics obtained using the confusion matrix have been observed to address this gap effectively. These metrics derived from the confusion matrix offer a more comprehensive assessment of the model's performance by considering various aspects of its predictions and their alignment with the real data.

Performance metrics obtained using the confusion matrix rely on four fundamental parameters that result from various calculations of the values within this matrix. These parameters are named True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), respectively. TP represents the count of positively labeled data the proposed model successfully classified as positive. In contrast, TN represents the count of negatively labeled data that the model correctly classified as negative. On the other hand, FP indicates the number of instances that were actually negative but were incorrectly classified as positive by the model, and finally, FN represents the count of instances that were positive but were classified as negative by the model. Using these parameters, calculations for performance metrics such as accuracy, precision, recall, and F-Score for the proposed model can be conducted.

The first of these performance metrics, accuracy, emerges as a performance metric that facilitates the evaluation of the overall performance of a model. As one of the most widely used performance metrics in classification problems, accuracy is calculated by taking the ratio of the data points correctly classified by the model to the total number of data points. Accuracy measures the model's ability to classify data across all classes correctly and is a fundamental metric for assessing classification model performance. The formula used to calculate this metric is provided below:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}.$$

Precision focuses on the accuracy of positive predictions made by the proposed model, demonstrating its ability to classify positive instances among all instances predicted as positive correctly. Accordingly, the precision formula can be defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP}.$$

Alternatively referred to as sensitivity or the true positive rate, Recall assesses the model's capacity to detect all genuine positive instances within the dataset accurately. This metric proves invaluable when the primary objective is the reduction of false negatives. The mathematical representation for recall is provided below:

$$\text{Recall} = \frac{TP}{TP + FN}.$$

The F1-Score metric, also known as the F-Score, is a measure of the overlap between the model's predictions and the real data. This metric takes on a value of zero (0) when there is no overlap between the model's predictions and the real data and takes on a value of one (1) when there is a complete and comprehensive overlap. Consequently, when the F1-Score value obtained by the proposed model gets closer to 1, it indicates a higher level of success in addressing the specific problem or dataset under consideration. Finally, the F-Score, or the F1-Score, is a harmonized metric that balances precision and recall. It provides a single measure that considers both false positives and false negatives. The F-Score is calculated using the following formula:

$$\text{F-Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}.$$

4. Experimental Results and Discussion

In this study, ML and DL methods, including NB, RF, LR, SVM, CNN, and LSTM, were implemented through a program written in Python to assess user opinions related to airlines. Google Colab was chosen as the working environment here. The main reason for this choice is that Google Colab provides a convenient platform for creating Python-based projects and enables rapid deployment, especially in ML and DL applications, due to its inclusion of libraries for data preprocessing, deep learning, machine learning, and sentiment analysis, as well as its support for TPU usage in the online environment for users. The process, components, and parameters described in Section 3 were taken into consideration in the creation of models obtained using these methods. In addition to that, the values given in Table 6 were used as the hyperparameters of ML and DL methods.

The experiments were conducted under the conditions of 0.75 split. The performance results were acquired by applying the mentioned ML and DL models to the unbalanced dataset obtained from the Kaggle platform and the balanced dataset obtained from this dataset with SMOTE for the US Airlines. The results obtained by applying 3660 test data instances to ML and DL models trained with 10980 instances from the imbalanced dataset, which contains 14640 instances, are shown in Table 7.

When evaluating the results in the table, it can be seen that NB, LR, and SVM exhibit the best performance with an accuracy of 0.79, as well as recall metrics, while SVM achieves the highest performance among all models with values of 0.79 for accuracy, precision, recall and F-score metrics. On the other hand, LSTM has the lowest performance with an accuracy of 0.64, precision of 0.41, recall of 0.64, and an F-score of 0.50. Considering the numerical changes in the table, when we compare the performances of ML models with DL models, it can be concluded that ML models outperform LSTM and CNN DL models in terms of all metrics.

The results obtained by applying the ML and DL models, trained with 20650 instances on the balanced dataset containing 27534 instances, to the test data with 6884 instances are presented in Table 8. Upon initial examination of the performance metrics presented in the table, a notable observation emerges, indicating that, akin to the imbalanced dataset, machine learning (ML) methods exhibit superior performance relative to deep learning (DL) methods in this context as well.

For this dataset, RF method achieves the highest performance in terms of all metrics with the values of 0.88. Another ML method, SVM, produces the closest second result with the ones of 0.87 to RF's performance. The LSTM method obtains the lowest performance in this dataset, with an accuracy of 0.75, a precision of 0.74, a recall of 0.75, and an F-score of 0.74.

TABLE 6. The hyperparameters of ML and DL methods

Method	Hyperparameter	Value	Description
NB	alpha	1.0	Smoothing parameter used to handle the case where a certain term doesn't appear in the training data.
RF	n_estimators	500	The number of decision trees in the forest.
	random_state	42	The number set the seed ensuring that the same random decisions are made during the training process.
LR	solver	liblinear	The algorithm to use in the optimization problem.
	C	1.0	Inverse of regularization strength.
SVM	kernel	rbf	The kernel type to be used in the algorithm.
	C	1.0	Inverse of regularization strength
CNN	optimizer	adam	Method adjusting the internal parameters of the model in order to minimize the loss function during training.
	Loss Function	Categorical Crossentropy	Mathematical function that quantifies the difference between the predicted values of the model and the actual target values.
LSTM	optimizer	adam	Method adjusting the internal parameters of the model in order to minimize the loss function during training.
	Learning Rate Schedule	('lr') for the first 20 epochs, decrease by a factor of 0.1 for the remaining epochs	Adjusts the learning rate during training.

TABLE 7. The performance results of the ML and DL models for the imbalanced dataset

Method	Accuracy	Precision	Recall	F-Score
NB	0.79	0.78	0.79	0.76
RF	0.77	0.76	0.77	0.75
LR	0.79	0.78	0.79	0.78
SVM	0.79	0.79	0.79	0.79
LSTM	0.64	0.41	0.64	0.50
CNN	0.68	0.72	0.68	0.69

Table 8. The performance results of the ML and DL models for the balanced dataset

Method	Accuracy	Precision	Recall	F-Score
NB	0.82	0.82	0.82	0.82
RF	0.88	0.88	0.88	0.88
LR	0.81	0.82	0.81	0.81
SVM	0.87	0.87	0.87	0.87
LSTM	0.75	0.74	0.75	0.74
CNN	0.78	0.78	0.78	0.78

As a result of applying the SMOTE process, the obtained dataset indicates an improvement in the performance of all models. Notably, for the accuracy metric, an increase of 3% in NB, 11% in RF, 2% in LR, 8% in SVM, 11% in LSTM, and 10% in CNN has been observed. These results demonstrate increased accuracy ranging from 2% to 11% across all models. RF and LSTM have achieved the most significant improvement with 11% in the accuracy metric. Similarly, an increase ranging from 0.06 to 0.24 is evident when examining the F-Score metric. Similar to the accuracy metric, the performance of all models has also improved for this metric. The most substantial increase, with 24%, has been observed in the LSTM model. In conclusion, the results highlight that the application of the SMOTE process has the potential to significantly enhance the performance of models operating on imbalanced dataset.

The performance results of the model that achieved the best results among the models trained in this study, as well as the performance values of the models proposed in other studies in the literature, are provided in Table 9.

Table 9. Performance results of previous research and the most successful model in this research

Study	Model	Performance Metric	Performance Result
[1]	LSTM-GRU	F-Score	0.96
		Accuracy	0.97
[2]	ALBERT	F-Score	0.89
This Study	RF	F-Score	0.88
		Accuracy	0.88
[12]	SVM	Accuracy	0.82
[6]	NB	F-Score	0.82
[16]	SVM	Accuracy	> 0.80

Upon examining the performance results, it is evident that the RF algorithm, which yielded the most favorable outcome in this study, demonstrates a level of effectiveness that can be considered competitive with models utilizing DL methods like LSTM-GRU and NLP techniques such as ALBERT, as reported in previous studies. The researchers aim to expand this comparison by incorporating ensemble learning models, natural language processing models, and transfer learning models to provide a comprehensive performance comparison and analysis of all models used in this field.

5. Conclusion

Today, users often share opinions about a particular event, situation, or product on social media platforms in the form of textual comments. Evaluating these posts is vital in receiving customer feedback, conducting market research, developing products and services, and increasing marketing effectiveness. However, these comments on social media platforms constitute a large data stacks, and the human ability to analyze this data remains beyond their capacity in terms of time and evaluation. Therefore, evaluating these shares with artificial intelligence-based sentiment analysis methods instead of directly with humans has become an important necessity for social media users and companies that need to benefit from these analyzes on the above-mentioned issues. In this study, models including DL and ML

methods were developed to evaluate user opinions regarding airline companies, and their performances were compared with each other. Due to the unbalanced nature of the dataset, this dataset is augmented with the synthetic minority over-sampling technique to improve the model performance. Accuracy, precision, recall and F-score metrics were used for performance evaluation. The results obtained from the experimental studies showed that SVM achieved the best performance in all metrics for the imbalanced dataset, while RF achieved the best performance for the balanced dataset. In the balanced dataset, SVM had the best performance after RF. In both datasets, ML methods performed better than DL methods. LSTM, one of the DL methods, produced the lowest performance among all methods in both balanced and unbalanced datasets. When the SMOTE process was applied, significant performance improvements were observed in all models, highlighting the crucial role of this process in measuring the performance of models in imbalanced datasets. In our study, when we compared RF, which achieved the highest performance in the balanced dataset, with similar methods in the literature, it can be seen that RF is an approach that can compete with DL and natural language processing methods proposed in literature. In our future studies, we plan to develop simple hybrid models incorporating DL and ML-based methods to further enhance performance.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Authors Contribution Statement

The authors have been involved in the planning of the study, the creation of models, the conduct of experimental studies, and the analysis of results. Based on the obtained results, they have worked together on writing the article and giving it its final form. Both authors have read and approved the published version of the article.

References

[1] Aljedaani, W. et al., *Sentiment analysis on Twitter data integrating TextBlob and deep learning models: The case of US airline industry*, Knowledge Based Systems, **255**(2022), 109780.

[2] Al-Qahtani, R., Bint Abdulrahman, P.N., *Predict sentiment of airline tweets using ML models*, EasyChair, **5228**(2021).

[3] Atacak, İ., Şencan, Ö.A., *Mamdani ve Sugeno tip bulanık çıkarım sistemleri ile sosyal medya haber popülerliğinin tahmini*, Uluslararası Muhendislik Arastırma ve Geliştirme Dergisi, **14**(3)(2022), 303–320.

[4] Bibi, M. et al., *A novel unsupervised ensemble framework using concept-based linguistic methods and machine learning for twitter sentiment analysis*, Pattern Recognition Letters, **158**(2022), 80–86.

[5] Chaw, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., *SMOTE: Synthetic Minority Over-sampling Technique*, Journal of Artificial Intelligence Research, **16**(2002), 321–357.

[6] Figure Eight, Twitter US Airline Sentiment, (2023), Accessed: Oct 25, 2023, https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment.

[7] Garciai, K., Berton, L., *Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA*, Applied Soft Computing **101**(2021), 107057.

[8] Greer, C.R., Lei, D., *Collaborative innovation with customers: A review of the literature and suggestions for future research*, International Journal of Management Reviews, **14**(1) (2012), 63–84.

[9] Guan, H., Zhao, L., Dong, X., Chen, C., *Extended natural neighborhood for SMOTE and its variants in imbalanced classification*, Engineering Applications of Artificial Intelligence, **124**(2023), 106570.

[10] Guo, W., Wang, G., Wang, C., Wang, Y., *Distribution network topology identification based on gradient boosting decision tree and attribute weighted naive Bayes*, Energy Reports, **9**(2023), 727–736.

[11] Hasib, K. Md., Habib, Md. A., Towhid, N.A., Showrov, Md. I.H., *A novel deep learning based sentiment analysis of Twitter data for US airline service*, International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), (2021), 450–455.

[12] Heba, H., Aljarah, I., Al-Shboul, B., *Online social media-based sentiment analysis for US airline companies*, Proceedings of the New Trends in Information Technology (NTIT-2017), (2017), 176–181.

[13] Khairnar, J., Kinikar, M., *Machine learning algorithms for opinion mining and sentiment classification*, Citeseer, **3**(6)(2013), Accessed: May 25, 2023, https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=269d91e79049092bdf0651241d0d66830aa9fafc.

[14] Kong, J., Wang, J., Zhang, X., *Hierarchical BERT with an adaptive fine-tuning strategy for document classification*, Knowledge Based Systems, **238**(2022), 107872.

[15] Li, W.C., Jiang, L., *Learning from crowds with robust logistic regression*, Information Sciences, **639**(2023), 119010.

[16] Liu, Y., Bi, J.-W., Fan, Z.-P., *Multi-class sentiment classification: The experimental comparisons of feature selection and machine learning algorithms*, Expert Systems with Applications, **80**(2017), 323–339.

[17] Liu, F., Zheng, J., Zheng, L., Chen, C., *Combining attention-based bidirectional gated recurrent neural network and two-dimensional convolutional neural network for document-level sentiment classification*, Neurocomputing, **371**(2020), 39–50.

[18] Mercha, E.M., Benbrahim, H., *Machine learning and deep learning for sentiment analysis across languages: A survey*, Neurocomputing, **531**(2023), 195–216.

[19] Pang, B., Lee, L., Vaithyanathan, S., *Thumbs up? Sentiment classification using machine learning techniques*, Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, (2002), 79–86.

[20] Pavitha, N. et al., *Movie recommendation and sentiment analysis using machine learning*, Global Transitions Proceedings, **3**(1)(2022), 279–284.

[21] Qiu, J., et al., *GCC: Graph contrastive coding for graph neural network pre-training*, Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, New York, NY, USA: ACM, (2020), 1150–1160.

[22] Rodríguez-Ibánez, M., Casánez-Ventura, A., Castejón-Mateos, F., Cuenca-Jiménez, P.-M., *A review on sentiment analysis from social media platforms*, Expert Systems with Applications, **223**(2023), 119862.

[23] Statista, *Number of worldwide social network users 2027*, (2023), https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/, accessed Mar. 10, 2023.

[24] Svetnik, V., Liaw, A., Ton, C., Christopher Culberson, J., Sheridan, R.P., Feuston, B.P., *Random forest: A classification and regression tool for compound classification and QSAR modeling*, Journal of Chemical Information and Modeling, **43**(6)(2003), 1947–1958.

[25] Şencan, Ö.A., Atacak, İ., Doğru, İ.A., *Systematic literature review of detecting topics and communities in social networks*, Bilişim Teknolojileri Dergisi, **15**(3)(2022), 317–329.

[26] Vapnik, V.N., Lerner, A.Y., Recognition of patterns with help of generalized portraits, (1963).

[27] Wen, S. et al., *Memristive LSTM network for sentiment analysis*, IEEE Transactions on Systems, Man, and Cybernetics, **51**(3)(2021), 1794–1804.

[28] Wen-wen, G., Lv, Y., Jia-yu, Y., Wang, Z., Yuan-hai, S., *Fast support vector classifier with generalization-memorization kernel*, Procedia Comput Sci, **214**(2022), 55–62.

[29] Wilson, T., Wiebe, J., Hoffmann, P., *Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis*, Computational Linguistics, **35**(3)(2009), 399–433.

[30] Zhou, L., Zhao, C., Liu, N., Yao, X., Cheng, Z., *Improved LSTM-based deep learning model for COVID-19 prediction using optimized approach*, Engineering Applications of Artificial Intelligence, **122**(2023), 106157.