Knowledge-Based Systems 255 (2022) 109780

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Sentiment analysis on Twitter data integrating TextBlob and deep learning models: The case of US airline industry

Wajdi Aljedaani [a], Furqan Rustam [b], Mohamed Wiem Mkaouer [c], Abdullatif Ghallab [d], Vaibhav Rupapara [e], Patrick Bernard Washington [f], Ernesto Lee [g], Imran Ashraf [h,*]

[a] University of North Texas, TX, USA
[b] Department of Software Engineering, School of Systems and Technology, University of Management and Technology, Lahore, 54770, Pakistan
[c] Rochester Institute of Technology, NY, USA
[d] Faculty of Computing and Information Technology, University of Science and Technology, Sana'a, Yemen
[e] School of Computing and Information Sciences, Florida International University, USA
[f] Division of Business Administration and Economics, Morehouse College, Atlanta, GA, USA
[g] Miami Dade College, College of Engineering and Technology , Miami, FL, 33176, USA
[h] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, South Korea

ARTICLE INFO

Article history:
Received 30 January 2022
Received in revised form 22 August 2022
Accepted 23 August 2022
Available online 27 August 2022

Keywords:
Knowledge-based systems
Sentiment analysis
Lexicon-based approach
Machine learning
Natural language processing

ABSTRACT

Twitter being among the popular social media platforms, provide peoples' opinions regarding specific ideas, products, services, etc. The large amounts of shared data as tweets can help extract users' sentiment and provide valuable feedback to improve the quality of products and services alike. Similar to other service industries, the airline industry utilizes such feedback for determining customers' satisfaction levels and improving the quality of experience where needed. This, of course, requires accurate sentiments from the user tweets. Existing sentiment analysis models suffer from low accuracy on account of the contradictions found in the tweet text and the assigned label. From this perspective, this study proposes a hybrid sentiment analysis approach where the lexicon-based methods are used with deep learning models to improve sentiment accuracy. Experiments involve analyzing the impact of TextBlob on the classification accuracy of models as against the original annotations, considering that the probability of the false annotations cannot be overlooked. Furthermore, the efficacy of TextBlob against Afinn and VADER (Valence Aware Dictionary for Sentiment Reasoning) is also evaluated. The CNN (Convolutional Neural Network), LSTM (Long Short-Term Memory), GRU (Gated Recurrent Unit), and CNN-LSTM are deployed in comparison with state-of-the-art machine learning models. Additionally, the efficiency and efficacy of TF-IDF (Term Frequency-Inverse Document Frequency) and BoW (Bag of Words) are also investigated. Results suggest that models perform better when trained using the TextBlob assigned sentiments as compared to the original sentiments in the dataset. LSTM-GRU outperforms all models and previous studies with the highest 0.97 accuracy and 0.96 F1 scores. From machine learning models, the support vector classifier and extra tree classifier achieve the highest accuracy score of 0.92, with TF-IDF and BoW, respectively. Despite the good performance of the models using the TextBlob labels, TextBlob-based annotation cannot replace humans. Our stance is that with humans, bias, error-proneness, and subjectivity cannot be ignored; so we propose that the TextBlob-annotated labels can be used as assistance for human annotators where human annotators can wet the TextBlob-annotated dataset.

1. Introduction

Text mining is an emerging field of data mining, often used to extract helpful information from raw data. Today, around 2.5 quintillion bytes of data are generated daily [1] advocating the importance of text mining as a potential tool to extract meaningful information from such a large amount of data [2,3]. Text classification has emerged as an important research area, especially after the inception and explosive growth of social media platforms like Twitter, Facebook, Linked-In, etc. People share the information and views on such platforms which are analyzed to find criticism and appreciation regarding products and services. The analysis of sentiments using text is called the 'sentiment analysis and it has been deployed to extract users' reactions,

* Corresponding author.
E-mail address: imranashraf@ynu.ac.kr (I. Ashraf).

https://doi.org/10.1016/j.knosys.2022.109780
0950-7051/

reviews, opinions, and feedback for a service or product of a company [4]. The extracted sentiments are used to develop and improve policies to enhance product penetration and improve services for the customers. The widespread usage of these platforms generates informative data that can help extract meaningful information. In recent years, databases of social media have gained immense popularity due to their versatility and richness.

The feedback from social media platforms can be used for both commercial and industrial purposes. From the commercial perspective, it helps companies develop policies to gain customer attraction and revise present policies for increasing the acceptance of products and elevating the quality of service. For example, [5] states that political campaigns are designed and updated according to the political reviews as analyzed by the data from Twitter. Also, companies can use customers' sentiments regarding products for better decision-making to enhance the quality of products [6]. Industrial purpose, on the other hand, involves using the online reviews of numerous services and products to analyze customers' purchasing trends [7]. For example, [8] states that customers are more comfortable buying a product with a five-star rating compared to one with a four-star rating.

Similar to other service-providing organizations, the competition in the airline industry has been on the rise. Airlines aim to increase revenue by improving the offered services, developing advanced schemes and policies for upcoming years, and increasing the quality of customer satisfaction. Predominantly, airline companies use conventional customer feedback forms, which are not very user-friendly and are time-consuming [9]. Therefore social media platforms like Twitter serve a crucial role in these enhancements because the customer's reviews give valuable insights into the products [10,11]. The analysis of such reviews depends upon the expressions contained in the reviews. These reviews are high in volume and many experts are required for classification and analysis. Several machine learning classifiers have been developed to reduce human efforts in the classification of these reviews. These techniques still need improvements for increasing classification accuracy.

For the most part, existing studies utilize a machine learning approach where the annotated data is used for sentiment analysis and the focus is on improving the performance of models [12–14]. This study, on the other hand, proposes a novel hybrid method for sentiment analysis combining machine learning techniques with a lexicon-based model called TextBlob. Previously, a few approaches and predominantly machine learning models have been investigated to enhance the accuracy, and the characteristic of datasets have been overlooked. The probability of the false annotations cannot be overlooked as tweets tagged as positive may have neutral or positive sentiments instead. Therefore, utilizing the machine learning models with such data can influence the performance of the models negatively. Keeping in view such presumption, this study investigates the performance of using TextBlob as compared to the original dataset for sentiment analysis. Despite previous efforts to optimize the performance of the models through hyperparameter settings, model architecture optimization, preprocessing pipelines, and feature extraction and selection approaches, the models showed no further improvements or marginal improvements. This inspired us to look into the aspects related to the dataset. When we inspected the dataset, we found some inappropriate labels where the assigned labels are contradictory to the given text. The labels do not correspond to the sentiments given in the text. So, initial experiments were performed with a smaller dataset which was labeled using lexicon-based approaches as the dataset is already manually labeled. The difference in the performance of machine learning models inspired us to delve deep and perform an in-depth analysis.

This study proposes a framework to perform sentiment analysis on six US (United States) airline companies. Prior studies [13,15–17] utilized the same dataset with traditional machine learning techniques. In comparison, this study additionally adopts four deep learning models including CNN (Convolutional Neural Network), LSTM (Long Short-Term Memory), GRU (Gated Recurrent Unit), and CNN-LSTM augmented with lexicon-based technique TextBlob to investigate how the TextBlob method influences the accuracy of the classification models. Specifically, we address the following research questions in this study

RQ1: *Of the manual annotation and TextBlob annotation, which one is better and why?*

RQ2: *To what extent can the TextBlob method improve the models' accuracy for sentiment analysis?*

RQ3: *How efficient the models are when the sentiment analysis is performed using the TextBlob with respect to other lexicons?*

The main contributions of our work include

- A framework is proposed to obtain high classification accuracy of sentiment analysis for airline-related tweets using machine learning and deep learning techniques. For this purpose, a novel deep learning architecture LSTM-GRU is contrived that leverages the benefits of LSTM and GRU structures. The models are combined sequentially in a stack where LSTM deals with the appropriate patterns from data and GRU learns from those patterns to make predictions.
- A comprehensive analysis of the text classification problem is carried out to analyze the accuracy of models when used with TextBlob annotated data against the sentiments from the original annotations. In addition, the performance of VADER (Valence Aware Dictionary and sEntiment Reasoner) and AFINN is compared with TextBlob. A hybrid annotation approach is additionally investigated where TextBlob, Vader, and AFINN approaches are combined to analyze their efficacy for investigating the impact on the classification accuracy of machine learning and deep learning models. Annotation approaches are combined under majority voting criteria.
- Machine learning and deep learning models are analyzed regarding the performance where TF-IDF and BoW are used for feature extraction to train the machine learning models. Results are compared with several recent approaches in text classification.

The rest of this study is organized into five sections. A few important related studies are discussed in Section 2. A brief description of machine learning and deep learning models is given in Section 3 along with the architectural details of the proposed methodology. Results are discussed in Section 4 while Section 5 concludes this study.

2. Related work

The sentiment analysis can be categorized into lexicon sentiment analysis, machine learning-based sentiment analysis, and hybrid techniques. The lexicon sentiment analysis relies on the polarity of words in a given text. A lexicon is a library or a dictionary, comprising a large number of words that are ranked based on their polarity score. Predominantly, people use very informal vocabulary in the reviews that are not a part of lexicons. Therefore, experts emphasize the application of alternative methods for sentiment detection in the text. This notion gave rise to the second category of sentimental analysis techniques

which utilize machine learning methods. Models are trained on sample datasets and predictions are made on an unseen dataset afterward. The sentiment analysis can be formulated as a classification problem. For example, a document can be represented using a feature set [6]. Later, the document can be assigned a label concerning its polarity (i.e., neutral, positive, and negative) and then changed into a feature matrix.

Text classification holds a great potential to study sentiments, and many researchers have explored the progression of sentiment analysis by identifying emotions in the scripts [18,19]. Others have suggested that sentiment assessment techniques are devised by studying human reactions to particular experiences [20]. The use of machine learning practices involving NB (Naïve Bayes), ME (Maximum Entropy), and SVM (Support Vector Machine) for sentiment classification has also been analyzed [21]. For instance, the study [22] utilized NB, ME, and SVM on the IMDB (Internet Movie Database), which comprises movie reviews. The methodology is assessed using accuracy and recall processes. This study functioned as a model for several authors, and the same methods have been applied through several domains. Likewise, [23] analyzes sentiments from travelers' feedback for airline companies. The authors conclude that the best results can be obtained using appropriate features and data over-sampling. Furthermore, the skewed distribution of the classes discovered mostly in small datasets is decreased, devoid of over-fitting. The research findings demonstrate the convincing indication that the recommended model has greater classification accuracy when forecasting the three classes positive, negative, and neutral. The authors followed a similar methodology in [24] and presented a multiclass sentiment classification. The implementation of NB, DTC, radial basis function neural network, SVM, and k nearest neighbor is analyzed with 10-fold cross-validation.

In addition, [25] utilizes customers' responses to examine various aspects of airline services such as trustworthiness, happiness, etc. The trustworthiness is concluded through airline attributes, such as operational factors, competitive factors, and flyer programs. The study determines that the customer's quality of service is attributed to company repute, staff service, aircraft, frequent flyer program, and reliability. Kumar et al. introduced a novel approach for analyzing the sentiments from tweets [26]. The authors obtained the opinion words which are a mixture of adjectives, adverbs, and verbs, to find the sentiment. The corpus-based technique is applied to find the semantics of adjectives while the semantics of adverbs and verbs are obtained using the dictionary-based technique. The sentiment is then determined using a linear equation by incorporating emotion intensity. Hasan et al. [27] presented sentiment analysis applying a machine learning method. The divergence is observed using SentiWordNet, TextBlob, and WSD (Word Sense Disambiguation) sentiment analyzers.

Pandey et al. [28] proposed a meta-heuristic technique CSK which is centered on CS (Cuckoo Search) and K (K means). Since clustering plays an essential role in analyzing the perspectives and sentiments in consumer tweets, the study suggests a technique to find the optimal cluster head from the Twitter dataset. Results are promising with improved performance over traditional models. Several studies focus on emotion identification from the dialogues [29,30]. Similarly, [31] captured general semantics and structural semantics of words , [32] classified tweet text using graph convolutional network, and [33] leveraged the syntactic dependencies of the sentence. Zhao and Yu [34] proposed a pre-trained model BERT (Bidirectional Encoder Representations from Transformers) for the use of aspect-based sentiment analysis. The study conducted experiments using Chinese universities' MOOC (Massive Open Online Courses) platforms to obtain the embedding vectors of entities in the knowledge base as well as text in a vector space. While Chiong et al. [35] proposed 90 features to investigate the effect of sentiment lexicon to detect depression using tweets.

It is found that various classifiers have different capabilities for sentiment classification and different preprocessing techniques can be used to supplement various classifiers. For instance, the authors of [36] demonstrated that the choice of an applicable preprocessing method can provide improved classification results. The investigation using different preprocessing pipelines reveals that different preprocessing techniques perform a significant role in discovering the most delicate classification figures. In the same manner, several feature extraction practices have been established to improve classification precision. Text mining has numerous feature extraction techniques, but TF (Term Frequency), IDF (Inverse Document Frequency), TF-IDF, doc2vec, and word2vec are among the very frequently utilized feature extraction techniques [37]. The authors of [38] examined TF, IDF, and TF-IDF, along with linear classifiers, including LR, SVM, and perception, with a local language identification system. The TF-IDF weighting on characteristics confirms to overtake other practices when utilized with uni-and bi-grams. Likewise, [39] examined the use of three feature extraction methods with a neural network for the text analysis. The research indicates that TF-IDF supports the model to accomplish better precision. For less significant datasets, the combination of TF-IDF and LSA is appropriate to obtain comparable precision. For providing a comprehensive analytical overview of the discussed results, Table 1 summarizes these works.

3. Materials and methods

In this section, we describe the dataset used for sentiment analysis and it is visualized for initial analysis. Additionally, the proposed methodology for sentiment analysis is discussed.

3.1. Proposed approach

Fig. 1 shows the architecture of the methodology followed in this study. Data collection is the first phase of this study. The data of six US airlines are collected from Kaggle, containing 14 640 tweets. The second phase is data preprocessing, where the data is cleaned to reach the best performance of the selected classifiers. In the third phase, we applied the sentiment annotation using the Textblob library[1] which is a lexicon-based method. Data split and training is carried out in the fourth phase, followed by the feature engineering phase, which contains TF-IDF and BoW. The sixth phase involves the use of deep learning and machine learning algorithms, followed by data prediction. In the last phase, the performance of the trained models is evaluated.

3.2. Data collection

The Kaggle dataset, which contains tweets for six US airlines, is used in this study. The dataset name is 'twitter-airline-sentiment',[2] and it includes 14 640 tweets where each record is classified as positive, negative, or neutral accordingly. The number of samples for each company is shown in Fig. 2.

The number of tweets for each of the six companies is different and the distribution of positive, negative, and neutral tweets for each company is shown in Fig. 3. It shows that predominantly, the number of negative tweets is higher than both neutral and positive sentiments, and class distribution for US Airways, United airways, and American airways is highly imbalanced.

[1] https://textblob.readthedocs.io/en/dev/.

[2] https://www.kaggle.com/crowdflower/twitter-airline-sentiment.

Table 1
Summary of the studies discussed in the related work.

Study	Year	Purpose	Approach	Data source	Dataset	Techniques
[23]	2017	Tweet classification	LS, ML	Twitter	14,640 tweets	TF-IDF, SMOTE, AB, SVM, NB, RF, KNN,DT
[24]	2017	Performance of FS & ML	ML	Movie reviews	18,908 reviews	DF, CHI, IG, GR, DT,NB, SVM, RBFNN, KNN
[28]	2017	Sentiment analysis	CSK	Twitter	3965 tweets	K-means, cuckoo search
[27]	2018	Tweet classification	LS, ML	Twitter	6250 tweets	NB, SVM
[36]	2018	Text analysis & opinion mining	SentiWordNet, SenticNet, VS	Twitter	256 tweets	VSM, NB, KNN, silhouette coefficient
[38]	2018		ML	TOEFL	12,100 essays	TF-IDF, SVM, LR, Perceptron
[39]	2019	Text categorization	ML	alio.lt, skelbiu.lt	10,000 advertisements	TF-IDF, LDA, LSA, NN
[13]	2019	ML performance	ML	Twitter	14,640 tweets	SVM, DT, ET, GB, LR, FR, SGDC, TF, TF-IDF, word2vec
[17]	2021	Airlines sentiment analysis	ML, DL	Twitter	14,640 tweets	NB, LG, CNN, BERT, ALBERT, XLNET
[16]	2021	Airlines emotion analysis	ML, DL:	Twitter	14,640 tweets	Meta Data + TF-IDF + Trainable Embedding
[34]	2021	Aspect–sentiment relationship	DL	MOOC platform	9123 posts	KNEE, CG-BERT, R-GAT+BERT, BERT+Liner, BERT+SKG
[35]	2021	Depression detection	ML, DL	Twitter	22,191 tweets	LR, SVM, DT, MLP, BP, RF, AB, GB
[29]	2022	Dialogues emotions	DL	TV-series Friends	142,182 Dialogues	CLSTM, CNN, BERT BASE, DialogueRNN, KET, CANet, Sentic GAT VAD, Sentic GAT Hourglass, Sentic GAT Intensity
[30]	2022	Dialogues emotions	DL	TV-series Friends, IEMOCAP DATABASE	56,780 Dialogues	SMIN, SMIN – IIM, SMIN – CIM
[31]	2022	Words' semantics	DL	Twitter	18,744 tweets	ATAE-LSTM, RAM, AF-LSTM, CDT, ASGCN, InterGCN, BiGCN, R-GAT, RepWalk, DGEDT
[32]	2022	Tweet classification	ML, DL	Twitter	1342 tweets	Lasso, LR, SANT, SASS, GCN-Kipf, ChebNet, Stacking model, MLPs
[33]	2022	Aspect dependencies		Public reviews	13,348 reviews	MemNet, IAN, RAM, GCAE, IARM, MGAN, AOA, TNet-LF, TransCap, IACapsNet, TD-GAT, ASGCN-DT, ASGCN-DG, CDT, DGEDT, R-GAT, LCFS-ASC, Affective GCN, Sentic GCN, Sentic GCN-D, LSTM & BERT variants
Current	2022	ML performance with TextBlob	ML	Twitter	14,640 tweets	LR, RF, SVC, DT, ETC, GBC TF, TF-IDF, LSTM, CNN, CNN-LSTM, GRU

The information of different attributes is provided in Table 2. The attribute used to determine the sentiment is the original text posted on Twitter. This text is preprocessed and extracted features from it are used to train the models for sentiment prediction.

3.3. Data preprocessing

Data cleaning is done in full preprocessing to increase the learning performance of machine learning models. The preprocessing is performed using Python natural language toolkit [40]. Tweets include punctuation, stop-words, and a mixture of small and capital letters that may influence models' learning capacity. Fig. 4 shows the steps followed in the data preprocessing phase. In the beginning, the punctuation is removed from the tweets, e.g. [], () |,' #?. Additionally, Twitter allocated '@user' to each user is also eliminated during this process. Although punctuation makes a sentence readable, it impairs the models' ability to discriminate between punctuation and other characters [41]. Numbers are omitted as they do not have a major effect on sentiments.

All the text in the tweets is translated to lowercase after numeric elimination. This step is essential as the interpretation of the text is case-sensitive. Yang and Zhang [41] concluded that the probabilistic machine learning models count the occurrence of each word, for example, 'good' and 'Good' are assumed to be two separate words. Stemming is an effective preprocessing method since eliminating affixes from words and translating them into their root form improves model performance [42]. Words, for example, may have several variations in the text with the same meaning. For example, 'does' and 'doing' are changed forms of 'do'. In the current study stemming is carried out using Porter stemmer algorithms [43]. Lemmatization considers the context of

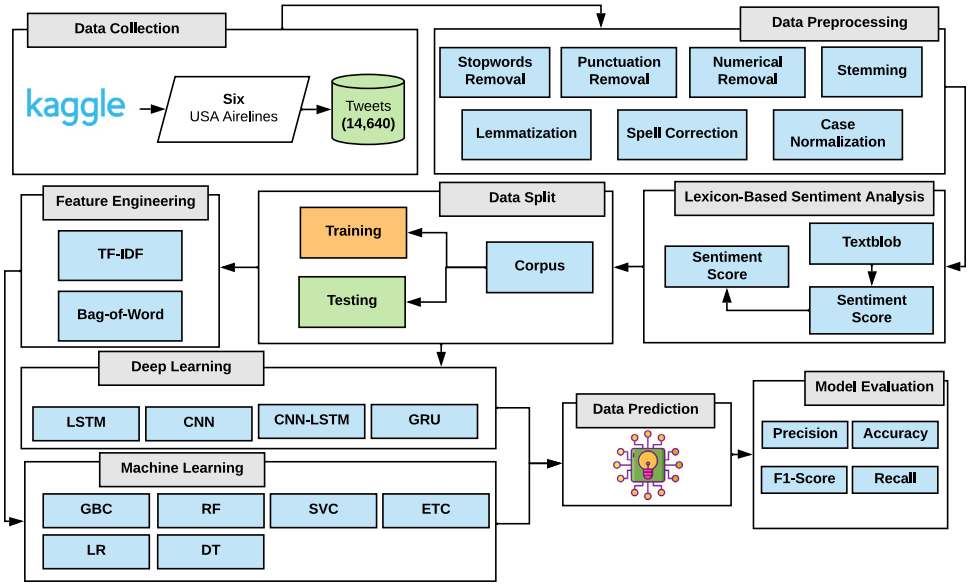


Fig. 1. Architecture of the proposed approach.

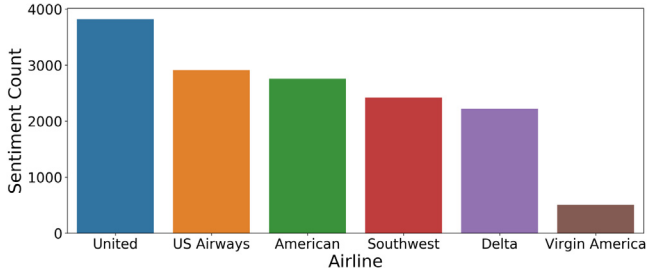


Fig. 2. Tweets count for each airline company.

Table 2
Information included in our dataset.

Attribute	Description
Airline sentiment confidence	Numeric value indicates the level of confidence for placing a tweet to one of the three classes.
Negative reason	This indicates why a given tweet is considered negative.
Negative reason confidence	Confidence level in identifying a negative tweet.
Airline	The name of the airline company.
Retweet count	Number of people who reposted (retweet) a tweet.
Text	This is the original posted tweet
Airline sentiment	Shows tweets labels which can be positive, negative, or neutral

the words to extract the root form of words. For this purpose, complete dictionaries are required to link the words to their lemma. After that, spell corrections are used to repair wrong words. Finally, the elimination of stop words is done as stop words do not have a critical significance for text processing. The sample text after carrying out each preprocessing step is shown in Table 3.

3.4. Use of TextBlob

This study investigated the use of the TextBlob to improve classification models' accuracy.

Table 3
Sample tweets after data preprocessing steps.

Pre-processing step	Example
Original tweet	@"VirginAmerica This is such a great deal! Already thinking about my 2nd trip to @Australia & I haven't even gone on my 1st trip yet! ;p"
Remove tags and username	"This is such a great deal! Already thinking about my 2nd trip to & I haven't even gone on my 1st trip yet! ;p"
Punctuation removal	"This is such a great deal Already thinking about my 2nd trip to I haven t even gone on my 1st trip yet p"
Numerical removal	"This is such a great deal Already thinking about my nd trip to I haven t even gone on my st trip yet p"
Length less than two words removal	"This is such great deal Already thinking about my nd trip to haven even gone on my st trip yet"
Case normalization	"this is such great deal already thinking about my nd trip to haven even gone on my st trip yet"
Stemming	"this is such great deal alreadi think about my nd trip to haven even go on my st trip yet"
Stop-words removal	Great deal nd trip go st trip

3.4.1. Why TextBlob is needed?

Predominantly existing studies use the 'US airline dataset' with the original annotation which is performed manually. Presumably, human experts have been regarded as the best annotators and several important facts have been ignored. Manual annotations are subjective where the text is analyzed by human experts to determine its sentiment. However, this subjectivity varies from one expert to another. Annotations are also affected by the mental state of the expert. Similarly, the element of bias and human error cannot be ignored fully. Existing studies focus on improving the performance of models by optimizing models and features engineering approaches and the characteristics of datasets are ignored or underexplored. Considering the above-mentioned issues, this study adopts the use of TextBlob.

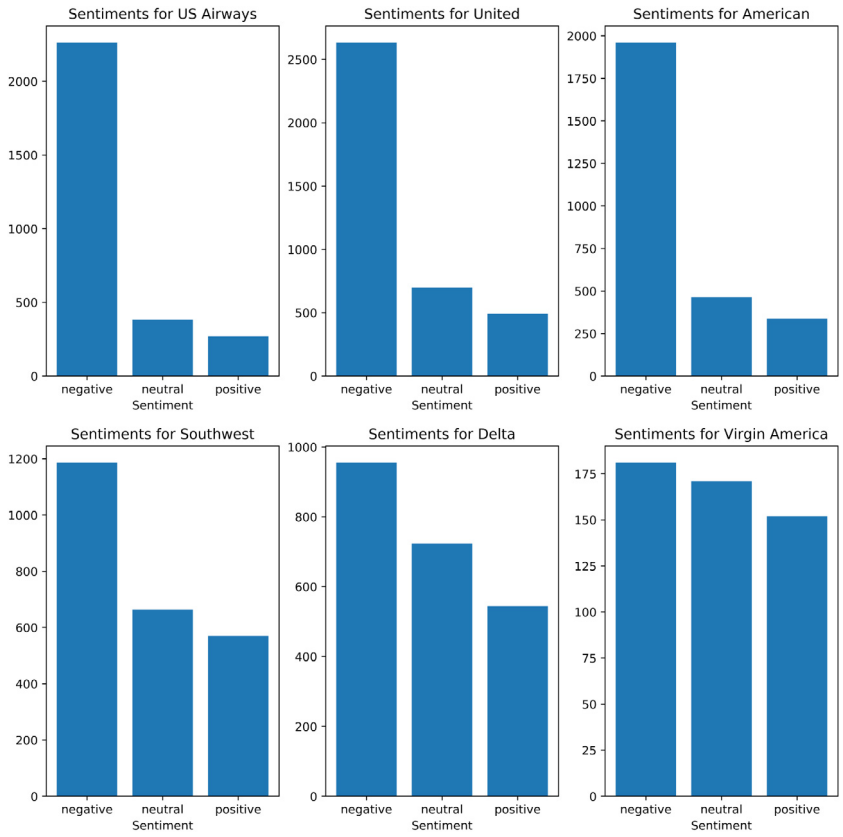


Fig. 3. Each sentiment count in the original dataset for each airline company.

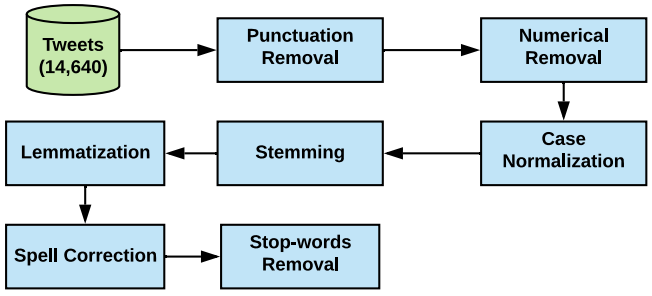


Fig. 4. Data preprocessing steps followed on tweets dataset.

Table 4
Sample tweets with contradictory labels.

Tweets	Label	
	Original	TextBlob
flying @virginamerica	Negative	Neutral
@virginamerica awaiting return phone call would prefer use online self service option	Negative	Neutral
@virginamerica random distribution elevate avatars bet kitty disproportionate share	Neutral	Negative

3.4.2. Preliminary analysis of data annotations

As a preliminary step, this study first analyzes the dataset labels manually to investigate if any contradictions exist between the given text and its label. To our surprise, many such examples are found, a few sample tweets with contradictory labels are provided in Table 4

This preliminary analysis is also important to determine the existence of contradictions. The majority of existing studies use TextBlob for sentiment analysis only, only a few have used it for data annotation like [44,45]. Although high accuracy is reported when using machine learning models on TextBlob annotated data, an in-depth analysis is missing. Largely, studies do not discuss why the performance of models is better with TextBlob. However, the results given in Table 4 indicate the reason that discrepancy between the text and its label often leads to poor performance of models with the original labels.

3.4.3. TextBlob

TextBlob is a popular sentiment analysis lexicon-based library model available in Python that provides simplified text processing [46]. The data scientists prefer using TextBlob since it can be a faster and more compact library. The simple API (Application Programming Interface) of TextBlob facilitates many of the common text processing and NLP tasks, such as language translation, POS (Parts of Speech) tagging, tokenization, phrase extraction, classification, sentiment analysis, and more [46,47]. TextBlob allows appropriate sentiment analysis of the text and enables the translation of tweets from one language to another. It has pre-trained machine learning models to perform sentiment analysis which is considered a complex machine learning problem [47]. The default sentiment analyzer of TextBlob performs valuable NLP tasks to determine the sentiments of a given text [47].

3.5. Feature extraction

Input cannot be directly fed to the machine learning models in text form. We have to convert input text into numerical representation before passing it to learning models. For this, different extraction techniques are available and this study uses two well-known feature extraction techniques including BoW and TF-IDF [48,49].

Table 5
Sample result data of the BoW technique on the preprocessed data.

big	delay	flight	passenger	problem
0	1	1	0	0
1	1	0	1	1

Table 6
Sample result of TF-IDF technique on the preprocessed data.

big	delay	flight	passenger	problem
0	0.579	0.815	0	0
0.534	0.379	0	0.534	0.534

3.5.1. Bag of words

The BoW is the simplest feature extraction method which provides an easy and flexible way to obtain document features. In this method, the histogram of the words is considered, i.e., every word counts as a feature. The frequency of a word is used as a function for the training set. To implement the BoW method, CountVectorizer from Scikit-learn is used. The CountVectorizer operates on the frequency of words, which implies that tokens are counted, and a limited matrix of tokens is generated [50]. Table 5 shows the BoW features for the preprocessed sample text, 'flight delay', and 'delay big problem passenger', respectively.

3.5.2. Term frequency-inverse document frequency

TF-IDF is another widely used feature extraction technique that converts the text data into numerical representation by assigning a weight to each word in the corpus [14]. TF-IDF computes the weight by multiplying the term frequency (TF) of a term and inverse document frequency for terms (IDF). TF can be computed as:

$$TF = tf_{t,d} \tag{1}$$

where t refers to a unique term and d is the document.

The IDF can be computed as:

$$IDF = log(\frac{D}{D_t}) \tag{2}$$

where D is the number of documents and D_t is the number of documents containing the term t.

$$tf - idf = tf_{t,d} \cdot log(\frac{D}{D_t}) \tag{3}$$

The calcualted TF-IDF for the preprocessed sample data is shown in Table Table 6.

3.6. Supervised machine learning models

This study deploys six machine learning algorithms to solve the classification problem. A brief description of each algorithm is provided here for completeness.

3.6.1. Decision tree

It is a tree-based model used for regression and classification problems. DT predicts the class by learning simple decision rules [51]. DT uses nodes and leaves by sorting them down from the root and adopting the representation of the sum of the product. To create the tree, DT utilizes the tree's Gini Index, or IG (Information Gain). These algorithms make the choice of the most suitable division in DT vital. The optimal split is selected by maximizing the data gain when training a DT. The following equation is used to determine IG

$$Entropy = -\sum_{i}^{N} Probability_i(class_1) + logProbability_i(class_1) \tag{4}$$

where N is the number of target classes.

DT finds the entropy for each class which is used to find the IG using the following

$$IG = \sum_{z} P(z) log P(z) \tag{5}$$

3.6.2. Random forest

RF combines a number of decision trees under majority voting criteria for the classification of data [52]. RF builds multiple decision trees to make a prediction and makes a prediction using the majority voting criteria. RF is an ensemble architecture and uses the bagging method to train each DT under its umbrella making it significant for any type of data. RF can be described mathematically as

$$rf = mode\{T_1, T_2, T_3, \ldots, T_n\} \tag{6}$$

or,

$$rf = mode\{\sum_{i=1}^{n} T_i\} \tag{7}$$

where $T_1, T_2, T_3, \ldots, T_n$ are the trees in RF and n is the number of trees.

3.6.3. Extra trees classifier

ETC is an ensemble model that combines multiple decision trees under majority voting criteria. It works similar to RF as it trains multiple weak learners to make predictions for a target class [53]. Prediction by each tree is considered as a vote for target classes. The target class with more votes is the final class. The difference between ETC and RF lies in the data selection for training where ETC uses a sample for each tree training while RF uses a different random sample for the training of each tree. This study used the ETC with 300 decision trees that make predictions under majority voting criteria [54].

3.6.4. Gradient boosting classifier

GBC is an ensemble tree-based classifier like RF and ETC, but it uses boosting methods to improve the accuracy. GBC reduces the error rate by using the learning rate concept, which makes it significant compared to other models. GBC follows an iterative method of adding the trees (weak learners). After each iteration, the value of the loss function must be reduced. GBC obtains the odds in the log of the objective value.

$$log(odd) = log(\frac{class1}{class2}) \tag{8}$$

where log(odds) is used for classification by converting it into probability using the following equation

$$p(class1) = \frac{e^{log(odds)}}{1 + e^{log(odds)}} \tag{9}$$

3.6.5. Logistic regression

It is a statistical model used for classification and can perform well when the feature set for the training is large. LR finds the interdependencies between dependent and independent entities [13]. It uses the sigmoid function for the separation/classification of data. Sigmoid function is a S-shaped function $\sigma : R \longrightarrow (0, 1)$ defined as

$$g(x) = \frac{L}{1 + e^{-m(z - z_o)}} \tag{10}$$

Table 7
Hyperparameters of machine learning used in our study.

Model	Hyperparameter	Description
RF	n_estimators = 300	The number of decision trees
	random_state = 5	The random state sample was taken these random decisions to be managed
	max_depth = 52	The maximum depth between each tree
	random_state = 52	The random state uptake and accumulation these random decisions to be managed
ETC	n_estimators = 300	The number of decision trees
	random_state = 5	The bootstrapping of the samples used when building trees
	max_depth = 52	The number of trees depth
GBC	max_depth = 300	Maximum depth of both the estimated regression estimation techniques
	learning_rate = 0.2	Learning rate by backpropagation decreases the contribution of every other tree
	n_estimators = 300	The number of steps to improve efficiency. Gradient boosting is reasonably resilient to over-fitting.
	random_state = 52	The random seed provided to each node estimator.
DT	max_depth = 300	The maximum tree depth.
LR	multi_class = 'multinomial'	Best to solve the multi-class classification problem.
	C = 3.0	Inverse of regularization strength
SVC	kernel = 'linear'	It maps the observations into some feature space.
	C = 2.0	The penalty parameter of the error term
	random_state = 52	The opposite of the power of regularization; it must have been a positive float

3.6.6. Support vector classifier

SVC is a classification model which can solve both linear and non-linear problems [55]. SVC generates multiple hyperplanes to classify the data with a great margin from data points but the one with the best margin will be selected for classification. This study used SVC with a linear kernel which can be more effective for text classification [13]. SVC prediction can be defined mathematically as

$$h(z_i) = sign(\sum_{j=1}^{s} u_j v_j K(z_j, z_i) + d) \quad (11)$$

$$K(v, v') = exp(\frac{\| \| v - v' \|\|^2}{2y^2}) \quad (12)$$

The selected models are optimized by tuning many hyper-parameters using the GridSearchCV method and a list of the hyperparameters used in this study is provided in Table 7.

4. Results

This section discusses the results and findings regarding the research questions.

4.1. Of the manual annotation and TextBlob annotation, which one is better and why?

Wrong annotations may be caused by subjective evaluations by different experts, influenced mental state, human error, and bias. Consequently, the performance of machine learning models may become poor if trained on incorrectly annotated data. The probability of incorrect annotations cannot be overlooked due to human error and TextBlob is used to analyze the difference in the ratio of positive, negative, and neutral sentiments. Table 8 shows the sample text from tweets along with the original sentiments from the dataset and the sentiment assigned by TextBlob after the preprocessing. It indicates that the assigned sentiments can be very different from the original sentiments.

Fig. 5 shows the sentiment ratio for the original annotations and TextBlob sentiments. It can be observed that the ratio before and after applying the TextBlob is significantly different. It shows that the original dataset has a higher number of negative sentiments followed by neutral and positive sentiments. However, the distribution of positive, negative, and neutral sentiments for TextBlob annotated text has a lower difference in the number of tweets. Specifically, the ratio of neutral and positive sentiments is almost similar. This substantial difference in the distribution of tweets is further investigated to analyze the impact on the performance of machine learning classifiers.

Table 8
Sample of tweets dataset.

Tweets	Original sentiment	Our annotation
"@VirginAmerica plus you've added commercials to the experience... tacky."	Positive	Neutral
"@VirginAmerica I didn't today... Must mean I need to take another trip!"	Neutral	Negative
Nice RT @VirginAmerica: Vibe with the moodlight from takeoff to touchdown	Neutral	Positive
@VirginAmerica you're the best!! Whenever I use any other airline I'm delayed and Late Flight :(	Negative	Positive

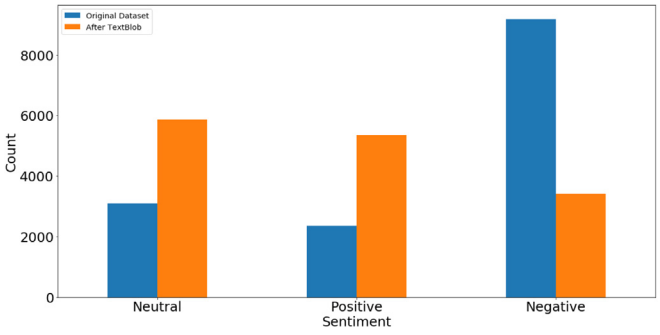


Fig. 5. Sentiment count in the original dataset and after applying TextBlob.

4.2. RQ2: To what extent can the TextBlob method improve the models' accuracy for sentiment analysis?

The focus of this research question is to examine the performance of the classifiers when performing sentiment analysis using TextBlob. To investigate this, we apply six supervised machine learning algorithms. We also want to compare the two selected feature extraction methods to observe which feature extraction method performs well compared with the previous work [13]. We apply the same data splitting approach as the study [13] to make an accurate and reliable comparison. By performing this comparison, we will be able to come up with a conclusion if the TextBlob method can improve the classifier's accuracy.

Table 9
Performance of machine learning with TF-IDF features.

Classifier	Accuracy	Class	Precision	Recall	F1 score
RF	0.87	0	0.81	0.98	0.89
		1	0.92	0.84	0.88
		2	0.91	0.71	0.79
		Macro Avg.	0.88	0.84	0.85
LR	0.90	0	0.86	0.98	0.91
		1	0.94	0.88	0.91
		2	0.91	0.78	0.84
		Macro Avg.	0.90	0.88	0.89
GBC	0.77	0	0.66	0.99	0.79
		1	0.93	0.70	0.80
		2	0.90	0.49	0.64
		Macro Avg.	0.83	0.73	0.74
SVC	0.92	0	0.89	0.98	0.93
		1	0.95	0.90	0.92
		2	0.91	0.84	0.87
		Macro Avg.	0.92	0.90	0.91
DT	0.90	0	0.90	0.98	0.94
		1	0.91	0.86	0.89
		2	0.86	0.81	0.83
		Macro Avg.	0.89	0.88	0.89
ETC	0.92	0	0.91	0.98	0.94
		1	0.94	0.90	0.92
		2	0.90	0.84	0.87
		Macro Avg.	0.91	0.90	0.91

Table 10
Performance of machine learning with BoW features.

Classifier	Accuracy	Class	Precision	Recall	F1 score
RF	0.90	0	0.85	0.99	0.92
		1	0.93	0.88	0.90
		2	0.92	0.76	0.84
		Macro Avg.	0.90	0.88	0.88
LR	0.91	0	0.88	0.98	0.93
		1	0.95	0.89	0.92
		2	0.91	0.81	0.86
		Macro Avg.	0.91	0.89	0.90
GBC	0.92	0	0.89	0.98	0.93
		1	0.94	0.90	0.92
		2	0.92	0.84	0.88
		Macro Avg.	0.92	0.90	0.91
SVC	0.92	0	0.89	0.98	0.94
		1	0.95	0.89	0.92
		2	0.91	0.83	0.87
		Macro Avg.	0.92	0.90	0.91
DT	0.91	0	0.92	0.98	0.95
		1	0.92	0.88	0.90
		2	0.86	0.81	0.83
		Macro Avg.	0.90	0.89	0.90
ETC	0.92	0	0.91	0.98	0.95
		1	0.94	0.91	0.92
		2	0.91	0.85	0.88
		Macro Avg.	0.92	0.91	0.92

The accuracy scores of the learning models show that the performance of machine learning models has been increased as compared to the previous study [13]. Table 9 presents the results of all models using TF-IDF technique. SVC and ETC both show an accuracy of 0.92. Results indicate that when using the TextBlob method with the TF-IDF technique, the DT classifier performs well and showed 0.22 better accuracy than the previous work [13], followed by the SVC, ETC with a 0.18 and 0.16 increase in accuracy, respectively. Then, RF and LR classifiers show a 0.12 better performance while the GBC shows marginal improvements. The improvements shown in the performance of classifiers are based on the number of correct and wrong predictions made by each classifier.

Table 10 shows the results of the BoW technique which shows that when using the TextBlob method with the BoW technique, the DT classifier performs significantly better with a 24% improvement compared to the previous study. The accuracy of the GBC classifier is increased from 0.74 to 0.92. The improvement indicates a change of 0.18 compared with not using the TextBlob method. SVC and ETC show a similar improvement with a 0.15 increase in accuracy scores. However, the accuracy of LR is increased from 0.78 to 0.91 indicating an improvement of 0.13 in the accuracy score.

Comparing both BoW and TF-IDF techniques, improvement in the classification accuracy is observed as compared to the previous study [13]. BoW technique presents better results with the TextBlob method compared to using the TextBlob with the TF-IDF technique. GBC shows a significant improvement when using the TextBlob method with the BoW technique where the accuracy is increased by 0.18. Overall, using the BoW technique with the TextBlob method can improve the models' accuracy with a minimum of 0.13 accuracy score.

The impact of implementing the TextBlob method on the classifiers' performance is analyzed using the number of correct and wrong predictions with the help of the confusion matrix, as given in Fig. 6. It indicates that SVC and ETC perform equally well. ETC gives 3330 correct predictions out of 3660 and gives 330 wrong predictions, while SVC gives 3353 correct predictions out of 3660 and gives 307 wrong predictions. This indicates that SVC is the best performer compared to the other models when using TF-IDF with the TextBlob method.

When the BoW technique is used with the TextBlob method, the accuracy of the three classifiers is found to be 0.92. For further analysis, the confusion matrix is analyzed, as shown in Fig. 7. It indicates that GBC, SVC, and ETC classifiers have an equal highest accuracy score. Therefore, we further investigate the prediction for each classifier by the number of correct and wrong predictions. GBC gives 3351 correct predictions out of 3660 and gives 309 wrong predictions, and SVC gives 3349 correct predictions out of 3660 and gives 311 wrong predictions while ETC gives 3357 correct predictions out of 3660 and has 303 wrong predictions. Thus, ETC is the best performer compared to the other models when using BoW with the TextBlob method.

Results indicate that classifiers perform better using the TextBlob than the previous study [13] which implements the models on the original labels. A significant improvement is observed in the performance of SVC, ETC, and LR classifiers. Similarly, linear models also perform better with TF-IDF techniques.

RQ:2 Discussion The machine learning model's performance depends on the features and if the feature will be more correlated to the target class then learning will be good and performance will be efficient. To make a good correlation between features and target class an accurate annotation is required which leads each sample to an accurate label. The used dataset was manually labeled and may the annotator do it based on context such as if sentence words are somehow positive but the context of the sentence is negative such as "do you this it's a very comfortable or good flight". This sentence contains positive words but the context is neutral. This contextual-based manual data annotation leads to complexity in the learning procedure of models. For example, Table 11 shows two samples with the same features but different labels. In one step model is learning that "comfortable", "good", "flight", and "very" features belong to the neutral class and in the next step is learning that the same "comfortable", "good", "flight", "very" features are belonging to the positive class. So this is a problem and it's solved by the Textblob because it didn't annotate the dataset on the base of context but based on polarity scores and according to polarity both sentences are positive. This correction in data annotation improved the performance of learning models.

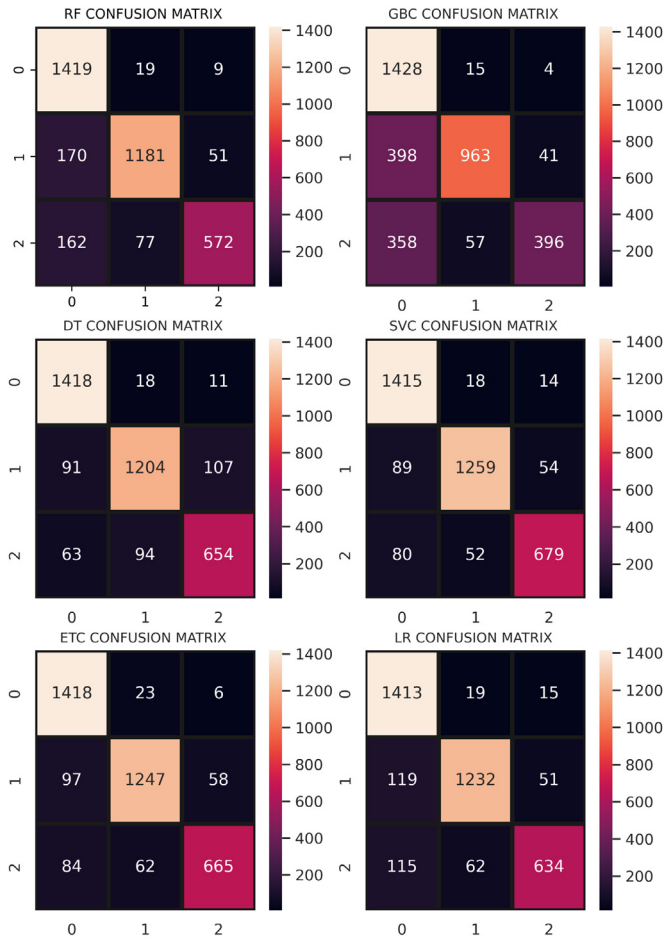


Fig. 6. Confusion matrices of classifiers using TF-IDF technique. (0 for neutral, 1 for positive, & 2 for negative).

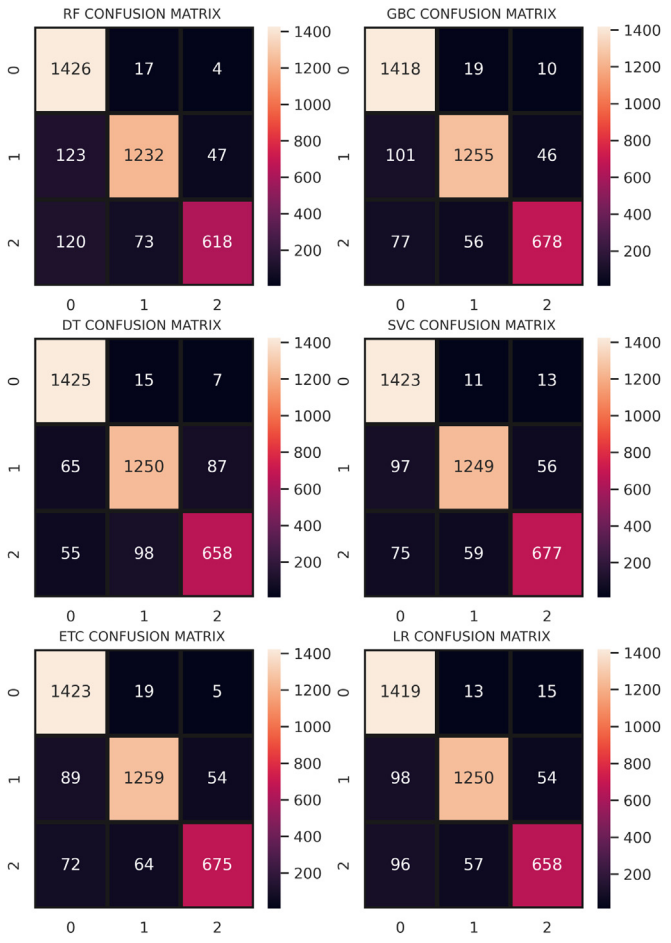


Fig. 7. Confusion matrices of classifiers using the BoW technique. (0 for neutral, 1 for positive, & 2 for negative).

Table 11
Data samples for discussion of RQ2.

No.	Tweet	Label
1	Do you this it's a very comfortable or good flight	Neutral
2	It's a very comfortable or good flight	Positive

4.3. RQ3: How efficient the models are when the sentiment analysis is performed using the TextBlob with respect to other lexicons?

To measure the efficiency of the lexicon-based technique TextBlob, this study also utilizes VADER (Valence Aware Dictionary for sEntiment Reasoning) and Afinn on the same dataset. The results comparison between TextBlob, VADER, and Afinn is given in Table 12. Depending on the approach followed to assign a sentiment to text, the assigned labels are different for each approach and so does the performance of machine learning classifiers. VADER depends on mapping the lexicon features into sentiment scores or sentiment intensities using a dictionary [56]. AFINN is the most straightforward lexicon technique with a dictionary of 3300+ words along with the polarity score. AFINN maps the polarity score to each word in the text and sums up the score of each word. While TextBlob assigns the polarity score to each word between −1 and 1 based on the polarity and subjectivity. According to the results, TextBlob outperforms VADER and Afinn with both BoW and TF-IDF features, followed by VADER which obtains better results than Afinn with BoW and TF-IDF.

Regarding the RQ3, the efficiency indicates the classification accuracy of models. The performance of the machine learning models is improved with the new annotation approach as compared to previous studies. The high correlation between the text and the assigned label leads to increased performance for models. Previous studies utilize manually labeled datasets that have problems, as discussed previously. Data annotation using lexicon approaches such as VADER and AFINN also show that the TextBlob is better and shows significantly better results for sentiment analysis as it is a more accurate dictionary as compared to others [57,58].

Table 12
Models' performance comparison using Afinn, VADER, and Textblob.

Model	Afinn		VADER		Textblob	
	BoW	TD-IDF	BoW	TF-IDF	BoW	TF-IDF
RF	0.81	0.81	0.82	0.81	**0.90**	**0.87**
LR	0.84	0.83	0.88	0.88	**0.91**	**0.90**
GBC	0.84	0.71	0.85	0.74	**0.92**	**0.77**
SVC	0.85	0.84	0.88	0.87	**0.92**	**0.92**
DT	0.81	0.74	0.84	0.75	**0.91**	**0.90**
ETC	0.84	0.84	0.85	0.84	**0.92**	**0.92**

Fig. 8 shows the feature space using original annotation and TextBlob annotation where legends 0, 1, and 2 indicate 'negative', 'neutral', and 'positive' classes, respectively. It can be observed in Fig. 8(a) that the negative class dominates other target classes as negative class features are overlapping the positive and neutral class features. This overlapping of features creates complexity for learning models. On the other hand, Fig. 8(b) shows feature space using the TextBlob annotation and the overlapping effects are

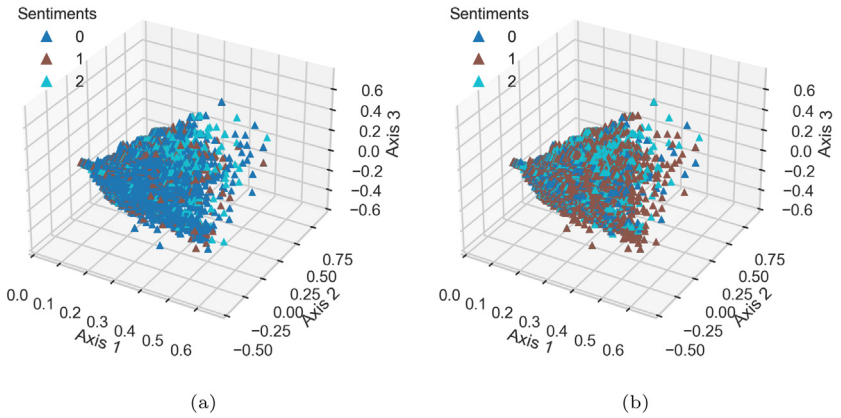


Fig. 8. Feature space, **(a)** Feature space using original annotation, and **(b)** Feature space using TextBlob annotation.

very low as each target has its separate feature space which helps the learning models to obtain a good fit with TextBlob features.

4.4. Results of deep learning model

Deep learning models are also implemented in comparison with machine learning models. Through the machine learning approach, we investigate the lexicon approaches and find Textblob as the best performer. So for deep learning experiments, we used only Textblob for dataset annotation and we do not further investigate VADER and AFINN. We deploy state-of-the-art deep learning models including LSTM, CNN, GRU, and a combination of CNN and LSTM as CNN-LSTM. The architecture of all deep learning models is shown in Fig. 9 with a detailed description of layers and parameters. We compile all models using categorical cross-entropy loss function because of multi-class data and use the 'Adam' optimizer. Models are fitted with a batch size of 32 and 100 epochs.

In addition to CNN, LSTM, GRU, and CNN-LSTM, this study proposes an ensemble deep learning model where LSTM and GRU are combined to obtain better performance. LSTM and GRU are ensembled regarding their superior performance for the task at hand. Both models are stacked and the architecture of the ensemble model is given in Fig. 10. LSTM-GRU consists of eight layers, one embedding layer, three dropout layers, one LSTM layer, one GRU layer, and one dense layer. The embedding layer takes text input features with a vocabulary size of 5000 while its output dimension is 200. The output of the embedding layer is fed to the dropout layer to reduce the complexity in input while the LSTM layer with 200 units is trained on these features. The output from the LSTM layer is then used in the GRU layer which has 100 units. Each layer in LSTM-GRU is followed by a dropout layer with a 0.5 dropout rate. In the end, we use a dense layer with 3 neurons because of the prediction for three classes, and the Softmax activation function is used. We compile the model with categorical_crossentropy loss function and Adam optimizer while 100 epochs are used for training with a batch size of 128.

The performance of deep learning models is shown in Table 13 which suggests that LSTM and LSTM-GRU outperform all deep learning and machine learning models regarding accuracy. LSTM and LSTM-GRU achieve the highest 0.97 accuracy score, followed by the GRU with a 0.96 accuracy score. These results show that the deep learning approach is more suitable compared to machine learning. Deep learning models perform better on large datasets. Secondly, the automatic feature extraction by deep learning models and finding the complex relationship among the features are superior to machine learning models which leads to superior performance by deep learning models provided the dataset size is large enough for training. LSTM has a recurrent structure with memory which can learn better as compared to traditional machine learning models.

Table 13
Performance of deep learning models.

Classifier	Accuracy	Class	Precision	Recall	F1 score
LSTM	0.97	0	0.92	0.93	0.93
		1	0.99	0.98	0.98
		2	0.94	0.96	0.95
		Macro Avg.	0.95	0.96	0.95
CNN	0.95	0	0.90	0.95	0.92
		1	0.99	0.98	0.98
		2	0.97	0.94	0.95
		Macro Avg.	0.95	0.96	0.95
CNN-LSTM	0.88	0	0.80	0.82	0.81
		1	0.93	0.92	0.93
		2	0.83	0.82	0.83
		Macro Avg.	0.85	0.86	0.85
GRU	0.96	0	0.93	0.94	0.93
		1	0.99	0.97	0.98
		2	0.92	0.95	0.93
		Macro Avg.	0.94	0.95	0.95
LSTM-GRU	0.97	0	0.94	0.94	0.94
		1	0.99	0.98	0.99
		2	0.94	0.97	0.96
		Macro Avg.	0.96	0.96	0.96

Both LSTM and its ensemble with GRU obtain the same accuracy score of 0.97, however, the performance of LSTM-GRU is superior with respect to the macro average for precision and F1 score as it achieves a 0.96 score for both precision and F1 score.

Table 14 shows the results of machine learning and deep learning models for sentiment classification. The proposed model LSTM-GRU shows the highest performance using the hybrid annotation approach with a 0.92 accuracy score. The performance of models with both BoW and TF-IDF is significant as SVC achieves a 0.91 accuracy with both features. However, the performance of the models is not as good as it is using the labels from TextBlob alone. It is so because using AFINN and VADER annotations, models show poor performance indicating that both provide inappropriate labels. Using their output in the hybrid model also influences the annotations from the hybrid annotation approach.

4.5. Experiments with additional dataset

For further verification of the ambiguity found in the manually labeled dataset, additional experiments are conducted. A new dataset is collected from Twitter which contains 1000 tweets. The tweets are labeled manually for experiments. The labels are assigned using three human experts with the following criteria.

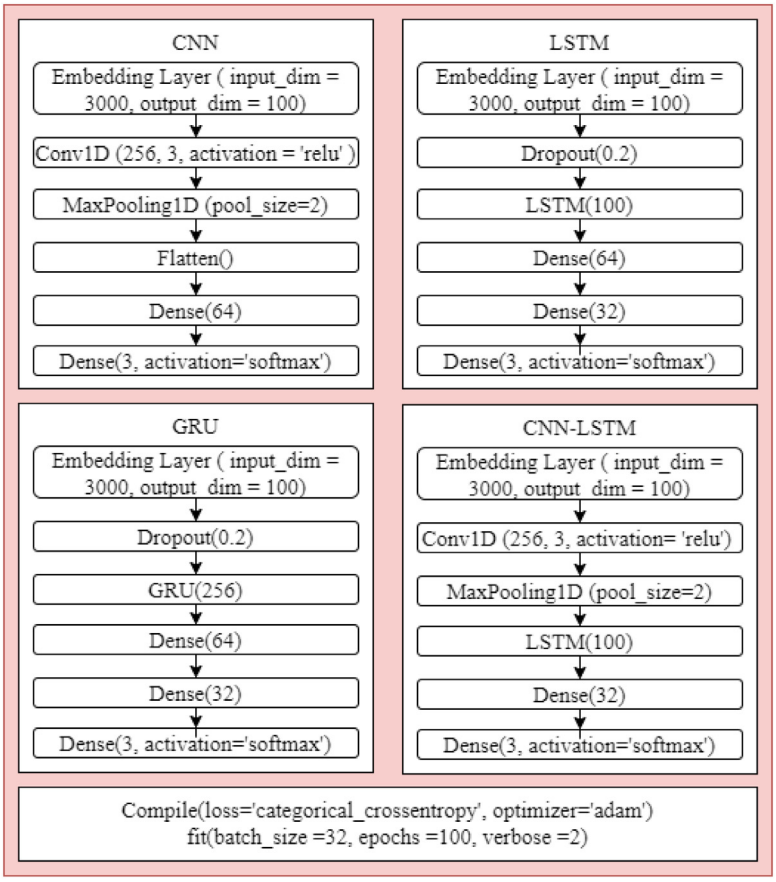


Fig. 9. Deep learning models architectures.

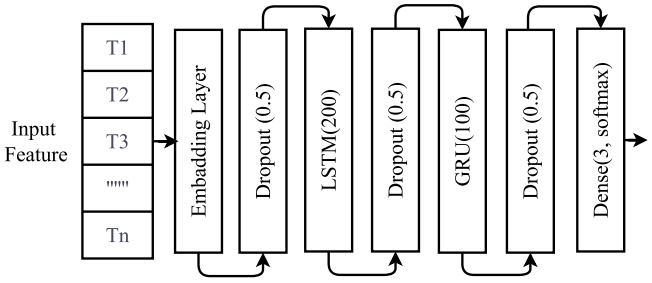


Fig. 10. Architecture of proposed LSTM-GRU model.

- Labels are assigned by individual human experts.
- The final label is assigned if at least two of the experts agree on the same label.
- Tweet, where three experts have different labels is discarded.

The distribution of the positive, negative, and neutral samples using the human experts is shown in Fig. 11(a). For experiments, the same dataset is annotated using the TextBlob as well to analyze the difference in the assigned labels from the human experts and TextBlob. The distribution of tweets using TextBlob annotation is given in Fig. 11(b). It can be observed that there is a significant difference, especially in the number of positive and negative tweets when human experts and TextBlob are used for annotation. The reason for this difference is the contextual label from a human expert. Table 15 shows two samples from the collected data and their associated labels from human experts and TextBlob. It can be observed that labels are different.

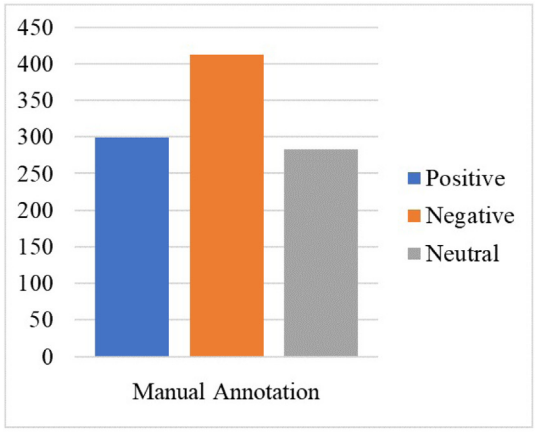
Table 14
Results of machine learning and deep learning models using hybrid data annotation.

Model	Feature	Accuracy	Precision	Recall	F1 Score
RF	BoW	0.89	0.89	0.89	0.89
LR		0.90	0.90	0.89	0.89
GBC		0.91	0.91	0.91	0.91
SVC		0.91	0.91	0.91	0.91
DT		0.60	0.73	0.57	0.57
ETC		0.90	0.90	0.89	0.89
RF	TF-IDF	0.88	0.89	0.88	0.88
LR		0.88	0.88	0.87	0.87
GBC		0.90	0.90	0.89	0.89
SVC		0.91	0.91	0.91	0.91
DT		0.60	0.73	0.57	0.57
ETC		0.88	0.88	0.88	0.88
CNN	–	0.89	0.89	0.89	0.89
LSTM		0.91	0.91	0.91	0.91
CNN-LSTM		0.76	0.76	0.76	0.76
GRU		0.91	0.91	0.91	0.91
LSTM-GRU		0.92	0.92	0.92	0.92

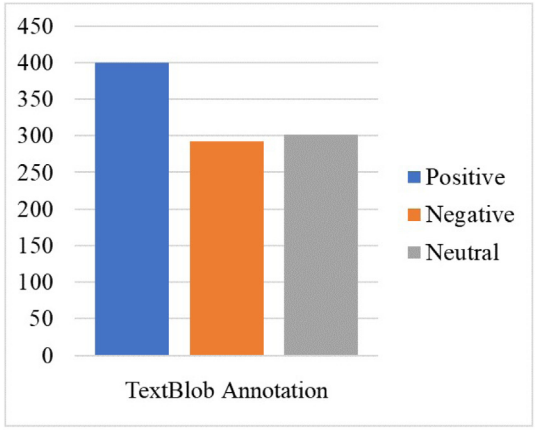
Table 15
Samples from the collected dataset.

Text	Contextual/Manual	TextBlob
Its not good and not bad	Neutral	Negative
Its a good thing?	Neutral	Positive

Another problem is the subjectivity of the human expert which may change from one expert to another and even for one expert if he is to label the same text over different times. TextBlob, on the other hand, assigns the label based on polarity



(a)



(b)

Fig. 11. Distribution of labels, (a) Manual annotation, and (b) TextBlob annotation.

Table 16
Models results using manually labeled dataset.

Model	Accuracy	Precision	Recall	F1 Score
RF	0.44	0.45	0.42	0.42
LR	0.43	0.43	0.43	0.43
GBC	0.38	0.38	0.38	0.38
SVC	0.43	0.42	0.43	0.42
DT	0.41	0.14	0.33	0.19
ETC	0.45	0.46	0.44	0.44

and has the same label even when the same text is tried at different times.

Experiments are performed using a human expert-labeled dataset and TextBlob-labeled dataset separately. Table 16 shows the results when a manually labeled dataset is used for experiments. Results indicate that a maximum of 0.45 accuracy is obtainable with the ETC model. RF shows a marginally reduced accuracy.

Similar to a manually labeled dataset, the TextBlob-labeled dataset is used with the same models to analyze their performance. Experimental results are shown in Table 17. It can be observed that the performance of the models is elevated when trained and tested using the TextBlob-labeled dataset. The best performance is obtained using the GBC model which obtains a 0.70 accuracy score and similar scores for precision, recall, and F1. The performance of LR and ETC is marginally lower with a 0.69 score each. Results show better performance of machine learning models using TextBlob-labeled dataset.

Table 17
Models results using TextBlob-labeled dataset.

Model	Accuracy	Precision	Recall	F1 Score
RF	0.65	0.69	0.68	0.65
LR	0.69	0.70	0.70	0.69
GBC	0.70	0.71	0.70	0.70
SVC	0.65	0.66	0.67	0.66
DT	0.56	0.68	0.61	0.53
ETC	0.69	0.71	0.70	0.69

Table 18
Comparison with previous studies. Results for [13,16,17] are obtained using the human-annotated dataset.

Ref.	Year	Approach	Accuracy
[13]	2019	ML	79%
[16]	2021	ML, DL, DNN + CNN	87%
[17]	2021	ML, DL	89%
This work	**2022**	**ML (SVC, ETC)**	**92%**
		DL (LSTM, LSTM-GRU)	**97%**

In addition, to analyze the difference between manual annotation and TextBlob annotation, the number of records where the label is different are counted for the newly collected dataset. Of the 1000 tweets, 491 records have the same labels from manual annotation and TextBlob annotation while 509 labels are different showing a difference of approximately 51% (50.9% to be precise) in the annotation. Different labels are found for 38 neutral tweets, 304 negative tweets, and 149 positive tweets from the manually labeled dataset.

4.6. Comparison with previous studies

We also compare the results with several previous studies that use the same dataset for experiments. Table 18 shows the performance comparison indicating that results with the current approach is far better than previous state-of-the-art studies using both machine learning and deep learning models. The study [16] used a deep learning approach with the embedding schema, which requires high computational cost and was still not able to provide very good results. While the study [17] uses machine learning models with different features and achieves the highest F1 score of 0.89. Existing studies use a manually annotated dataset while the current study makes use of a TextBlob-based annotated dataset. An increase in the performance of machine learning models is observed when used with TextBlob labels. This study, on the other hand, achieves the highest F1 score of 0.96 using an ensemble of LSTM and GRU while the highest accuracy is 0.97 by LSTM and LSTM-GRU models.

4.7. Statistical T-test

To show the statistical significance of the proposed approach, the statistical T-test is performed. The statistical T-test takes two hypotheses for the output as follows

- Null hypothesis (H_0): The proposed approach is statistically significant compared to previous approaches.
- Alternative Hypothesis (H_a): The proposed approach is not statistically significant compared to previous approaches.

The T-test results accept the H_0 when TextBlob is used compared to the original annotated data indicating that the use of TextBlob has statistical significance for both cases of BoW and TF-IDF. When we perform the T-Test for deep learning and machine learning models, it accepts the H_0 in favor of deep learning models which shows that deep learning is statistically significant than machine learning.

5. Conclusions and future work

The large use of social media platforms to share views and opinions regarding products, services, events, and personalities opened new possibilities to use for improving the quality of products and services. For this purpose, predominantly Twitter data is used and several annotated datasets are already available. However, the probability of wrong labels cannot be overlooked where the subjectivity of human experts, the influence of the mental state, human error, and bias can lead to wrong labels. This study analyzes the impact of TextBlob on the performance of machine learning and deep learning models in comparison to other studies that utilize the original labels. Extensive experiments are performed on the US airline tweets dataset using several machine learning models with BoW and TF-IDF and deep learning models. Linear models perform better using TF-IDF weighted features. Results indicate that machine learning models ETC and SVC outperform other machine learning models with both BoW and TF-IDF by achieving a 0.92 accuracy score. The deep learning model LSTM and ensemble model LSTM-GRU achieve the highest 0.97 accuracy score when used with TextBlob with LSTM-GRU having the highest precision and F1 scores of 0.96. The performance of both machine learning and deep learning models is significantly improved when trained using TextBlob sentiments. Furthermore, in comparison with other lexicon methods VADER and Afinn, TextBlob shows far better results.

We discuss the problem with manual annotation and the significance of TextBlob annotation. Despite the good performance of the models using the TextBlob labels, it cannot replace humans as the most precise annotation is regarded as the one which is done by human annotators. Our stance is that with humans, bias, error-proneness, and subjectivity cannot be ignored. So we propose that the TextBlob-annotated labels can be used as assistance for human annotators. Instead of annotating the dataset from scratch, human annotators can wet the TextBlob-annotated dataset and modify it as they deem fit. Expanding this study to include more datasets on several airline companies and different services can be done as future work.

CRediT authorship contribution statement

Wajdi Aljedaani: Conceptualization, Formal analysis, Writing – original draft. **Furqan Rustam:** Conceptualization, Methodology, Data curation. **Mohamed Wiem Mkaouer:** Data curation, Software, Visualization. **Abdullatif Ghallab:** Formal analysis, Methodology, Validation. **Vaibhav Rupapara:** Software, Resources, Validation. **Patrick Bernard Washington:** Data curation, Resources, Validation. **Ernesto Lee:** Formal analysis, Software, Visualization. **Imran Ashraf:** Methodology, Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. Devakunchari, Analysis on big data over the years, Int. J. Sci. Res. Publ. 4 (1) (2014) 1.

[2] R. Jacobson, 2.5 Quintillion Bytes of Data Created Every Day. how Does Cpg & Retail Manage It, IBM.

[3] Q. Wang, A. Kealy, S. Zhai, Introduction for the special issue on beyond the hypes of geospatial big data: Theories, methods, analytics, and applications.

[4] R.K. Bakshi, N. Kaur, R. Kaur, G. Kaur, Opinion mining and sentiment analysis, in: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, 2016, pp. 452–455.

[5] L. Rainie, J. Horrigan, Election 2006 online. pew internet & american life project, 2007, p. 2010, Retrieved on May 20.

[6] L.M. Qaisi, I. Aljarah, A twitter sentiment analysis for cloud providers: a case study of azure vs. aws, in: 2016 7th International Conference on Computer Science and Information Technology, CSIT, IEEE, 2016, pp. 1–6.

[7] K.L. Xie, C. Chen, S. Wu, Online consumer review factors affecting offline hotel popularity: evidence from tripadvisor, J. Travel Tour. Mark. 33 (2) (2016) 211–223.

[8] J. Horrigan, Online shopping: Internet users like the convenience but worry about the security of their financial information, 2008, retrieved april 7, 2008.

[9] H.-J. Kwon, H.-J. Ban, J.-K. Jun, H.-S. Kim, Topic modeling and sentiment analysis of online review for airlines, Information 12 (2) (2021) 78.

[10] S. Banerjee, L. Chai, Effect of individualism on online user ratings: Theory and evidence, J. Global Mark. 32 (5) (2019) 377–398.

[11] Q. Ye, R. Law, B. Gu, The impact of online user reviews on hotel room sales, Int. J. Hosp. Manag. 28 (1) (2009) 180–182.

[12] M. Umer, I. Ashraf, A. Mehmood, S. Kumari, S. Ullah, G. Sang Choi, Sentiment analysis of tweets using a unified convolutional neural network-long short-term memory network model, Comput. Intell. 37 (1) (2021) 409–434.

[13] F. Rustam, I. Ashraf, A. Mehmood, S. Ullah, G.S. Choi, Tweets classification on the base of sentiments for us airline companies, Entropy 21 (11) (2019) 1078.

[14] M. Mujahid, E. Lee, F. Rustam, P.B. Washington, S. Ullah, A.A. Reshi, I. Ashraf, Sentiment analysis and topic modeling on tweets about online education during covid-19, Appl. Sci. 11 (18) (2021) 8438.

[15] A. Rane, A. Kumar, Sentiment classification system of twitter data for us airline service analysis, in: 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Vol. 1, IEEE, 2018, pp. 769–773.

[16] K.M. Hasib, M.A. Habib, N.A. Towhid, M.I.H. Showrov, A novel deep learning based sentiment analysis of twitter data for us airline service, in: 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), IEEE, 2021, pp. 450–455.

[17] R. Al-Qahtani, P.N. bint Abdulrahman, Predict Sentiment of Airline Tweets using Ml Models, Tech. rep., EasyChair, 2021.

[18] H.T. Vo, H.C. Lam, D.D. Nguyen, N.H. Tuong, Topic classification and sentiment analysis for vietnamese education survey system, Asian J. Comput. Sci. Inf. Technol. 6 (3) (2016) 27–34.

[19] S. Sarkar, T. Seal, Sentiment analysis-an objective view, J. Res. 2 (02).

[20] A. Devitt, K. Ahmad, Sentiment analysis and the use of extrinsic datasets in evaluation, in: LREC, 2008.

[21] J. Khairnar, M. Kinikar, Machine learning algorithms for opinion mining and sentiment classification, Int. J. Sci. Res. Publ. 3 (6) (2013) 1–6.

[22] B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, arXiv preprint cs/0205070.

[23] H. Hakh, I. Aljarah, B. Al-Shboul, Online social media-based sentiment analysis for us airline companies, New Trends Inf. Technol. (2017) 176.

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

[25] B.A. David Mc, Service quality and customer satisfaction in the airline industry: A comparison between legacy airlines and low-cost airlines, Am. J. Tour. Res. 2 (1) (2013) 67–77.

[26] A. Kumar, T.M. Sebastian, Sentiment analysis on twitter, Int. J. Comput. Sci. Issues (IJCSI) 9 (4) (2012) 372.

[27] A. Hasan, S. Moin, A. Karim, S. Shamshirband, Machine learning-based sentiment analysis for twitter accounts, Math. Comput. Appl. 23 (1) (2018) 11.

[28] A.C. Pandey, D.S. Rajpoot, M. Saraswat, Inf. Process. Manage. 53 (4) (2017) 764–779.

[29] G. Tu, J. Wen, C. Liu, D. Jiang, E. Cambria, Context-and sentiment-aware networks for emotion recognition in conversation, IEEE Trans. Artif. Intell..

[30] Z. Lian, B. Liu, J. Tao, Smin: Semi-supervised multi-modal interaction network for conversational emotion recognition, IEEE Trans. Affect. Comput..

[31] A. Dai, X. Hu, J. Nie, J. Chen, Learning from word semantics to sentence syntax by graph convolutional networks for aspect-based sentiment analysis, Int. J. Data Sci. Anal. (2022) 1–10.

[32] A. Keramatfar, H. Amirkhani, A.J. Bidgoly, Modeling tweet dependencies with graph convolutional networks for sentiment analysis, Cognit. Comput. (2022) 1–12.

[33] B. Liang, H. Su, L. Gui, E. Cambria, R. Xu, Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks, Knowl.-Based Syst. 235 (2022) 107643.

[34] A. Zhao, Y. Yu, Knowledge-enabled bert for aspect-based sentiment analysis, Knowl.-Based Syst. 227 (2021) 107220.

[35] R. Chiong, G.S. Budhi, S. Dhakal, Combining sentiment lexicons and content-based features for depression detection, IEEE Intell. Syst. 36 (6) (2021) 99–105.

[36] D.M. Eler, D. Grosa, I. Pola, R. Garcia, R. Correia, J. Teixeira, Analysis of document pre-processing effects in text and opinion mining, Information 9 (4) (2018) 100.
[37] J. Li, G. Huang, C. Fan, Z. Sun, H. Zhu, Key word extraction for short text via word2vec, doc2vec, and textrank, Turk. J. Electr. Eng. Comput. Sci. 27 (3) (2019) 1794–1805.
[38] B.G. Gebre, M. Zampieri, P. Wittenburg, T. Heskes, Improving native language identification with tf-idf weighting, in: The 8th NAACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA8), 2013, pp. 216–223.
[39] R. Dzisevič, D. Šešok, Text classification using different feature extraction approaches, in: 2019 Open Conference of Electrical, Electronic and Information Sciences (EStream), IEEE, 2019, pp. 1–4.
[40] S. Vijayarani, R. Janani, et al., Text mining: open source tokenization tools-an analysis, Adv. Comput. Intell.: Int. J. (ACII) 3 (1) (2016) 37–47.
[41] S. Yang, H. Zhang, Text mining of twitter data using a latent dirichlet allocation topic model and sentiment analysis, Int. J. Comput. Inf. Eng 12 (2018) 525–529.
[42] M. Anandarajan, C. Hill, T. Nolan, Practical Text Analytics, Maximizing the Value of Text Data, in: Advances in Analytics and Data Science, vol. 2, Springer.
[43] N. Safdari, H. Alrubaye, W. Aljedaani, B.B. Baez, A. DiStasi, M.W. Mkaouer, Learning to rank faulty source files for dependent bug reports, in: Big Data: Learning, Analytics, and Applications, Vol. 10989, International Society for Optics and Photonics, 2019, p. 109890B.
[44] P. Gupta, S. Kumar, R. Suman, V. Kumar, Sentiment analysis of lockdown in india during covid-19: A case study on twitter, IEEE Trans. Comput. Soc. Syst. 8 (4) (2020) 992–1002.
[45] S. Sazzed, S. Jayarathna, Ssentia: a self-supervised sentiment analyzer for classification from unlabeled data, Mach. Learn. Appl. 4 (2021) 100026.
[46] D. Sarkar, Text Analytics with Python: A Practitioner's Guide To Natural Language Processing, Springer, 2019.
[47] P.J. Deitel, H. Dietal, Intro To Python for Computer Science and Data Science: Learning To Program with AI, Big Data and the Cloud, Pearson Education, Incorporated, 2020.
[48] R. Jamil, I. Ashraf, F. Rustam, E. Saad, A. Mehmood, G.S. Choi, Detecting sarcasm in multi-domain datasets using convolutional neural networks and long short term memory network model, PeerJ Comput. Sci. 7 (2021) e645.
[49] F.F. Bocca, L.H.A. Rodrigues, The effect of tuning, engineering, feature and feature selection in data mining applied to rainfed sugarcane yield modelling, Comput. Electron. Agric. 128 (2016) 67–76.
[50] S.C. Eshan, M.S. Hasan, An application of machine learning to detect abusive bengali text, in: 2017 20th International Conference of Computer and Information Technology, ICCIT, IEEE, 2017, pp. 1–6.
[51] S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE Trans. Syst. Man Cybern. 21 (3) (1991) 660–674.
[52] V. Svetnik, A. Liaw, C. Tong, J.C. Culberson, R.P. Sheridan, B.P. Feuston, Random forest: a classification and regression tool for compound classification and qsar modeling, J. Chem. Inf. Comput. Sci. 43 (6) (2003) 1947–1958.
[53] A.L. Samuel, Some studies in machine learning using the game of checkers, IBM J. Res. Dev. 3 (3) (1959) 210–229.
[54] C. Hayashi, What is data science? fundamental concepts and a heuristic example, in: Data Science, Classification, and Related Methods, Springer, 1998, pp. 40–51.
[55] B. Alkhazi, A. DiStasi, W. Aljedaani, H. Alrubaye, X. Ye, M.W. Mkaouer, Learning to rank developers for bug report assignment, Appl. Soft Comput. (2020) 106667.
[56] C. Gilbert, E. Hutto, Vader: A parsimonious rule-based model for sentiment analysis of social media text, in: Eighth International Conference on Weblogs and Social Media (ICWSM-14), Vol. 81, 2014, p. 82, Available at (20/04/16) http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf.
[57] S. Kumari, Z.A. Memon, Extracting feature requests from online reviews of travel industry, Acta Sci. Technol. 44 (2022) e58658.
[58] A.A. Reshi, F. Rustam, W. Aljedaani, S. Shafi, A. Alhossan, Z. Alrabiah, A. Ahmad, H. Alsuwailem, T.A. Almangour, M.A. Alshammari, et al., Covid-19 vaccination-related sentiments analysis: A case study using worldwide twitter dataset, in: Healthcare, Vol. 10, MDPI, 2022, p. 411.