scientific reports

OPEN

Sentiment analysis of emoji fused reviews using machine learning and Bert

Amit Khan[1]✉, Dipankar Majumdar[2] & Bikromadittya Mondal[3]

The usage of Natural Language Processing (NLP) technology powered by Artificial Intelligence in processing of customer feedback has helped in making critical decisions for business growth in the aviation sector. It is observed that in many of the cases, emojis and emoticons are found to convey a lot of significant information about the user's opinion or experience regarding a certain product, a service or an event. Consequently, it is very much essential that these emojis/emoticons are considered for processing because they are found to play a vital role in sentiment expression, often conveying more explicit information than the text alone. Their inclusion helps in capturing nuanced sentiments, improving the overall accuracy of sentiment classification. In Spite of the fact that these elements are a significant part of the review comment provided by the customer, it is a common practice among the contemporary researchers to eliminate them right at the data-cleaning or the preprocessing stage. With an objective to provide a solution to the above drawback, we present a novel approach that performs sentiment analysis, with effective utilization of emojis and emoticons, upon the US Airline tweet dataset using various Machine Learning classifiers and the BERT model. Finally, the proposed model was evaluated using various performance metrics and achieved 92% accuracy, outperforming contemporary state-of-the-art frameworks by 9%.

Keywords Sentiment analysis, Emojis, Machine learning, TF-IDF, BERT

Taking to inherent significant advantages from the perspectives of both manufacturers as well as end users, entrepreneurs have massively shifted their business paradigm[1,2] from offline (physical) mode to online mode in the last decade. Through its online operations, organizations are able to reach out to a huge number of customers[3] worldwide. At the same time, customers and end users get the chance to explore an extensive variety of items available across the globe[1–3].

Additionally, the customers also have the option to express their emotions about products[3] and items that they have used or come across on the social media platform. The posted reviews help the organization craft and implement changes or enhancements[2,3] for future and upcoming versions. This works both for the existing products and newly designed products[4,5] as well. In recent times, the writing pattern of social media users has evolved considerably, incorporating more pictographs known as emojis alongside text to enhance description, interactivity, and liveliness. Due to distance communication through social media like Facebook, Twitter, Instagram, and so on, people are unable to show their facial gestures and voice modulation[6,7] to emojis provides an excellent solution. Consequently, any social media comments or conversations regarding a certain product or service of recent times generally contain lots of emojis[8] and emoticons communicating useful information from the end user's perspective and experience.

Realizing the above-mentioned potential of the textual social media comments inclusive of emojis and emoticons from a business perspective, the current work aims to develop a framework for analyzing their cumulative effect during sentiment analysis (SA) from an airline review dataset as a case study. This work deployed both Machine Learning (ML) and Bidirectional Encoder Representations from Transformers (BERT) model to analyze the performance. Traditional ML models provide interpretability and require less computational resources, making them suitable for scenarios with limited data, while BERT leverages deep contextual understanding to enhance sentiment classification and its popularity and effectiveness in different Natural language Processing tasks. Our goal is to analyze their performance differences and determine the most

[1]Department of Information Technology, RCC Institute of Information Technology, Kolkata 700015, India. [2]Department of Computer Science & Engineering, RCC Institute of Information Technology, Kolkata 700015, India. [3]Department of Computer Science & Engineering, B. P. Poddar Institute of Management and Technology, Kolkata 700052, India. ✉email: amit.khan@rcciit.org.in

effective approach. We have also implemented data augmentation in order to balance our dataset to get rid of overfitting. The main contributions of this work are listed as follows.

(1) Implementation of Data Augmentation Techniques to address the data imbalance problem and overfitting problems.
(2) Comparison of SA models based on ML techniques and the BERT-based Transformer model using the Airline Tweets Dataset is conducted under three preprocessing scenarios: removing emojis and emoticons, replacing them with tag words using an in-built emoji dictionary, and replacing them with sentiment-based words using our customized emoji dictionary, EMOJIXT[1] copyrighted under Indian copyright act. This study explores how each emoji-handling approach influences sentiment classification performance.

The remaining sections of the article are arranged as follows: "Related works" Section presents an overview of related works. "Proposed approach" Section mentions the proposed approaches. Experiments conducted for validation have been described in "Experiments and result analysis" section followed by conclusion and future scope in "Conclusion and future scope" section.

Related works

This section reviews SA in the airline sector, focusing on customer reviews and tweets using ML[9,10] and deep learning (DL)[10–13]. As businesses move online, e-commerce[14] relies on NLP for large data volumes. While some studies address SA in aviation, our work is the first to include emojis and emoticons in airline reviews, a novel approach not previously explored in this context.

Alfreihat et al.[2] created an Emoji Sentiment Lexicon (Emo-SL) for Arabic tweets, improving sentiment classification by integrating emoji features with ML. Using 58,000 Arabic tweets, they calculated sentiment scores for 222 common emojis based on their distribution in positive and negative categories.

Suman et al.[15] broadened the scope of attention mechanisms by incorporating emoticon information into their model, resulting in significant performance improvements. However, it's important to recognize that conventional sentence-level attention mechanisms face challenges when applied to short text analysis.

Shiha and Ayvaz[16] proposed an emoji-based SA approach using Twitter data. Their study had two main objectives: to assess emotions during New Year's Eve celebrations, finding mostly positive sentiments, and to analyze tweets about the December 31, 2016, Istanbul attack, revealing predominantly negative feelings.

Liu et al.[17] used CEmo-LSTM to analyze sentiment during the COVID-19 pandemic using Weibo data. They found that the epidemic caused more passive emotions, like fear and grief. Their advanced emoji-embedding algorithm effectively integrated emojis with the SA model, enhancing emotion-mining tasks.

Li et al.[18] proposed a SA approach using micro-blog reviews that include emoji. The authors introduced an innovative technique for emoji vectorization to convert emojis into vectors. They then employed a Bi-LSTM model that integrates both emoji and text data for sentiment detection, achieving superior performance with their classification model.

Fernández-Gavilanes et al.[19] used resources like Emojipedia, Emojis. Wiki, CLDR annotations, and iEmoji to develop an unsupervised method for evaluating emoji sentiment lexicons from online corpora. They also generated a joint lexicon[20], calculating emoji sentiment by averaging scores from the unsupervised analysis. Their joint lexicon showed highly promising performance.

Surikov et al.[21] used emojis, emoticons, and punctuation to express emotions in text. They developed models and a lemmatizer to classify text as positive or negative using the RuSentiment corpus, outperforming traditional models by 6%. The best results, with 91% accuracy and a 0.937 ROC curve area, came from combining Word2vec and emotional indicators.

Prabhakar et al.[22] used Adaboost for SA of tweets about top US airlines, evaluating different ML algorithms and effectiveness via confusion matrix and accuracy metrics. Patel et al.[23] focused on BERT for SA of airline reviews, comparing its performance with other ML and DL methods, including Random Forest (RF) as a baseline.

Rane and Kumar[24] used ML to analyse sentiment in tweets about top US airlines, employing seven classifiers for comparison. Meanwhile, Rahat et al.[25] collected 10,000 airline tweets, preprocessing with NLTK, and used Naive Bayes (NB) and Support Vector Machine (SVM) classifiers for SA, excluding emojis and emoticons during preprocessing.

Subba and Kumari[26] used multiple word embedding techniques for SA on four reviews and Twitter datasets. They ensembled DL classifiers and validated their framework across multiple datasets, achieving high performance.

Table 1 summarized a comparative analysis SA frameworks proposed in literature Various authors have developed SA approaches for airline data, often treating emojis and emoticons as noise and removing them. This article, however, focuses on improving customer satisfaction by incorporating emojis and emoticons into the SA process. It also addresses data imbalance with effective data augmentation techniques to reduce bias and overfitting.

Proposed approach

The proposed work is based on a research study that has been executed on customer reviews[27] for the aviation sector. The choice of the dataset is mainly taking to the dynamic and competitive market of the aviation sector that has expanded at an exponential rate over the past 20 years[24,28]. In order to determine the demands and

[1] © 2024 Dipankar Majumdar, Amit Khan, Bikromadittya Mondal. "EMOJIXT: TRANSFORMING EMOJIS INTO TEXT" published under the Indian Copyright Act.

Methods	Feature Extraction Techniques	Advantages	Limitations	Proposed By
Emoji Sentiment Lexicon (Emo-SL) and ML ((SVM, K-nearest neighbors (KNN), RF, NB))	Emoji Feature, Parts Of Speech (POS) tagging, Positive and negative word count.	Improves sentiment classification by integrating emoji features.	Limited to emoji-based sentiment, language-dependent. The Lexicon approach may not fully capture dialectal and informal variations.	Alfreihat et al.[2]
LSTM + Modality Attention Mechanism for fusing two modalities (Text and Emoji)	Emoji Features, Text features, GloVe, Emoji2vec, Attention Mechanism	Enhances SA by incorporating emoji information	The system does not currently use advanced word embeddings like Word2Vec, FastText, or BERT, which could enhance feature extraction.	Suman et al.[15]
Emoji-based SA Using Lexicon (SentiWordNet) Based Approach.	Emoji Lexicon and SentiWordNet Lexicon. Sentiment score from the above lexicons are used as features.	Assesses emotions in specific events using emoji sentiment	Event-specific, may not generalize well to other contexts. Lexicons may not cover all the emoji/emoticons.	Shiha and Ayvaz[16]
Chinese emoji-embedding LSTM model (CEmo-LSTM)	LSTM emoji and text embeddings	Effectively integrates emojis with SA models, and enhances emotion mining.	Requires large datasets, limited to Weibo-based analysis	Liu et al.[17]
Emoji-text integrated bidirectional LSTM (ET-BiLSTM)	Bi-LSTM Emoji Vectorization.	Superior performance by integrating emoji and text data.	Requires robust emoji representation, computationally intensive	Li et al.[18]
Unsupervised Emoji Sentiment Lexicon	Online Emoji Resources (Emojipedia, CLDR, etc.)	Generates a joint sentiment lexicon from multiple sources	Limited by the accuracy of external resources, lacks contextual understanding.	Fernández-Gavilanes et al. [19]
Combination of ML and semantic-orientation approaches	Lexicon features, Standford POS tagger, Uni-grams, Bi-grams	The approach successfully integrates features from both semantic orientation and ML, leading to better performance.	Adding multiple features might lead to overfitting, especially if the dataset is small or not diverse enough.	Dang et al.[20]
Recurrent Neural Network (RNN) with hidden LSTM layer.	TF-IDF, Word2Vec, Emotional Indicators	Achieved 91% accuracy, outperformed traditional models by 6%	Requires specialized preprocessing and corpus-specific adaptation	Surikov et al.[21]
AdaBoost	TF-IDF, Bag of Words, Feature selection	Effective sentiment classification for airline tweets	Dependent on dataset quality and model tuning	Prabhakar et al.[22]
BERT, ML (SVM, RF, DT, NB)	BERT base embeddings, TF-IDF, Word2Vec	Obtained remarkable performance.	For BERT, it requires large computational power.	Patel et al.[23]
ML (DT, RF, SVM, GNB, LR, KNN, AdaBoost)	Doc2vec	Captures semantic meaning, generalises well	High computational cost, sensitive to hyperparameters, less effective to unseen data	Rane and Kumar[24]
Naïve Bayes & SVM for Airline Tweets	BoW, TF-IDF, NLTK Preprocessing	Effective with well preprocessed data	Excludes emojis and emoticons; may lose contextual sentiment.	Rahat et al.[25]
Stacking Ensemble Classifier (LSTM, GRU, Bi-GRU).	Word2Vec, GloVe, BERT Embeddings,	High performance with heterogeneous stacking ensemble	Requires ensemble tuning and high computational cost	Subba and Kumari[26]

Table 1. A summarized comparative analysis of various SA frameworks proposed in the literature.

preferences of travellers, comprehensive feedback is necessary. This had been a very challenging task a few decades back. However, with the advent and massive growth of social networking platforms such as Facebook, Twitter, and other platforms, feedback acquisition has now become easier[29]. After acquisition, the analysis of the feedback is executed. This helps predict the following:

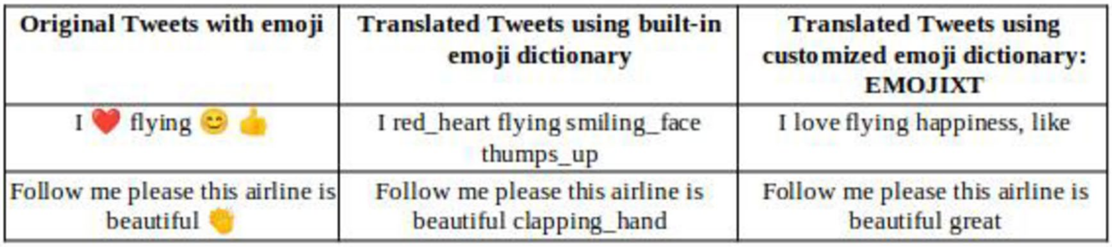
- Reaction of the customer towards the product or service.
- Satisfaction level of the customer with the price and quality of service.

With the help of 3 predefined categories—positive, negative, or neutral—SA seeks to categorize various review texts acquired from social media[30,31]. In recent times, the analysis and categorization have become far more challenging than earlier. This is due to the large-scale usage of multimodal data viz. Emojis, emoticons, symbolic texts, etc[6,7,32,33]. Emojis are tiny graphic symbols or Unicode characters like smiley faces. They are employed to deliver information or to express the person's feelings. Emojis are supported more by mobile phone technology, creating a platform for a visually appealing and engaging communication environment[34].

In this paper, we present a novel approach to efficiently utilize the effect of emojis/emoticons present in the text for SA. We have used a transformation model to substitute the emoji/emoticons with appropriate textual representations. The model is implemented using both the inbuilt dictionary as well as our own proposed dictionary, "EMOJIXT," customized for context. An example of the replacement of emojis using both dictionaries is shown in Fig. 1.

The said emoji dictionary has been designed such that sentimental words can be substituted synonymously against the emojis present in the dataset. We have used multiple ML approaches for their simplicity and the pre-trained BERT-base transformer model for fine-tuning due to the popularity and effectiveness of the BERT model in different NLP tasks, including text classification, to accomplish the multilevel classification objective.

Further, we have also noticed an imbalance in the dataset. This was difficult to work with due to the presence of skewed and biased records. Hence, the dataset had to be augmented for the elimination of such skewness and bias. This helped us with effective training and validation. We have used the Synthetic Minority Oversampling Technique (SMOTE)[35] to address our problem. The block diagram of our proposed model is shown in Fig. 2.



Original Tweets with emoji	Translated Tweets using built-in emoji dictionary	Translated Tweets using customized emoji dictionary: EMOJIXT
I ❤️ flying 😊 👍	I red_heart flying smiling_face thumps_up	I love flying happiness, like
Follow me please this airline is beautiful 👏	Follow me please this airline is beautiful clapping_hand	Follow me please this airline is beautiful great

Fig. 1. Sample tweets and their translation using in-built and customized emoji dictionary EMOJIXT.

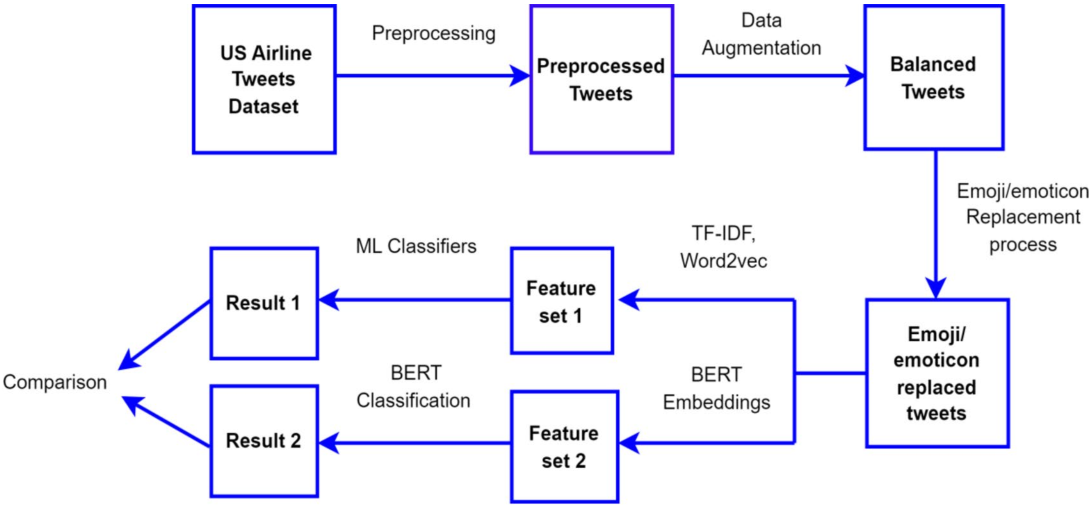


Fig. 2. Block Diagram of the proposed work.

Basic text processing operations performed

Preprocessing operations are essential for cleaning the data and making it suitable for feature extraction. We applied basic preprocessing techniques such as tokenization, lemmatization, stemming, and stop-word removal to the dataset, as discussed below.

Tokenization

Tokenization is the way to break the texts or sentences in smallest possible meaningful units. For example, suppose we have a sentence: The movie is interesting and I like it. If we pass the sentence through a tokenizer the result will be a set tokens: "The", "movie", "is", "interesting", "and", "I", "like", "it". For our proposed worked we have used both NLTK tokenizer and BERT-base tokeinzer.

Stemming

Stemming is mainly used to reduce the word length to make it a root form without considering the context of the word used in the text. For example, calls, called, and calling are all transformed into the word 'call.

Lemmatization

It is another way to normalize text faster than stemming. This method performed normalization by consideration of the context of the word within the text. For example, finally, final, and finalized are all transformed into the word "final". The concepts of both stemming and lemmatization are almost the same.

Stop word removal

All the tokens do not convey contextual or meaningful information during SA. For example, articles (a/an/the) and prepositions (to, about, over, etc.) are not carrying useful information. They are called stop words. We have eliminated the stop words using NLP toolkits (NLTK).

Removal of digits and special symbols

Some of the reviews contain digits and special symbols. These digits and symbols are removed during the pre-processing phases as they have no significance in SA. For proposed work, we initially retain the emojis/emoticons and remove or replace those using the proper strategies adopted in this study.

Removal of punctuation
The dataset contains some punctuation characters. These punctuations do not contribute anything during SA and are removed during pre-processing. We did this with the help of a regular expression string module.

Feature extraction methods

The pre-processed data is further processed to extract suitable features, which are then fed into the algorithms to obtain the final result. For our different ML classifiers, we used Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec features, while for the BERT model, we utilized BERT's embedding techniques. The different feature extraction techniques are discussed below.

TF-IDF
TF-IDF indicates the importance of a particular term within a particular document. TF-IDF can be calculated as follows:

$$\mathrm{TF-IDF} = \mathrm{TF}\ {*}\mathrm{IDF} \tag{1}$$

Where, TF is Term Frequency; IDF is Inverse Document frequency; t is term ; d is document; N is count of corpus ; corpus is total document set TF is individual to each document and word; hence TF can be written as follows:

$$\mathrm{TF}(\mathrm{t},\ \mathrm{d}) = (\mathrm{Number\ of\ \ t\ in\ d})\ /\ (\mathrm{word\ count\ in\ d}) \tag{2}$$

$$\mathrm{DF}\ (\mathrm{t}) = \mathrm{occurrence\ of\ \ t\ \ in\ N\ documents} \tag{3}$$

$$\mathrm{IDF}\ (\mathrm{t}) = \mathrm{N/DF} = \log(\ \mathrm{N}\ /\ (\ \mathrm{DF}\ +\ 1\ )) \tag{4}$$

Using the Eqs. (2) and (4), we can rewrite the Eq. (1) as below

$$\mathrm{TF-IDF}(\mathrm{t},\ \mathrm{d}) = \mathrm{TF}(\mathrm{t},\ \mathrm{d}) * \log\ (\ \mathrm{N}\ /\ (\ \mathrm{DF}\ +\ 1\ )) \tag{5}$$

Word2Vec
We have also used a very popular word embedding technique called Word2Vec with dimension 300 to extract features. The Word2Vec model works based on 2 major architectures, i.e. Continuous Bag Of Words (CBOW) and Skip-gram architectures. Figure 3 depicts the architecture of the CBOW and Skip-gram models. CBOW predicts the target word using context words. Probability: P(w_t | context) and Loss: – log P(w_t | context) and Skip-gram Predicts context words using the target word. Probability: P(context | w_t) and Loss: –∑ log P(context | w_t). Here w_t is the target word.

BERT embeddings
Before embedding, the BERT tokenizer generates the tokens. Then each token is mapped to a unique ID from BERT's vocabulary. After that, these IDs are passed into BERT for generating embeddings that are dense numerical vectors. Finally, BERET will apply a self-attention mechanism to adjust the vector based on the context. The below example shows how BERT embedding is created.

Input Sentence: I like to travel by plane.

Tokenized Output: ['[CLS]', 'I', 'like', 'to', 'travel', 'by', 'plane', '.', '[SEP]']

Convert Tokens to Token IDs : [101,146,2066,2000,4442,2011,4946,1012,102]

Convert Token IDs to embedding: [[0.12, -0.34, 0.56, …], [0.45, -0.23, 0.67, …] …….

Contextual Embedding: Here word plane may convey two meanings either aircraft or flat surface. So based on the context BERT modifies the embedding.

Here two special tokens are used CLS and SEP for classification task and separation of sentences respectively.

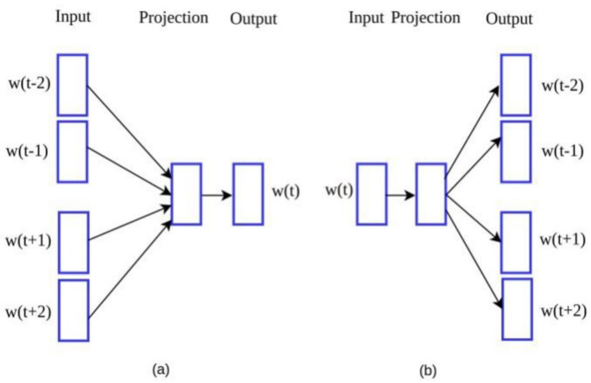


Fig. 3. (**a**) CBOW Model. (**b**) Skip-gram Model.

Variants of the proposed approach

We have addressed our main objectives in three variants. The details of the variants are discussed in the following subsections.

Variant I: without using emojis and data augmentation

Here we have not considered the effect of emojis/emoticons during SA. Data imbalance problems are also not addressed in this variant. Here, emojis/emoticons are considered as noises, and all the emojis and emoticons were discarded from the dataset before actual data preprocessing started. During actual pre-processing, operations such as tokenization, lemmatization, stemming, and stopword removal are performed. TF-IDF and Word2Vec features are separately extracted and used to train different ML classifiers. Their performance is evaluated using test data and various metrics. We also fine-tuned our work with a pre-trained BERT base model, as shown in Fig. 4. Hyperparameters for fine-tuning are detailed in Sect. 6.2.2, with the workflow depicted in Fig. 5.

Variant II: with data augmentation and Emojis replace by in-built Emoji dictionary

Variant II incorporates the impact of emojis and emoticons in SA. Emojis are substituted with corresponding tag words from an inbuilt emoji dictionary[36] tailored for various NLP tasks. Addressing data imbalance is tackled using SMOTE[35], a widely recognized data augmentation technique. Subsequent preprocessing steps include tokenization, lemmatization, stemming, and stop word removal on the balanced dataset to extract relevant features.

The extracted features are then used to train different ML classifiers, and their performance is evaluated using various indicators. Additionally, we fine-tuned our approach using a pre-trained BERT base model. Figure 5 visually represents the entire framework of this variant.

Variant III: with data augmentation and emojis replaced by customized dictionary EMOJIXT

In Variant III, we've developed a specialized emoji dictionary called EMOJIXT where each emoji is associated with corresponding sentiment words, totalling 250 emojis. Using this dictionary, emojis in our dataset are replaced with these words sourced from the World Wide Web (WWW). Following this, we balance the dataset using advanced data augmentation techniques similar to Variant II. Next, the processed dataset undergoes basic preprocessing steps, including tokenization, lemmatization, stemming, and stop word removal. Feature extraction is then applied to train various ML classifiers, and their performance is evaluated using diverse metrics. Additionally, Variant III incorporates fine-tuning with a pre-trained BERT base model to enhance our research goals, as illustrated in Fig. 5.

ML based classifiers

We have employed different ML classifiers like SVM, RF, Multinomial Naive Bayes (MNB), Logistic Regression (LR), etc. We have performed the experiment using all the aforesaid classifiers, but we have considered two best-performing classifiers for comparison purposes. From the experimental result, it is clear that RF and MNB

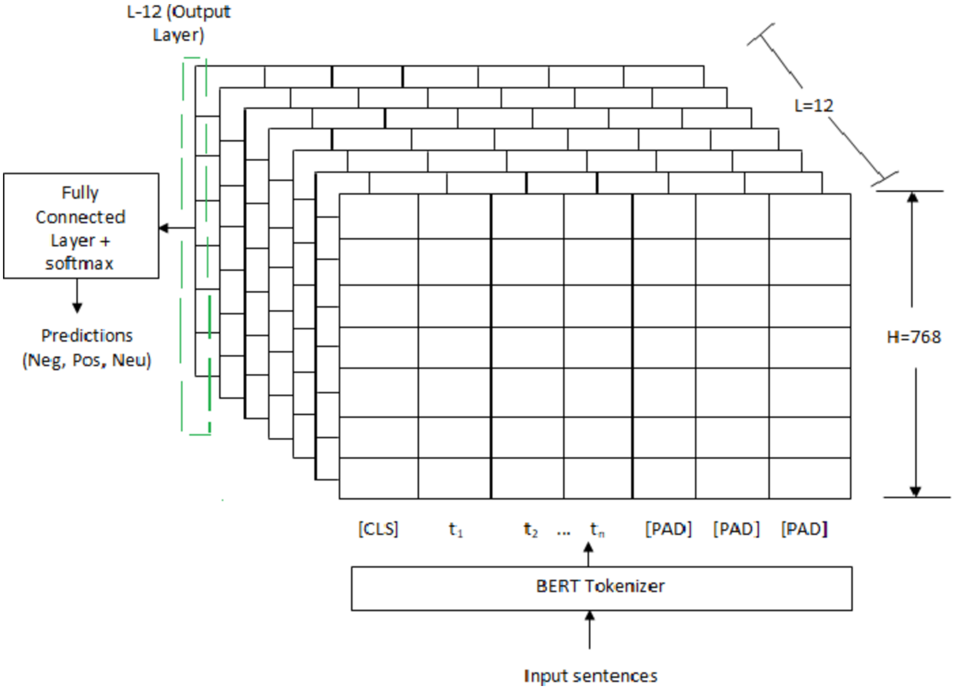


Fig. 4. Architecture of BERT-base Model.

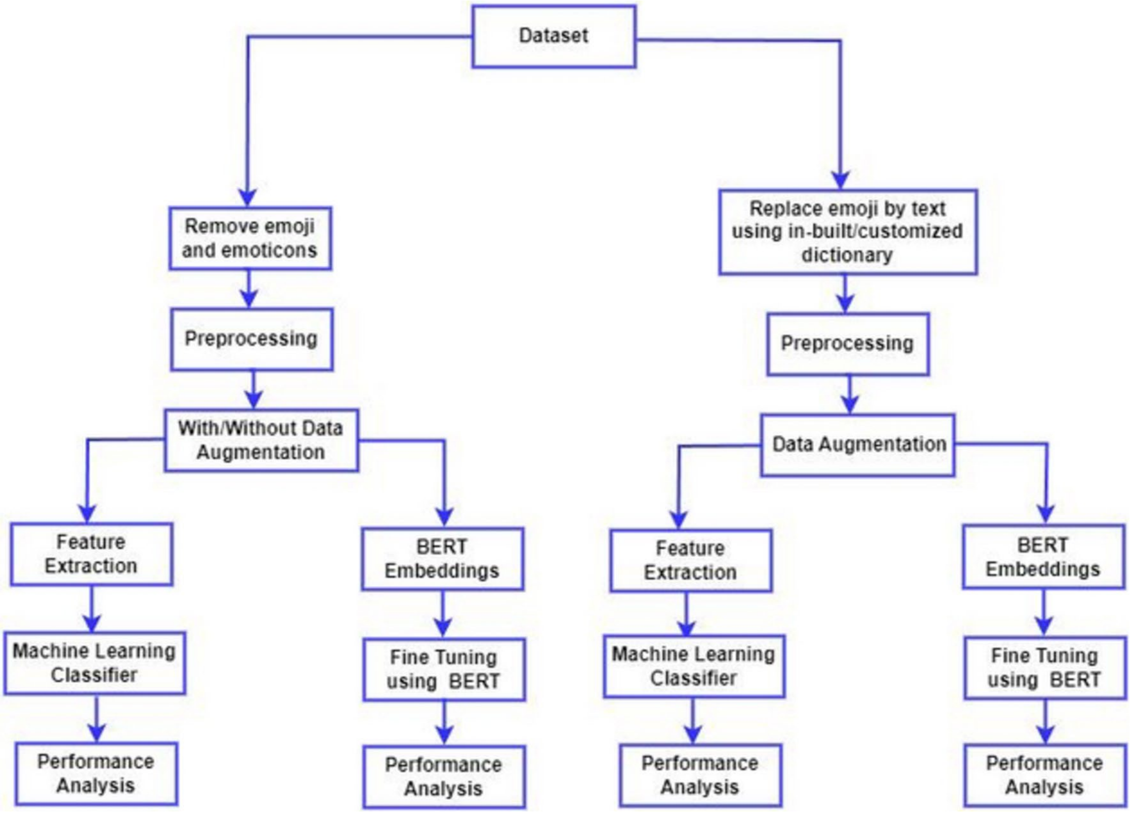


Fig. 5. Working Principle of Variant I, II and III.

classifiers performed best. For that, we performed all the comparisons based on different performance metrics as discussed in the result and discussion section.

LR

It works based on a supervised approach. LR is a simple and effective technique that works very well on TF-IDF or word2vec inputs, making it effective for SA with emojis. We have used multinomial LR (MLR) to solve multi-class classification problems. The model is adapted to learn and predict a multinomial probability distribution function. In symbols, a multinomial distribution requires a system that has k possible outcomes (X_1, X_2, X_3,. . ., X_k) with associated probabilities (p_1, p_2 ,p_3,. . ., p_k) such that $\Sigma\, p_i = 1$. For 'n' number of repeated trials, let x_i be the number of times the result X_i is generated, subject to constraints $0 \le x_i \le n$ and $\Sigma\, x_i = n$. The joint probability density function can be represented as

$$\mathrm{P(X1\ =\ x1, X2\ =\ x2, ., Xk = xk\ )} = \frac{\mathrm{n!}}{x_1!\mathrm{x}_2!\cdots x_k!}p_1^{x_1}p_2^{x_2}\cdots p_k^{x_k} \tag{6}$$

NB

NB is a very strong technique for textual data analysis and addresses multi-class classification problems. The NB classifier is inspired by the Bayes theorem. Bayes's theorem is stated in Eq. (7).

$$P\left(A/B\right) = \frac{P\left(B/A\right) * P\left(A\right)}{P\left(B\right)} \tag{7}$$

Where we are finding the probability of A when the probability of B is given. P(A) and P(B) are the prior probabilities of class A and class B, respectively. P(B/A) is the occurrence of predictor B when the probability of class A is known. There are many variations of Naïve Bayes, i.e., Bernoulli, Multinomial, and Gaussian Naive Bayes (GNB). Our study used an MNB classifier, where feature vectors represent the number of times a term appeared, i.e., frequency. MNB is very fast, efficient and effective for text classification and handles high dimensional sentiment data well including emojis/emoticons. For SA, the equation can be rewritten as Eq. (8).

$$P\left(\text{words/sentiment}\right) = \frac{P\left(\text{sentiment/words}\right) * P\left(\text{sentiment}\right)}{P\left(\text{words}\right)} \tag{8}$$

SVM

This works by finding the best possible hyperplane from the set of possible hyperplanes to distinguish two sets of patterns. In the proposed work, SVM works to solve multi-class classification problems by constructing and combining several binary classifiers. SVM works fine by determining complex decision boundaries and also good for small feature sets. The SVM optimization with soft constraints is given as

$$\min\frac{1}{2}||w||^2+\mathrm{C}\sum_{\mathrm{i}=1}^{l}t_i \tag{9}$$

$$\text{Subject to } y_i\,(\mathrm{wx}_i+\mathrm{b})+\mathrm{t}_i\geq\ 1,\mathrm{t}_i\geq\ 0,\mathrm{i}=1,2,3,\cdots.l \tag{10}$$

RF classifier

As an ensemble ML method, RF classifies unknown samples using predictions from a number of classifiers that have each been trained separately. RF is very effective for high dimensional text features like TF-IDF or word2vec and also reduces overfitting by averaging multiple decision trees, making it useful for noisy sentiment data like tweets. Multiple decision trees form the foundation of the RF algorithm. It is possible to use the bootstrap aggregating method, in which different decision trees are fitted to different subsets of the training data after being sampled with replacement. Additionally, to lessen the association between trees, the splits at each node in each decision tree are determined using the random subspace approach, which only considers a portion of all the available data. The averages of individual trees are then determined.

BERT model

BERT is a large pre-trained language model[8] for performing various NLP tasks. The BERT model excels in SA by capturing context-aware meanings of words and emojis, leveraging large text corpora for high accuracy and robustness. It's bidirectional processing aids to grasp the true meaning of text, significantly improving performance. For our study, we have fine-tuned the model for SA.

BERT architecture

There are many variations of BERT architecture. Here we have used the BERT-base architecture for simplicity. BERT-base's architecture is built on Transformers, employing 12 layers of Transformer encoders, and each layer contains 768 hidden units, as shown in Fig. 4.

Each encoder layer includes a multi-head self-attention mechanism and a feed-forward neural network, enabling the model to capture bidirectional context and intricate word relationships in text. For SA, the pre-trained BERT model is fine-tuned on labeled sentiment datasets. During this process, the [CLS] token, which aggregates the sequence information, is used for classification. Fine-tuning adjusts the model parameters to enhance sentiment prediction accuracy. The BERT-base model's robust architecture and comprehensive pre-training make it exceptionally effective for understanding and analyzing sentiment in textual data. We have set hyperparameters for fine-tuning our proposed work as follows:

- Input token length: 80.
- Batch Size: 16.
- No. of epochs: 10.
- Learning rate: 3e-5.
- Optimizer: Adam.

Pseudo code for sentiment prediction using BERT-base model

Input:

- S: Input sentence.
- V: Vocabulary.
- E: Embedding matrix.
- y: True label.
- W, b: Weights and bias for classification.

Output:

- ŷ: Predicted probabilities.
- ζ: Cross-entropy loss.
- Performance matrices: accuracy, F1-score, precision, recall etc.

Steps:

1. Tokenization: Tokenize the input sentence S into tokens T=[token$_1$, token$_2$,…,token$_m$].
2. Token ID generation: Map tokens to token IDs using the vocabulary V: ID$_T$=[ID$_1$,ID$_2$,…,Id$_m$].
3. Attention Masks: Generate attention masks M=[mask$_1$,mask$_2$,…,mask$_m$]: mask$_i$=1 if token$_i$ is a real token. mask$_i$=0 if token$_i$ is a padding token.
4. Embedding : Obtain token embeddings E$_T$ = [embedding$_1$,embedding$_2$,…,embedding$_m$]: embedding$_i$ =[Id$_i$].
5. BERT Encoding:

- Initialize H$^{(0)}$ = E$_T$.

- For each layer l from 1 to 12 (number of BERT layers for BERT base = 12): $H^{(l)} = \text{BERTLayer}^{l} (H^{(l - 1)}, M)$.
- Output the final hidden states $H = H^{(12)}$.

6. Self-Attention Mechanism:

- Compute attention scores using query (Q), key (K), and value (V) matrices:
- $\text{Attention}(Q, K,V) = \text{softmax}\left(\frac{QK^{T}}{\sqrt{d_k}}V\right)$

7. Classification Layer:

- Extract the hidden state corresponding to the [CLS] token: $h_{CLS} = H_{CLS}$.
- Compute the logits: $\text{logits} = Wh_{CLS} + b$.

8. Softmax and Loss Calculation:

- Apply softmax to compute predicted probabilities: $\hat{y}=\text{softmax}(\text{logits})$.
- Compute the cross-entropy loss: $\zeta = - \Sigma^{C}_{i=1} y_i \log(\hat{y}_i)$ where y is the true label and C is the number of classes.

9. Evaluate performance metrics such as accuracy, F1-score, precision, and recall based on ŷ and y.

Experiments and result analysis

In order to verify the effectiveness of our proposed method, we have used the Twitter US Airline Sentiment dataset[27] to perform experiments on different variants. The dataset is collected from Kaggle. The dataset contains information related to the feelings or opinions of passengers traveling through airlines. A total of 14,640 reviews are there.

Experimental settings

We have categorized the reviews into three types of sentiment classes, namely positive, neutral, and negative. The distribution of the dataset into different sentiment classes and the number of tweets with or without emojis are shown in Fig. 6. Out of 14,460 reviews, 794 (around 6%) reviews contain one or more emojis or emoticons.

Our experiment utilized both ML and DL techniques. For the ML part, we employed MNB, RF, SVM, and LR classifiers, while for DL, we used the BERT-base transformer model. According to our proposed methodology, the experiment was conducted with three different variants. In ML, we used both TF-IDF and Word2Vec features for the experiments. For the BERT-base transformer model, the experiment was conducted based on the algorithm detailed in "Word2Vec" Sect. In the case of ML approaches, we split the dataset into training and testing sets with an 80:20 ratio. On the other hand, in the case of the BERT model, we divided the dataset into training, testing, and validation sets with an 80:10:10 ratio. Hyperparameters for fine-tuning the BERT-base model are also enlisted in "Word2Vec" Sect. Besides, we use classification, precision, recall, and F-1 score as the evaluation metrics. The evaluation metrics are defined as follows:

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

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

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

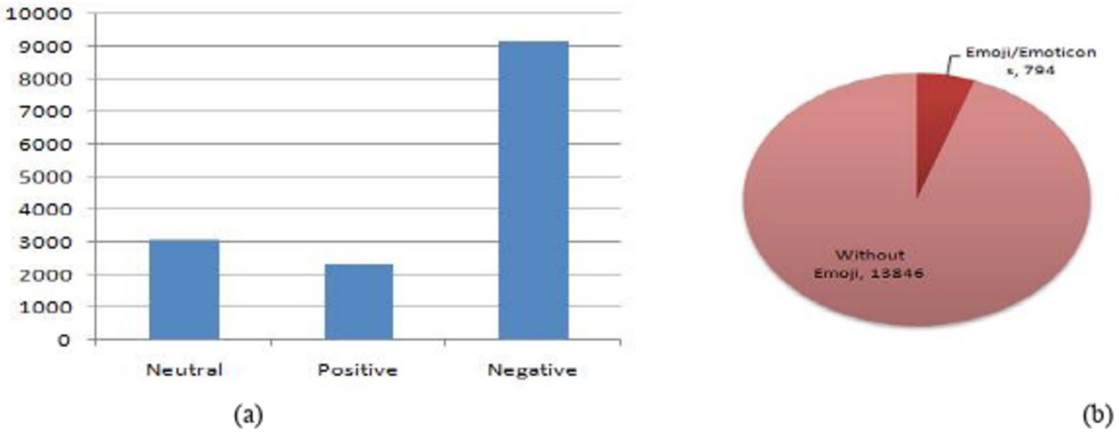


Fig. 6. (**a**) Distribution of dataset into different classes. (**b**) Dataset instances with /without emoji.

$$\text{F-1 Score}=\frac{2*Precision*Recall}{\text{Precision}+\text{Recall}}$$

Where the classification output given by the classifier algorithms are as follows:
TP (True Positive): Count of Positively Labeled Records classified as Positive.
TN (True Negative): Count of Negatively Labeled Records classified as Negative.
FP (False Positive): Count of Negatively Labeled Records classified as Positive.
FN (False Negative): Count of Positively Labeled Records classified as Negative.

Experimental results

Comparison of the performances of the various ML algorithms and BERT model with and without emojis and emoticons on balanced and im-balanced airline reviews dataset is done. We did our experiment using SVM, MNB, RF, and LR classifiers, and BERT model using different variants as discussed in "Proposed approach" Sect. In the case of the BERT-base model, our dataset is divided into three different parts training, validation, and testing. The validation part is mainly for the fine tuning process and the testing part is for evaluating the performance of the final model. Table 2 shows the comparison among various ML models and the BERT base model on different variants. Among the ML techniques, the RF classifier provides the best performance when we consider both emojis and data augmentation as depicted in Table 2. We obtained 85% accuracy in both variants II and III using the RF classifier whereas the MNB classifier is selected as second best ML classifier, and obtained 80% and 81% accuracy from it in Variant II and Variant III respectively. On the other hand, we achieved 91% and 92% accuracy from the BERT base model in Variant II and Variant III respectively.

From Table 2 it is clear that when we consider emojis and data augmentation during SA on the airline dataset, the RF classifier performs best among other ML classifiers in terms of accuracy and other performance metrics. BERT base model also performed very well whenever we considered emojis and data augmentation in both Variant II and III. BERT base performs slightly better in Variant III in terms of the different performance metrics as shown in bold in Table 2. Now we compared our best model performance obtained from both RF and BERT model with some other existing works. It has been seen from the obtained outcome that the proposed model significantly outperformed the state-of-the-art works. The performance comparison is shown in Table 3.

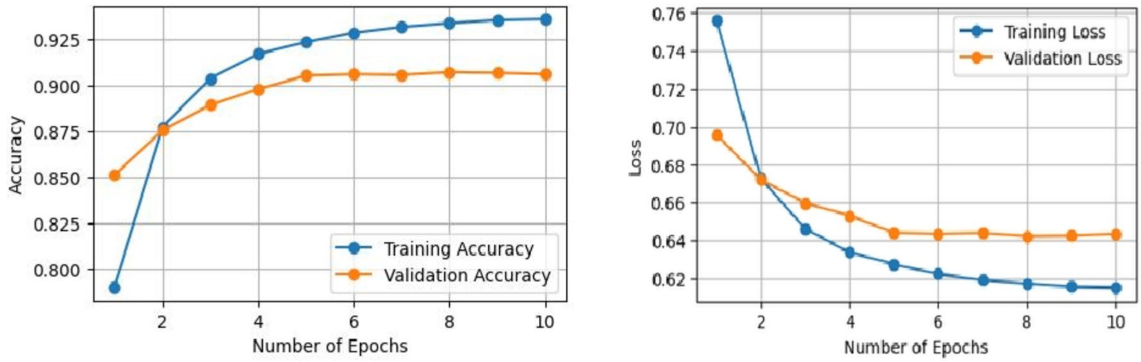
Figure 7(a) shows the training accuracy and validation accuracy comparison of the BERT-base model on Variant II and 7(b) shows the training loss and validation loss comparison of the BERT-base model on Variant II. Figure 8(a) shows the training accuracy and validation accuracy comparison of the BERT-base model on Variant III and 8(b) shows the training loss and validation loss comparison of the BERT-base model on Variant III. The training accuracy comparison of variant II and variant III for the BERT-base model is shown in Fig. 9(a) and the validation accuracy comparison of variant II and variant III for the same model is depicted in Fig. 9(b). Similarly, the training loss comparison on variant II and variant III for the BERT-base model is shown in Fig. 10(a), and the corresponding validation loss comparison for the same model is depicted in Fig. 10(b). From the figures, it is obvious that the BERT-base model on variant III significantly outperforms the other variants and models.

		Precision			Recall			F-1 score		
Model	Accuracy	Positive	Negative	Neutral	Positive	Negative	Neutral	Positive	Negative	Neutral
SVM(Variant I)	72%	0.78	0.77	0.66	0.52	0.95	0.38	0.62	0.85	0.48
SVM(Variant II)	74%	0.78	0.74	0.70	0.81	0.78	0.63	0.79	0.76	0.67
SVM(Variant III)	74%	0.77	0.73	0.73	0.83	0.78	0.63	0.80	0.76	0.67
LR(Variant I)	73%	0.76	0.76	0.64	0.49	0.94	0.36	0.60	0.84	0.46
LR(Variant II)	75%	0.77	0.73	0.68	0.79	0.75	0.65	0.78	0.74	0.67
LR(Variant III)	75%	0.75	0.74	0.68	0.79	0.76	0.64	0.77	0.75	0.67
RF(Variant I)	73%	0.69	0.61	0.53	0.76	0.91	0.39	0.60	0.83	0.47
RF(Variant II)	85%	0.88	0.85	0.82	0.89	0.81	0.85	0.88	0.83	0.83
RF (Variant III)	85%	0.90	0.86	0.81	0.90	0.80	0.85	0.90	0.83	0.83
MNB (Variant 1)	67%	0.84	0.66	0.73	0.17	0.99	0.14	0.28	0.79	0.24
MNB (Variant II)	81%	0.86	0.74	0.84	0.87	0.87	0.69	0.87	0.80	0.76
MNB (Variant III)	80%	0.85	0.72	0.83	0.86	0.85	0.68	0.86	0.85	0.78
BERT (Variant I)	85%	0.74	0.89	0.76	0.78	0.93	0.63	0.76	0.91	0.69
BERT (Variant II)	91%	0.93	0.89	0.91	0.94	0.91	0.87	0.93	0.90	0.89
BERT (Variant III)	**92%**	**0.95**	**0.91**	**0.94**	0.94	**0.94**	**0.88**	**0.95**	**0.92**	0.89

Table 2. Performance comparison on different variants. Significant values are in bold.

Model	Accuracy	Precision			Recall			F-1 score		
		Positive	Negative	Neutral	Positive	Negative	Neutral	Positive	Negative	Neutral
RF (Variant II)	85%	0.88	0.85	0.82	0.89	0.81	0.85	0.88	0.83	0.83
RF (Variant III)	85%	0.90	0.86	0.81	0.90	0.80	0.85	0.90	0.83	0.83
BERT (Variant II)	**91%**	**0.93**	**0.89**	**0.91**	**0.94**	**0.91**	**0.87**	**0.93**	**0.90**	**0.89**
BERT (Variant III)	**92%**	**0.95**	**0.91**	**0.94**	**0.94**	**0.94**	**0.88**	**0.95**	**0.92**	**0.89**
BERT[23]	83%	0.78	0.85	0.79	0.77	0.96	0.46	0.78	0.90	-
LSTM[28]	76%	0.72	0.83	0.74	0.72	0.77	0.78	0.72	0.80	-
Roberta[30]	80.08%	-	-	-	-	-	-	-	-	-
Electra[30]	79.80%	-	-	-	-	-	-	-	-	-

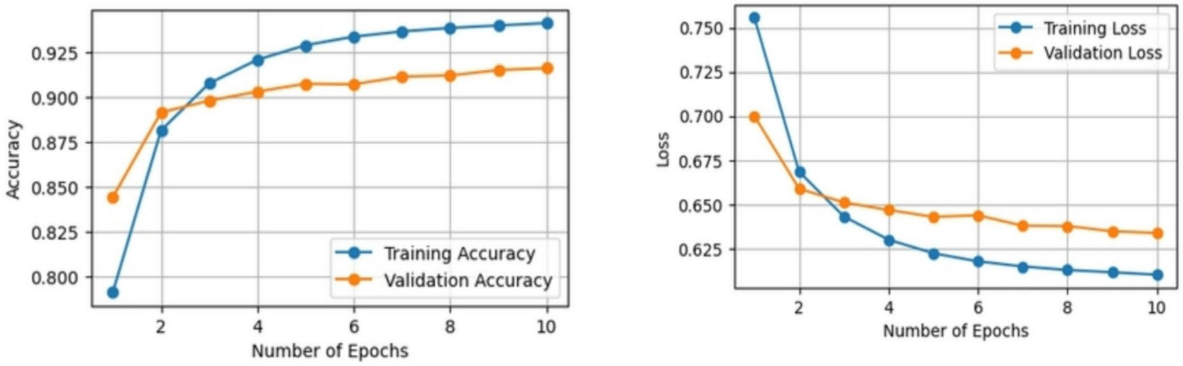
Table 3. Performance comparison with state of Art works. Significant values are in bold.



(a)

(b)

Fig. 7. (**a**) Accuracy comparison of BERT-base Model on Variant II (**b**) Loss comparison of BERT-base Model on Variant II.



(a)

(b)

Fig. 8. (**a**) Accuracy comparison of BERT-base Model on Variant III (**b**) Loss comparison of BERT-base Model on Variant III.

Conclusion and future scope

With an objective to measure the level of satisfaction that the customers get while utilizing the services provided by different airlines, the current paper principally concentrates on analysis of the review comments posted by them. The proposed model deals with the textual dataset along with the emojis and emoticons present in the airline reviews. The model considers them as vital inputs during feature extraction in order to boost the performance of the proposed SA work. For utilizing emojis we have used both an inbuilt emoji dictionary as

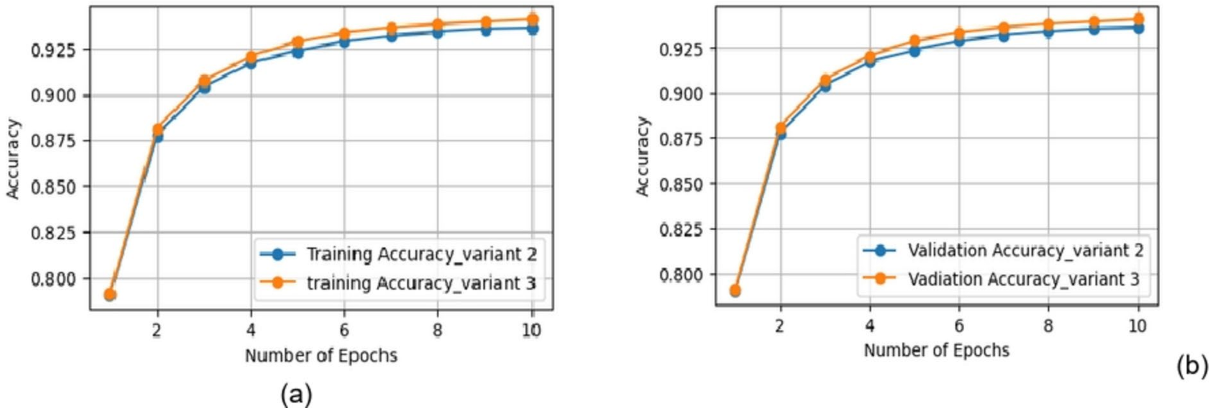


Fig. 9. (**a**) Training accuracy of BERT-base model on Variant II vs. Variant III (**b**) Validation accuracy of BERT-base model on Variant II vs. Variant III.

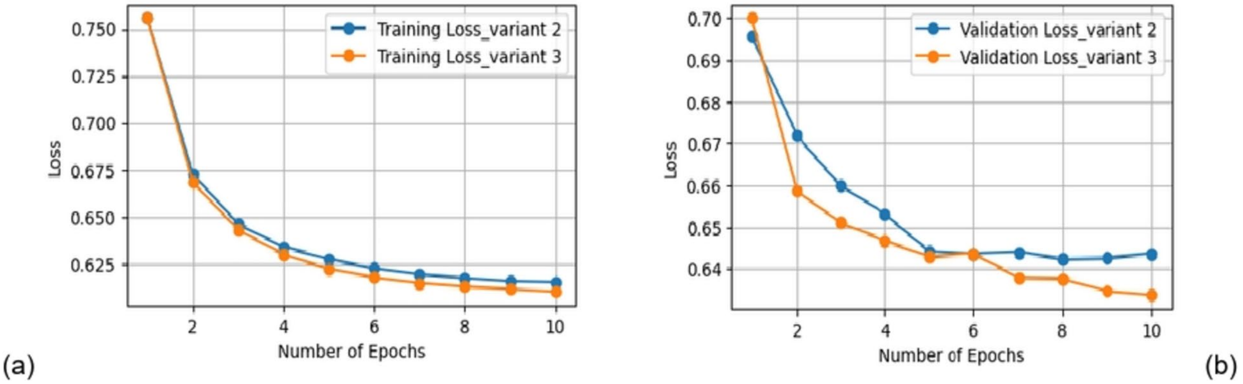


Fig. 10. (**a**) Training loss of BERT-base model on Variant II vs. Variant III. (**b**) Validation loss of BERT-base model on Variant II vs. Variant III.

well as our own customized emoji dictionary EMOJIXT. Here different machine learning classifiers and BERT base models were used to classify the sentiment of airline reviews into positive, negative, and neutral. The performance of different ML classifiers as well as the BERT model were compared and found RF and BERT as the best- performing models in terms of different performance metrics. We have also shown a comparison between two of the best-performing models used here i.e. RF and BERT with some recent state of the art works available in the literature. Our approach surpasses existing methods by approximately 2% with RF in Variant III and around 9% with BERT in Variant III. Since our work is limited to airline sector reviews only the future scope of the work includes expanding the approach using diverse datasets to make a more versatile model suitable for cross-domain analysis.

Data availability

Data for the present work is available in the link given below: https://www.kaggle.com/crowdflower/twitter-airline-sentiment.

Received: 7 November 2024; Accepted: 26 February 2025
Published online: 04 March 2025

References

1. Tanwar, G., Tripathi, S., Gaur, D. & Shakya, D. An approach to improve analysis of product reviews using sentiment analysis. In: Proceedings of the 2022 4th International Conference On Advances In Computing, Communication Control And Networking (ICAC3N). IEEE, Greater Noida, India, doi: (2022). https://doi.org/10.1109/ICAC3N56670.2022.10074049
2. Alfreihat, M. et al. Emo-SL Framework: Emoji Sentiment Lexicon Using Text-Based Features and Machine Learning for Sentiment Analysis, in *IEEE Access*, vol. 12, pp. 81793–81812, (2024). https://doi.org/10.1109/ACCESS.2024.3382836
3. Jain, S. & Singh, P. Systematic survey on sentiment analysis. In: Proceedings of the 2018 First International Conference On Secure Cyber Computing And Communication (ICSCCC).IEEE, Jalandhar, India, (2018). https://doi.org/10.1109/ICSCCC.2018.8703370

4. Rathan, M., Hulipalled, V., Murugeshwari, P. & Sushmitha, H. Every post matters: a survey on applications of sentimentanalysis in social media. In: Proceedings of the 2017International Conference On Smart Technologies For SmartNation (SmartTechCon). IEEE,Bengaluru, India, (2017). https://doi.org/10.1109/SmartTechCon.2017.8358463
5. Zitnik, M. Using sentiment analysis to improve businessoperations. *XRDS: Crossroads ACM Magazine Students.* **18** (4), 42–43. https://doi.org/10.1145/2173637.2173655 (2012).
6. Singh, G. V., Firdaus, M., Ekbal, A. & Bhattacharyya, P. Unity in diversity: multilabel Emoji identification in tweets. *IEEE Trans. Comput. Social Syst.* **10** (3), 1029–1038. https://doi.org/10.1109/TCSS.2022.3162865 (June 2023).
7. Zhao, Q., Wu, P., Lian, J., An, D. & Li, M. TaneNet: Two-Level Attention Network Based on Emojis for Sentiment Analysis, in *IEEE Access*, vol. 12, pp. 86106–86119, (2024). https://doi.org/10.1109/ACCESS.2024.3416379
8. Huang, J. et al. Jan. 'Incorporating Emoji Sentiment Information into a Pre-trained Language Model for Chinese and English Sentiment Analysis'. *Intelligent Data Analysis*1 : 1–25, (2024). https://doi.org/10.3233/IDA-230864
9. Pillai, R., Oza, P. & Sharma, P. Review of machine learning techniques in health care. In: Proceedings Of ICRIC 2019: Recent Innovations In Computing. 103–111 Springer International Publishing, 2020, (2020). https://doi.org/10.1007/978-3-030-29407-6_9
10. Ullah, A., Khan, S. & Nawi, N. Review on sentiment analysis for text classification techniques from 2010 to 2021. *Multimedia Tools Appl.* **82**(6), 8137–8193, doi: https://doi.org/10.1007/s11042-022-14112-3 (2023).
11. Hasib, K., Towhid, N. & Alam, M. Online review based sentiment classification on bangladesh airline service using supervised learning. In: Proceedings Of 2021 5th International Conference On Electrical Engineering And Information Communication Technology (ICEEICT),1–6. IEEE, (2021). https://doi.org/10.1109/ICEEICT53905.2021.9667818
12. Khan, A., MAJUMDAR, D., MONDAL, B. & MUKHERJEE, S. A deep learning approach to sarcasm detection from composite textual data. *INFOCOMP J. Comput. Sci.* **21**(2), (2022).
13. Zhong, Q. & Shao, X. 'A Cross-model Hierarchical Interactive Fusion Network for End-to-end Multimodal Aspect-based Sentiment Analysis'. *Intelligent Data Analysis*,1 Jan. : 1–16, (2024). https://doi.org/10.3233/IDA-230305
14. Yang, Z., Li, Q., Charles, V., Xu, B. & Gupta, S. Online product decision support using sentiment analysis and fuzzy Cloud-Based Multi-Criteria model through multiple E-Commerce platforms. *IEEE Trans. Fuzzy Syst.* https://doi.org/10.1109/TFUZZ.2023.3269741 (2023).
15. Suman, C., Saha, S. & Bhattacharyya, P. 'An attention-based multimodal Siamese architecture for tweet-user verification'. *IEEE Trans. Computat Social Syst.* **10** (5), 2764–2772. https://doi.org/10.1109/TCSS.2022.3192909 (2023).
16. Shiha, M. & Ayvaz, S. The effects of Emoji in sentiment analysis. *Int. J. Comput. Electr. Eng. (IJCEE).* **9** (1), 360–369. https://doi.org/10.17706/ijcee.2017.9.1.360-369 (2017).
17. Liu, C. et al. Improving sentiment analysis accuracy with Emoji embedding. *J. Saf. Sci. Resil.* **2** (4), 246–252. https://doi.org/10.1016/j.jnlssr.2021.10.003 (2021).
18. Li, X. et al. A novel deep learning-based sentiment analysis method enhanced with emojis in microblog social networks. *Enterp. Inform. Syst.* **17** (5), 2037160. https://doi.org/10.1080/17517575.2022.2037160 (2023).
19. Ferandez-Gavilanes, M., Costa-Montenegro, E., Garcıa Mendez, S., Gonzalez-Casta˜no, F. & Juncal-Martınez, J. Evaluation of online Emoji description resources for sentiment analysis purposes. *Expert Syst. Appl.* **184**, 115279. https://doi.org/10.1016/j.eswa.2021.115279 (2021).
20. Dang, Y., Zhang, Y. & Chen, H. A lexicon-enhanced method for sentiment classification: an experiment on online product reviews. *IEEE. Intell. Syst.* **25** (4), 46–53. https://doi.org/10.1109/MIS.2009.105 (2009).
21. Surikov, A. & Egorova, E. Alternative method sentiment analysis using emojis and emoticons. *Procedia Comput. Sci.* **178**, 182–193. https://doi.org/10.1016/j.procs.2020.11.020 (2020).
22. Prabhakar, E., Santhosh, M., Krishnan, A., Kumar, T. & Sudhakar, R. Sentiment analysis of US airline Twitter data using new adaboost approach. *Int. J. Eng. Res. Technol. (IJERT).* **7** (1), 1–6 (2019).
23. Patel, A., Oza, P. & Agrawal, S. Sentiment analysis of customer feedback and reviews for airline services using Language representation model. *Procedia Comput. Sci.* **218**, 2459–2467. https://doi.org/10.1016/j.procs.2023.01.221 (2023).
24. Rane, A. & Kumar, A. Sentiment classification system of twitter data for US airline service analysis. In: Proceedings of 2018 IEEE 42nd Annual Computer Software And Applications Conference (COMPSAC). 1, 769–773. IEEE, (2018). https://doi.org/10.1109/COMPSAC.2018.00114
25. Rahat, A., Kahir, A. & Masum, A. Comparison of Naive Bayes and SVM Algorithm based on sentiment analysis using review dataset. In: Proceedings Of 2019 8th International Conference System Modeling And Advancement In Research Trends (SMART). 266–270. IEEE, (2019). https://doi.org/10.1109/SMART46866.2019.9117512
26. Subba, B. & Kumari, S. A heterogeneous stacking ensemble based sentiment analysis framework using multiple word embeddings. *Comput. Intell.* **38** (2), 530–559. https://doi.org/10.1111/coin.12478 (2022).
27. Kaggle (2015). https://www.kaggle.com/crowdflower/twitter-airline-sentiment
28. Hasib, K., Habib, M., Towhid, N. & Showrov, M. A novel deep learning based sentiment analysisof twitter data for us airline service. In: Proceedings of the 2021 International Conference On Information And Communication Technology For Sustainable Development (ICICT4SD). IEEE, Dhaka, Bangladesh, (2021). https://doi.org/10.1109/ICICT4SD50815.2021.9396879
29. Tiwari, P., Yadav, P., Kumar, S., Mishra, B. & Nguyen, G. *& Gochhayat, Et. Al., Sentiment Analysis for Airlines Services Based on Twitter Dataset*149 (Computational Research Methods And Techniques, 2018).
30. Kumawat, S., Yadav, I., Pahal, N. & Goel, D. Sentiment analysis using language models: A study. In: Proceedings of the 2021 11th International Conference On Cloud Computing, Data Science Engineering (Confluence). Amity University, Uttar Pradesh, India, (2021). https://doi.org/10.1109/Confluence51648.2021.9377043
31. Pang, B. & Lee, L. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. *ArXiv Preprint Cs/0506075.* https://doi.org/10.48550/arXiv.cs/0506075 (2005).
32. Felbo, B., Mislove, A., Søgaard, A., Rahwan, I. & Lehmann, S. Using millions of Emoji occurrences to learn anydomain representations for detecting sentiment, emotion and sarcasm. *ArXiv Preprint ArXiv.* 170800524. https://doi.org/10.18653/v1/D17-1169 (2017).
33. Boia, M., Faltings, B., Musat, C. & Pu, P. A :) is worth a thousand words: How people attach sentiment to emoticons and words in tweets. In: Proceedings of 2013 International Conference On Social Computing, 345–350. IEEE, (2013). https://doi.org/10.1109/SocialCom.2013.54
34. Al-Azani, S. & El-Alfy, E. Emoji-based sentiment analysis of Arabic microblogs using machine learning. In: Proceedings of 2018 21st Saudi Computer Society National Computer Conference (NCC), 1–6. IEEE, (2018). https://doi.org/10.1109/NCG.2018.8592970
35. Chawla, N., Bowyer, K., Hall, L. & Kegelmeyer, W. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357. https://doi.org/10.1613/jair.953 (2002).
36. https://www.kaggle.com/datasets/divyansh22/emoji-dictionary-1

Author contributions

A.K. and D.M. conceptualised and designed the study. A.K. conducted the experiments and collected data. B.M analysed the data and contributed to the interpretation of results. A.K.wrote the main manuscript text. D.M. prepared all the figures. All authors reviewed and approved the final manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to A.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.