scientific reports

OPEN

A hybrid transformer and attention based recurrent neural network for robust and interpretable sentiment analysis of tweets

Md Abrar Jahin[1,3], Md Sakib Hossain Shovon[2,3], M. F. Mridha[2,3]✉, Md Rashedul Islam[4,5]✉ & Yutaka Watanobe[6]

Sentiment analysis is a pivotal tool in understanding public opinion, consumer behavior, and social trends, underpinning applications ranging from market research to political analysis. However, existing sentiment analysis models frequently encounter challenges related to linguistic diversity, model generalizability, explainability, and limited availability of labeled datasets. To address these shortcomings, we propose the Transformer and Attention-based Bidirectional LSTM for Sentiment Analysis (TRABSA) model, a novel hybrid sentiment analysis framework that integrates transformer-based architecture, attention mechanism, and recurrent neural networks like BiLSTM. The TRABSA model leverages the powerful RoBERTa-based transformer model for initial feature extraction, capturing complex linguistic nuances from a vast corpus of tweets. This is followed by an attention mechanism that highlights the most informative parts of the text, enhancing the model's focus on critical sentiment-bearing elements. Finally, the BiLSTM networks process these refined features, capturing temporal dependencies and improving the overall sentiment classification into positive, neutral, and negative classes. Leveraging the latest RoBERTa-based transformer model trained on a vast corpus of 124M tweets, our research bridges existing gaps in sentiment analysis benchmarks, ensuring state-of-the-art accuracy and relevance. Furthermore, we contribute to data diversity by augmenting existing datasets with 411,885 tweets from 32 English-speaking countries and 7,500 tweets from various US states. This study also compares six word-embedding techniques, identifying the most robust preprocessing and embedding methodologies crucial for accurate sentiment analysis and model performance. We meticulously label tweets into positive, neutral, and negative classes using three distinct lexicon-based approaches and select the best one, ensuring optimal sentiment analysis outcomes and model efficacy. Here, we demonstrate that the TRABSA model outperforms the current seven traditional machine learning models, four stacking models, and four hybrid deep learning models, yielding notable gain in accuracy (94%) and effectiveness with a macro average precision of 94%, recall of 93%, and F1-score of 94%. Our further evaluation involves two extended and four external datasets, demonstrating the model's consistent superiority, robustness, and generalizability across diverse contexts and datasets. Finally, by conducting a thorough study with SHAP and LIME explainable visualization approaches, we offer insights into the interpretability of the TRABSA model, improving comprehension and confidence in the model's predictions. Our study results make it easier to analyze how citizens respond to resources and events during pandemics since they are integrated into a decision-support system. Applications of this system provide essential assistance for efficient pandemic management, such as resource planning, crowd control, policy formation, vaccination tactics, and quick reaction programs.

Keywords RoBERTa Transformer, Attention-based BiLSTM, Unsupervised Labeling, Tweet Sentiment Analysis, XAI, SHAP, LIME

[1]Department of Industrial Engineering and Management, Khulna University of Engineering & Technology (KUET), Khulna 9203, Bangladesh. [2]Department of Computer Science, American International University-Bangladesh, Dhaka 1229, Bangladesh. [3]Present address: Advanced Machine Intelligence Research (AMIR) Lab, Dhaka 1229, Bangladesh. [4]Offshore AI Development Group, Department of R &D, Chowagiken Corp., Hokkaido, Japan. [5]Department of Computer Science and Engineering, University of Asia Pacific, Dhaka 1216, Bangladesh.

[6]Department of Computer Science and Engineering, University of Aizu, Aizu-Wakamatsu 965-8580, Japan. ✉email: firoz.mridha@aiub.edu; rashed.cse@gmail.com

Due to the growth of textual data on social media platforms, news stories, reviews, and consumer feedback, sentiment analysis (SA), a crucial aspect of natural language processing (NLP), has seen growing attention and usage across several domains[1]. Institutes may get crucial insights into public opinion, consumer preferences, market trends, and brand impression by identifying and analyzing feelings conveyed in text[2]. Thus, SA is critical in directing marketing initiatives, product development, company strategies, and reputation management[3]. Additionally, SA is useful in various domains, including politics, healthcare, economics, and the social sciences, where decision-making and policy development depend on a knowledge of human emotions and attitudes.

Despite its broad use, SA still has issues that need more study and creativity. The lack of generalizability and robustness of SA models is one of the primary issues, especially when applying them to various languages, domains, and datasets[4]. Because existing models frequently display different performance levels based on the properties of the data, they are less dependable in real-world situations where the data distribution may change greatly[5]. Furthermore, SA models' interpretability is still a major worry, particularly in high-stakes scenarios when model predictions are used to make judgments[6]. Deep learning (DL) models are black-box in nature, which makes it difficult to grasp how these models get to their conclusions. This makes it difficult to implement, impedes trust, and holds people accountable for important decision-making processes.

Inspired by these difficulties, this study aims to develop a strong, broadly applicable, and easily interpreted model of SA that will overcome the shortcomings of current techniques. Through improvements in DL, attention mechanisms, and interpretability methodologies, our goal is to develop a model that performs well on various datasets and offers insights into how it makes decisions. This research advances the area of SA by bridging the gap between model performance, interpretability, and practical application. We aim to improve the trustworthiness, transparency, and usefulness of SA models by employing empirical assessments and interpretability studies. This will enable enterprises to make well-informed decisions by relying on dependable sentiment insights.

Our research aims to fill several critical gaps in the existing literature. First, although SA has received a lot of attention-especially regarding social media data-more robust and interpretable models are still required to classify sentiments across various languages and domains accurately. Many current methods are not transparent, scalable, or generalizable, making it difficult to use them in real-world situations. Furthermore, a notable deficiency in current datasets for SA is the absence of representation for various English language usage patterns. Variations in vocabulary, grammar, and contextual usage of English across national boundaries result in subtle discrepancies in the presentation of distinct emotions. This variability presents a problem for SA models, making it difficult to assess sentiments appropriately in various language circumstances. More advanced methods are required to capture minute semantic subtleties and adjust to changing contextual signals since current models may not comprehend sarcasm, context-dependent sentiment changes, or nuanced sentiment expressions.

This study addresses the need for robust and generalizable SA models by proposing the "Transformer and Attention-based Bidirectional LSTM for Sentiment Analysis (TRABSA)" model. The TRABSA model integrates the strengths of transformer-based architecture and attention mechanisms with recurrent neural networks (RNNs) like bidirectional long short-term memory (BiLSTM) to enhance the performance and adaptability of SA tasks. Our method seeks to capture the broad diversity of English language usage and offer a more thorough knowledge of sentiment expression in various linguistic situations by combining data from several locations into a single dataset. By using this method, TRABSA can more effectively adjust to the subtle differences in English language usage across various groups, which improves the precision and significance of SA findings. We test the performance of the TRABSA model on a range of DL architectures and datasets, including extensive Twitter and external social media datasets, with a particular emphasis on scalability, accuracy, and consistency. Additionally, we do interpretability assessments utilizing the SHAP and LIME approaches to understand the model's decision-making mechanism. We show the TRABSA model's generalizability and robustness through our thorough examination, providing a viable method for SA in various real-world situations.

Our research makes eight-fold key contributions:

1. We propose the TRABSA model, a novel hybrid sentiment analysis framework that combines transformer-based architectures, attention mechanisms, and BiLSTM networks to improve sentiment analysis performance.
2. This research leverages the latest RoBERTa-based transformer model, trained on a vast corpus of 124M tweets, to bridge existing gaps in sentiment analysis benchmarks, ensuring state-of-the-art accuracy and relevance.
3. We extended the existing dataset by scraping 411,885 tweets from 32 English-speaking countries to include diversity in the Global Twitter COVID-19 Dataset, acknowledging the varied perspectives and discourse across regions. We scraped an additional 7500 tweets from different states of the USA to deepen geographical representation in the USA Twitter COVID-19 Dataset, allowing for localized insights and analysis.
4. This article thoroughly compares word embedding techniques, establishing the most robust preprocessing and embedding methodologies essential for accurate sentiment analysis and model performance.
5. We methodically label tweets using three distinct lexicon-based approaches and rigorously select the most effective one, ensuring optimal sentiment analysis outcomes and model efficacy.
6. We conduct extensive experiments to assess the TRABSA model's performance on the UK COVID-19 Twitter Dataset, benchmarking against 7 traditional machine learning (ML) models, 4 stacking models, and 4 DL models, demonstrating its superiority and versatility.
7. We evaluate the TRABSA model's robustness and generalizability across 2 extended and 4 external datasets, showcasing its consistent superiority and applicability across diverse contexts and datasets.

8. We provide insights into the interpretability of the TRABSA model through rigorous analysis using SHAP and LIME techniques, enhancing understanding and trust in the model's predictions.The rest of this article is structured as follows: section "Related works" reviews state-of-the-art literature in SA using ML-DL and interpretability techniques. In the section "Methodology," the data collection and preprocessing techniques, unsupervised text labeling, implemented ML and DL models for benchmarking, architecture, and methodology of the proposed TRABSA model are described. The section "Results" presents the experimental setup, evaluation metrics, results, and robustness analysis of the TRABSA model. The SHAP and LIME analysis conducted on the TRABSA model are covered in section "Interpretability analysis". Section "Discussions" addresses the findings and implications of our investigation. In conclusion, the article is summarized, and future research directions are outlined in section "Conclusions and future directions".

Related works

When it comes to classifying data into positive, neutral, and negative sentiment polarity, SA is essential. Exploring a wide range of emotions is the focus of the emerging domains of SA[7]. Sentiments can be further classified into categories like satisfaction and rage within certain settings, such as political disputes[8]. The development of SA approaches with ambivalence management has allowed classifying emotions into distinct classes, including sorrow, anger, anxiety, excitement, and happiness, leading to more nuanced outcomes[9]. While SA has typically focused on textual data, it has expanded to include multimodal SA, which explains data from devices that employ audio- or audio-visual formats[10]. The extension of SA into multimodal analysis highlights its variety and complexity, creating opportunities for a wide range of NLP applications. The variety of options is further highlighted by the fast growth of NLP, fueled by research in neural networks[11]. Notably, the development of Neurosymbolic AI, which combines symbolic reasoning and deep learning, offers a viable method of improving NLP capabilities[12], highlighting the various paths NLP research is taking. Lexicon-based methods, ML-based methods, and hybrid techniques are the three main methodologies for solving text categorization and emotion detection challenges. Word polarity is used by lexicon-based approaches, and ML techniques see text analysis as a classification problem that may be further divided into supervised, semi-supervised, and unsupervised learning approaches[13]. SA results are frequently improved in real-world applications by combining ML with lexicon-based techniques.

In Ahmed & Ahmed's work, positive and negative emotions were used to classify gathered fake newspapers using a variety of approaches, including TF-IDF, random forest (RF), Naïve Bayes (NB), etc.[14]. According to their results, out of all the classifiers used, the Naïve Bayes classifier had the best accuracy (89.30%). To identify feelings in the Twitter sentiment 140 datasets, Gaur et al.[15] used TF-IDF feature extraction and the Naïve Bayes Classifier. The model produced improved accuracy (84.44%) and precision when measured using several performance criteria, such as accuracy, recall, and precision. The COVID-19-related data that Qi & Shabrina[16] examined came from Twitter users in major English cities. They conducted a comparative analysis of ML models, including Vader and Textblob, RF, support vector classification (SVC), and multinominal Naïve Bayes (MNB) models. According to the results of their investigation, SVC with TF-IDF demonstrated better accuracy than the other models. To assess opinions about Saudi cruises, Al Sari et al.[17] created three different datasets from social media platforms. With oversampled Snapchat data, they used ML techniques, including RF, MLP, NB, voting, SVM, and the n-grams feature extraction approach to reach 100% accuracy with the RF algorithm. A customized approach for explicit negation detection was presented by Mukherjee et al.[18]. They used TF-IDF for feature extraction and various ML techniques, including NB, SVM, and Artificial Neural Networks (ANN), to analyze sentiment in Amazon reviews. According to their research, ANNs using negative classifiers had the best accuracy (96.32%). Using reviews from an international hotel, Noori developed a unique algorithm for classifying client sentiment[19]. Following the processing of the reviews, document vectors were created using the TF-IDF extractor and trained using SVM, ANN, NB, k-nearest neighbor (KNN), decision tree (DT), and C4.5 models. Outperforming other models, the DT model scored the highest accuracy (98.9%) with 1800 features. Using N-gram extraction, Zahoor and Rohilla[20] compared NB, SVM, RF, and long short-term memory networks (LSTM) classifiers on preprocessed datasets. In most datasets, including the BJP and ML Khattar datasets, NB showed the best accuracy. To turn COVID-19-related tweets into a text corpus and determine the most common terms using N-grams, Samuel et al. used logistic regression (LR) and NB models[21]. Their results showed that for short tweets, NB and LR had peak accuracy rates of 91% and 74%, respectively. For lengthier tweets, both models performed pretty poorly. Using Maximum Entropy (ME), SVM, and LSTM models, Kumar et al.[22] examined the effects of age and gender on customer reviews. LSTM used word2vec, but the NB, ME, and SVM algorithms used Bag of Words (BOW) feature extraction. For female data, the over-50 age group showed the highest accuracy. SVM and MNB with TF-IDF extraction were used by Zarisfi Kermani et al.[23] on four Twitter datasets, and they proposed semantic scoring techniques to represent features in the vector space. According to their findings, the suggested technique outperformed the MNB algorithm in three datasets, with the STS dataset showing the greatest MNB performance.

Recent advancements in SA and event detection have introduced several innovative models. DocTopic2Vec, proposed by Truică et al.[24], enhances document-level SA by combining local and global contexts through document and topic embeddings, outperforming traditional methods. EDSA-Ensemble[25] improves sentiment classification on social media by integrating event detection with SA using an ensemble approach. Petrescu et al.[26] bridges network and content analysis by combining event detection with SA, achieving high accuracy in sentiment determination. For imbalanced datasets, Truică and Leordeanu[27] compare machine learning algorithms, emphasizing the impact of dataset characteristics on classification performance. Lastly, ATESA-BÆRT by Apostol et al.[28] addresses aspect-based SA using a transformers-based ensemble, outperforming existing models in handling reviews with multiple aspects. Additionally, Mitroi et al.[29] introduces TOPICDOC2VEC, a new topic-document embedding that combines DOC2VEC and TOPIC2VEC, showing superior performance in polarity detection using game reviews.

In 13 languages with different Indic scripts, Bansal et al.[30] looked at the identification of objectionable language. They assessed four sophisticated transformer-based models and contrasted the Transformer-based method with traditional ML models. Out of all of them, XLM-RoBERTa with BiGRU performed better. Furthermore, adding emoji embeddings to XLM-RoBERTa improved the model's efficacy even further. Due to the combined dataset's code-mixing, training using datasets from 13 Indic languages performed better than training with separate models. Gupta et al.[31] presented a unique emotion analysis approach for real-time COVID-19 tweets, examining eight emotions in different domains. The analysis of tweets from India showed changes in emotional reactions, such as less happiness except for nature. Because of their commitment, teachers' faith in education has grown. In terms of precision and recall, the method by Gupta et al.[32] produced aspect-based graphical and textual summaries from mobile reviews, outperforming baseline approaches. Using Twitter data from the Delhi Election 2020, Gupta et al.[33] conducted political echo chamber experiments and investigated the elements that contribute to the creation of echo chambers as well as the role played by users of opposing parties in promoting partisan material. Gupta and colleagues employed ML algorithms and lexicon-based methods to assess sentiment in Hindi tweets. They found that an integrated CNN-RNN-LSTM model produced an accuracy of 85%[34]. Basiri et al.[35] investigated attitudes about the epidemic in eight different nations using DL classification algorithms, and their findings showed distinct sentiment patterns and relationships with pandemic indicators. Using the BERT model, Hayawi et al.[36] achieved excellent accuracy in their ML-based method for spotting COVID-19 vaccination disinformation. Using BERT, Vishwamitra et al.[37] were able to detect hate speech connected to elderly individuals and the Asian community on Twitter during the pandemic. They were able to identify separate word connections for various hate speech datasets. Before and after the initial COVID-19 case announcement, Chen et al.[38] monitored conversations in Luxembourg about policy and daily life; post-announcement, travel-related issues dominated, perhaps because of the region's large immigrant population. To emphasize changing emotions over time, Kabir et al.[39] used ML for word extraction and emotion categorization in COVID-19 tweets. To categorize COVID-19-related Twitter postings, Valdes et al.[40] created a BERT-based model, proving the use of domain-specific data for improved performance. During the pandemic, Tziafas et al.[41] used an ensemble architecture to recognize false information, using transformer-based encoders to achieve high accuracy. Sadia et al.[42] obtained high assessment scores using BERT to conduct SA of COVID-19 tweets. Song et al.[43] analyzed several facets of misinformation diffusion and created a model to categorize misinformation related to COVID-19. Using NLP models, Hossain et al. assessed a dataset for COVID-19-related misinformation detection, offering preliminary benchmarks for advancement[44]. During the COVID-19 lockdown, Chintalapudi et al.[45] used BERT to assess sentiment in Indian tweets, and they showed better accuracy than other models.

The literature review reveals several gaps in SA research, particularly in the context of COVID-19 and social media sentiment classification. While existing studies have explored SA using various ML algorithms and lexicon-based approaches, comprehensive investigations remain lacking across diverse datasets, including those from different geographic regions and languages. Additionally, previous research has focused on individual datasets or specific domains, neglecting SA models' broader applicability and generalizability. Moreover, studies that directly compare different SA techniques, including DL architectures and ensemble methods, are scarce in identifying the most effective approach across various contexts. The need for interpretability and explainability in SA models is also apparent, with few studies incorporating techniques such as SHAP and LIME for insights into model predictions. These gaps highlight the need for more comprehensive and comparative studies encompassing diverse datasets, languages, and evaluation metrics to advance the SA field effectively.

Methodology

Our proposed methodological framework outlines a structured approach to SA, encompassing several key stages to ensure robustness and effectiveness in model development and evaluation, as shown in Fig. 1. The first stage involves gathering relevant data for SA. We extend existing datasets by collecting additional tweets from diverse sources to enhance the dataset's representativeness and coverage. Following data collection, we perform cleaning and preprocessing tasks to ensure the quality and consistency of the data. This includes removing noise, expanding contractions, handling duplicates, emojis, and missing tweets, and standardizing text formats. The next step involves labeling the data into positive, neutral, and negative sentiments. We leverage the latest updated RoBERTa-based pre-trained transformer model for tokenization and sentiment labeling, enabling accurate and efficient text data processing. The dataset is divided into training, validation, and test sets after it has been labeled. This enables us to use the test set to assess the model's performance on untested data, refine hyperparameters using the validation set, and train the model on a portion of the data. In this step, we build the SA model based on our suggested hybrid DL architecture. We used Keras Tuner for hyperparameter optimization within the specified search space, focusing on minimizing the validation loss as our objective to achieve the best performance. Following model development, we compare the trained model's performance against baseline models or current state-of-the-art methods. We examine the model's ability to correctly classify sentiments into positive, neutral, and negative categories using a variety of metrics, including accuracy, precision, recall, and F1-score. Finally, we employ XAI techniques to interpret the model's predictions and gain insights into its decision-making process. This involves analyzing the model's internal mechanisms, such as attention weights or feature importance, to understand the factors influencing its predictions and enhance model interpretability.

Data collection and preprocessing

In this section, we provide a comprehensive overview of the data collection and preprocessing procedures undertaken in our study, which laid the foundation for robust SA of tweets.

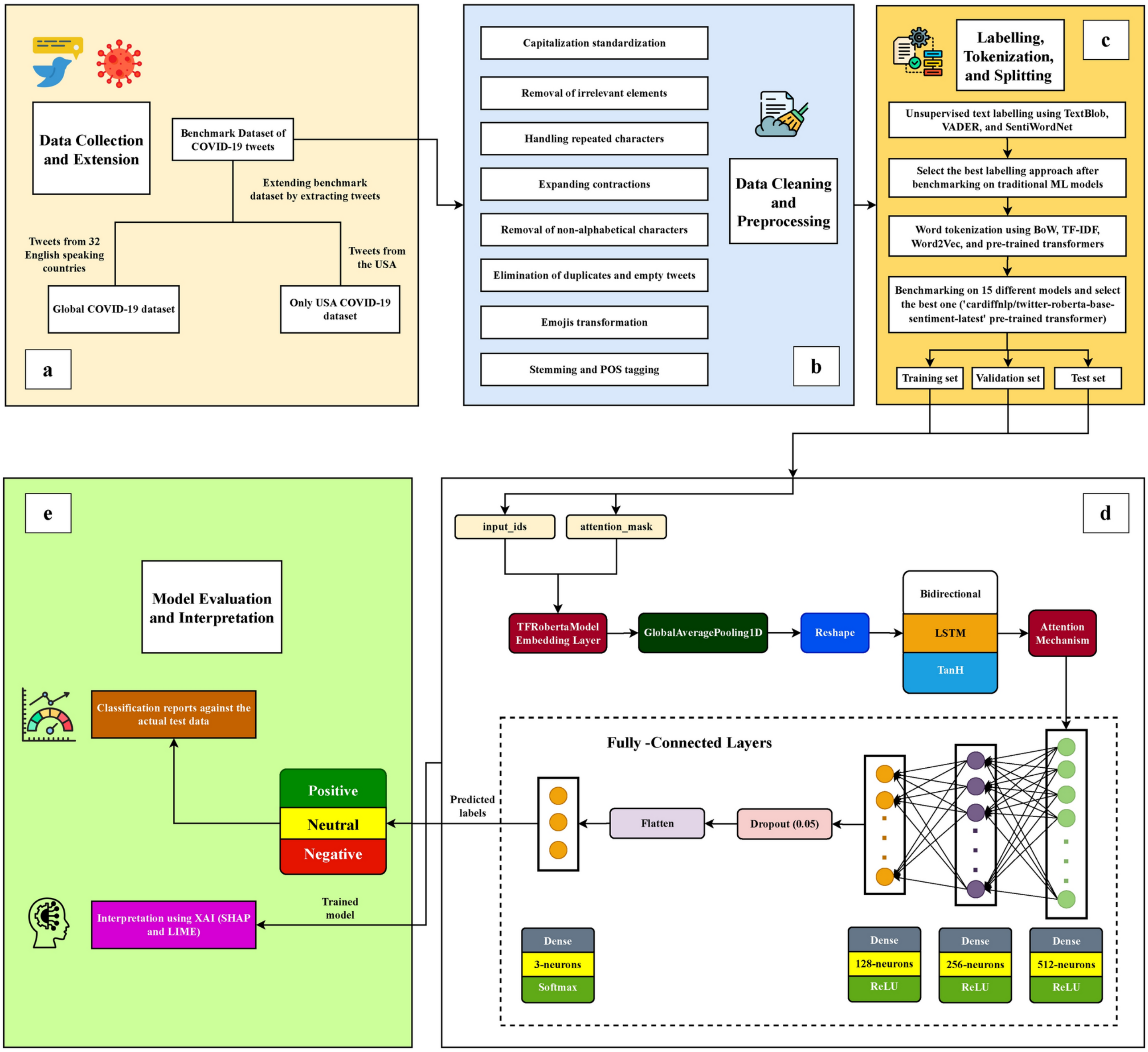

a
Data Collection and Extension
Benchmark Dataset of COVID-19 tweets
Extending benchmark dataset by extracting tweets
Tweets from 32 English speaking countries
Tweets from the USA
Global COVID-19 dataset
Only USA COVID-19 dataset
b
Capitalization standardization
Removal of irrelevant elements
Handling repeated characters
Expanding contractions
Removal of non-alphabetical characters
Elimination of duplicates and empty tweets
Emojis transformation
Stemming and POS tagging
Data Cleaning and Preprocessing
c
Labelling, Tokenization, and Splitting
Unsupervised text labelling using TextBlob, VADER, and SentiWordNet
Select the best labelling approach after benchmarking on traditional ML models
Word tokenization using BoW, TF-IDF, Word2Vec, and pre-trained transformers
Benchmarking on 15 different models and select the best one ('cardiffnlp/twitter-roberta-base-sentiment-latest' pre-trained transformer)
Training set
Validation set
Test set
d
input_ids
attention_mask
TFRobertaModel Embedding Layer
GlobalAveragePooling1D
Reshape
Bidirectional
LSTM
TanH
Attention Mechanism
Fully -Connected Layers
Flatten
Dropout (0.05)
Dense
3-neurons
Softmax
Dense
128-neurons
ReLU
Dense
256-neurons
ReLU
Dense
512-neurons
ReLU
Predicted labels
e
Model Evaluation and Interpretation
Classification reports against the actual test data
Positive
Neutral
Negative
Trained model
Interpretation using XAI (SHAP and LIME)

Fig. 1. This figure depicts the step-by-step methodological framework proposed for tweet sentiment analysis. It begins with (**a**) data collection and extension, followed by (**b**) data cleaning and preprocessing. Subsequently, (**c**) sentiment labeling into positive, neutral, and negative categories is performed using the 'cardiffnlp/twitter-roberta-base-sentiment-latest' pre-trained transformer, and the dataset is split into training, validation, and test sets. The framework proceeds with (**d**) model development, (**e**) model benchmarking, and evaluation against baseline models or state-of-the-art approaches. Finally, the process concludes with XAI interpretation techniques applied to gain insights into the model's predictions.

Data sources

This study employed a comprehensive set of seven distinct datasets to facilitate a thorough exploration of SA across various dimensions. These datasets were classified into three main categories: Benchmark, Extended, and External, each serving a unique purpose in our research.

Benchmark dataset: The benchmark dataset, which forms the basis of our research, was first assembled and curated by[16]. It functions as a standard by which our proposed model's performance is measured. Interestingly, our model outperformed this benchmark dataset, indicating significant progress in SA. This reference dataset consists of tweets with geotags from well-known cities in the United Kingdom during the third nationwide COVID-19 shutdown. Figure 2 shows the tweets gathered from the three stages in the UK. This group of cities includes Greater London, Bristol, South Hampton, Birmingham, Manchester, Liverpool, Newcastle, Leeds, Sheffield, and Nottingham. Over the course of three weeks, from January 6, 2021, to July 18, 2021, 77,332 tweets were gathered. 29,923 tweets were gathered in the first stage, 24,689 in the second, and 22,720 in the third. Major cities such as London, Manchester, Birmingham, and Liverpool were the source of most tweets, with London having the highest count with 37,678. Smaller cities, like Newcastle, had just 852 tweets in a six-month period.

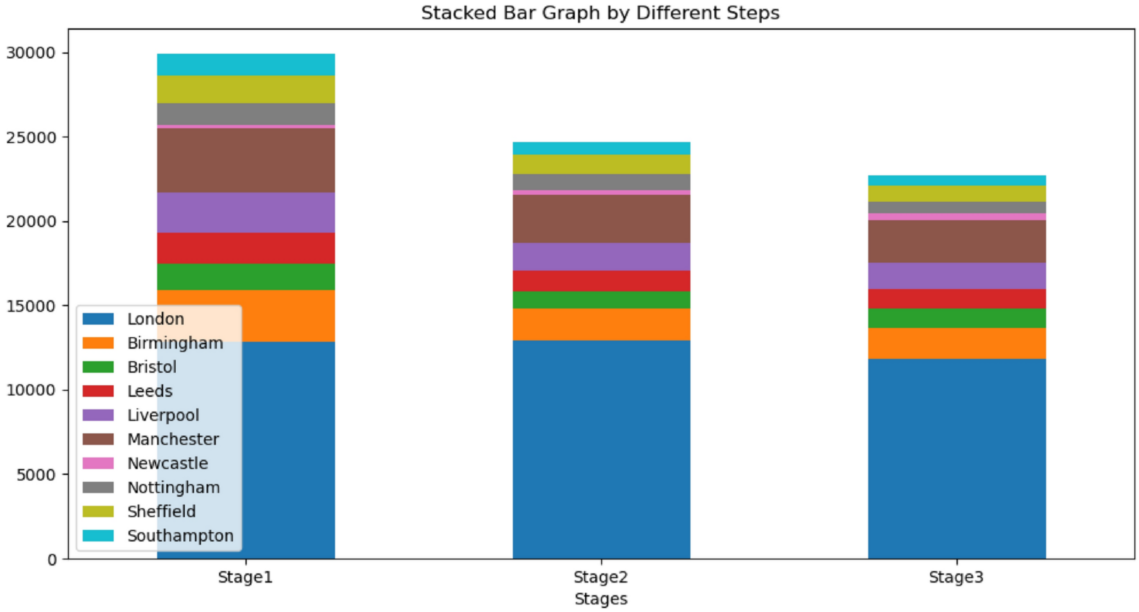


Fig. 2. Stacked bar chart showing tweet distribution in three stages of data collection during third lockdown period from the major cities of the UK.

The data distribution is in phases, with the first stage having the greatest data and the third stage having the least, as Fig. 2 illustrates. While Newcastle's contribution was connected with its population and density, London consistently supplied the most data.

Extended datasets: To augment our research's cross-cultural dimension and overcome the geographic limitations of the benchmark dataset proposed by Qi and Shabrina, we extended the existing UK COVID-19 Twitter dataset[16]. The tweets were sourced through a combination of data extraction tools, specifically Twint and the Twitter Academic API. These tools were chosen because of their ability to acquire tweets with geolocation information, which is crucial for conducting geographical analyses. However, it should be noted that only a few 1% of Twitter users actively opt to share their geographic location when composing tweets, and this feature is not enabled by default[46]. The extended datasets, comprising the Global Twitter COVID-19 Dataset and the USA Twitter COVID-19 Dataset, are publicly available in the Extended Covid Twitter Datasets repository[47].

To ensure a comprehensive dataset, we merged the data collected by Twint and the Twitter Academic API. This amalgamation allowed us to access a larger volume of tweets. In identifying tweets related to the COVID-19 pandemic, we employed specific keywords such as "corona" or "covid" in the Twint search configurations and the query field of the Twitter Academic API. This search strategy enabled us to extract tweets and associated hashtags containing these pertinent terms.

1. Extended Global COVID-19 Dataset: This extension involved the comprehensive scraping of 411,885 tweets from 32 English-speaking countries. This dataset expansion allowed us to capture sentiment variations across diverse English-speaking regions, as illustrated in Fig. 3. In particular, cities such as the "United States," the "United Kingdom," "Australia," and "New Zealand" exhibit high tweet volumes, while several other cities have comparatively lower tweet counts. Figure 4 illustrates word clouds and word frequencies within tweets of the extended datasets. Figure 4 (left) represents a visual summary of the most frequently occurring words in a vast dataset related to the COVID-19 pandemic. At the center of this cloud is the word "covid," which dominates with a staggering 226,463 mentions. Other significant terms around it, such as "vaccine," "case," "test," and "people," indicate the key topics and concerns worldwide during the pandemic. Words like "death," "pandemic," and "health" also hold prominence, highlighting the gravity of public health issues. Additionally, terms like "Trump" and "government" suggest the political dimensions entwined with the pandemic discourse.
2. Extended USA COVID-19 Dataset: In addition to the international extension, we further enriched our data by creating an extended dataset focusing exclusively on the United States. This dataset comprised 7500 tweets meticulously scraped from U.S.-based sources. Including this dataset allows for a closer examination of sentiment dynamics within a specific geographical context. Figure 4 (right) specific to the United States reveals notable trends and sentiments within the country during the COVID-19 pandemic. In this dataset, words like "corona," "coronavirus," and "covid" are prominent, underlining the ubiquitous presence of these terms in American discussions. Interestingly, words like "fool" and "joke" appear, possibly reflecting a spectrum of attitudes towards the pandemic response. Negative expressions like "shit," "fuck," and "die" also emerge, suggesting the emotional intensity and frustration associated with the situation. Terms like "test" and "case" point towards testing and infection rates concerns, while "april" indicates a temporal reference.

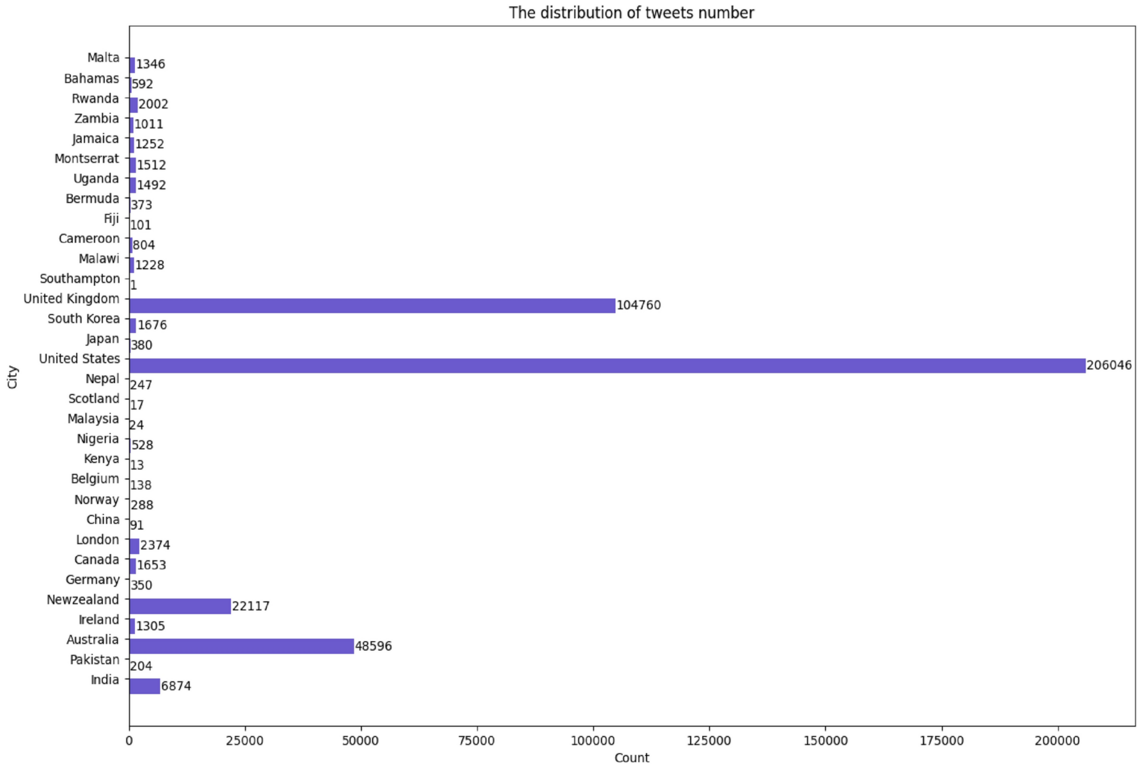


Fig. 3. Bar plot showing the distribution of tweets across 32 English-speaking countries.

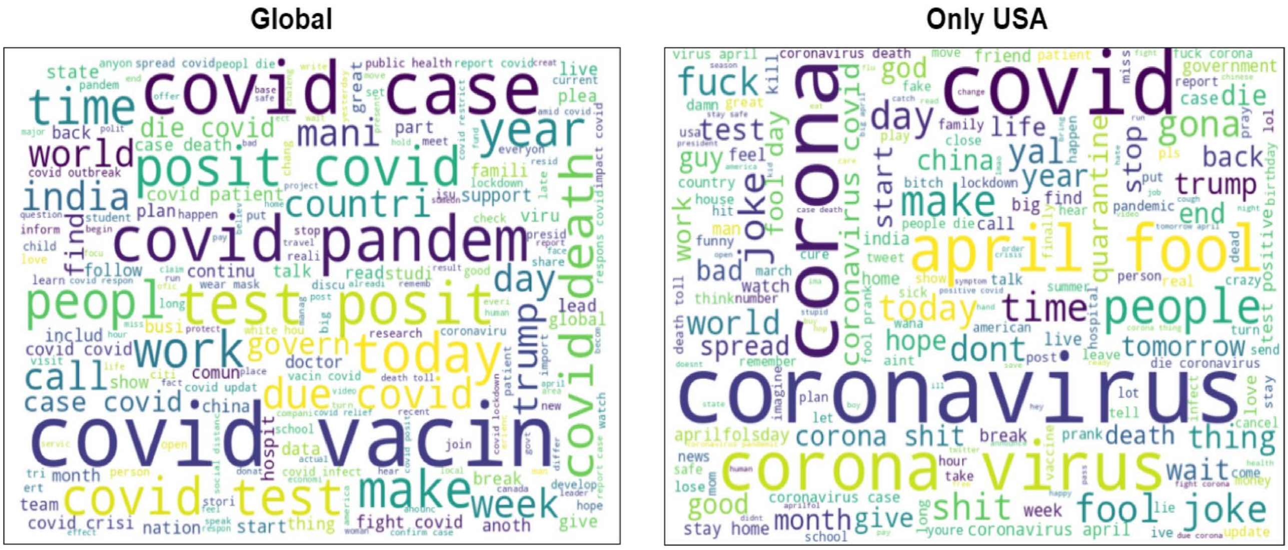


Fig. 4. This word cloud visualizes the most frequently occurring words in the "Global" (left) and "Only USA" (right) datasets of COVID-19-related tweets. The size of each word corresponds to its frequency in the dataset.

External datasets: In order to evaluate the robustness and generalizability of our proposed model, we incorporated four external datasets sourced from Kaggle, including the Twitter and Reddit Dataset, Apple Dataset, and US Airline Dataset (see Data availability section).

1. Twitter Dataset: This dataset, collected from Twitter, represents diverse tweets covering various topics and subject matter. It enables us to assess the model's adaptability to various Twitter content.
2. Reddit Dataset: The Reddit dataset encompasses user-generated content from the popular social media platform. Its inclusion allows us to explore sentiment patterns in a different online community, offering valuable insights into the model's versatility.

3. Apple Dataset: The Apple dataset consists of textual data related to the technology giant Apple Inc. By incorporating this dataset, we aim to analyze sentiment in a specific industry context, providing a more nuanced view of the model's performance.
4. US Airline Dataset: This dataset is centered around discussions related to U.S. airlines. It allows us to investigate sentiment trends within the context of the aviation industry, adding yet another layer of applicability to our model.

Data cleaning and preprocessing

As a previous study has indicated, preprocessing the raw Twitter data was essential to guarantee the accuracy and reliability of our SA because of their informal and unstructured character[48]. Our thorough data-cleaning procedure included the following crucial steps:

1. Capitalization Standardization: To prevent the recognition of identical words with varying capitalization as distinct, we uniformly converted all text to lowercase. This step was crucial for consistent word recognition.
2. Removal of Irrelevant Elements: We methodically eliminated any superfluous content that has no bearing on SA, including hashtags (#subject), stated usernames (@username), and any hyperlinks beginning with "www," "http," or "https." We also removed terms that were less than two characters and stop words. Stop words, though common in text, often lack significant sentiment polarity. Despite being often used in texts, stop words frequently lack strong emotive polarity. It's important to note that negations like "not" and "no" were kept in because removing them may change the sense of whole sentences.
3. Handling Repeated Characters: Some users utilize repeating characters in their tweets to highlight intense feelings. Words not present in standard lexicons were transformed into their correct forms to standardize such expressions. For instance, "sooooo goooood" was normalized to "so good."
4. Extending Contractions: Removing punctuation after a contraction, such "isn't" or "don't," presented difficulties. They were expanded into their full forms to maintain the meaningfulness of contractions. For instance, "isn't" became "is not."
5. Elimination of Non-Alphabetical Characters: All punctuation, numerals, and special symbols were removed, along with all other non-alphabetical characters and symbols. These extraneous characters had the potential to interfere with feature extraction.
6. Elimination of Duplicates and Empty Tweets: We identified and removed duplicated or empty tweets to ensure data integrity, creating a clean and consistent dataset.
7. Emojis Transformation: Given the prevalence of emojis in tweets to express sentiment and emotion, we adopted the 'demojize()' function from Python's emoji module to transform emojis into their corresponding textual meanings. This enhancement was especially beneficial for improving the accuracy of SA.
8. Advanced Cleaning for Specific Approaches: Depending on the SA approach employed, additional cleaning steps, such as stemming and Part-of-Speech (POS) tagging, were applied. These steps were particularly relevant for methods relying on resources like SentiWordNet.By rigorously implementing these data-cleaning procedures, we ensured that our SA was conducted on a high-quality dataset, minimizing noise and optimizing the extraction of meaningful sentiment features.

Word embeddings

We used various word embedding approaches in this work to extract contextual and semantic information from our textual material. Our NLP tasks performed much better thanks to these embeddings. The word embedding techniques we used in our studies are summarized in the next subsections.

Bag-of-Words (BoW): BoW is a classic technique for word representation. It transforms tweets into vectors by counting the frequency of words in each tweet. While it doesn't capture word order or context, it provides a straightforward and interpretable way to represent text data. To utilize the BoW approach, we employed the 'CountVectorizer' function from the scikit-learn library.

Term Frequency-Inverse Document Frequency (TF-IDF): We utilized the TF-IDF embeddings by employing the 'TfidfVectorizer' function from the scikit-learn module, which allocates weights to words according to their significance in individual tweets and their scarcity throughout the complete dataset. Additionally, it made it easier to down-weight frequent keywords, which allowed our models to concentrate on more informative words.

Word2Vec: One of Word2Vec's advantages is that it can record semantic similarities between words, which makes text data analysis more sophisticated. Using neural networks, it represents words as dense vectors in a continuous vector space. The 'word_tokenize' function from the NLTK library was utilized to tokenize our tweets, as it allows for the breakdown of sentences into individual words. Using the Gensim package, a Word2Vec model was produced with the vector size, window size, and skip-gram model set. By using a continuous vector space, this approach was able to express words as vectors. Two methods were used to encode full tweets as vectors: sum vectorization and average vectorization.

Pre-trained transformers: In our research, we harnessed the power of pre-trained transformer-based models from the Hugging Face Transformers library to leverage contextual embeddings for text data. Three distinct transformer models were employed, each bringing unique capabilities to the analysis. The 'distilbert-base-uncased' model, known for its efficiency and lightweight nature, was selected for its suitability in scenarios where computational resources are constrained. It produces context-aware word embeddings that consider each word's left and right context. We used a state-of-the-art 'cardiffnlp/twitter-roberta-base-sentiment-latest' model, updated in 2022, to capture sentiment-specific nuances in a tweet. This model was trained on an extensive dataset of approximately 124 million tweets collected from January 2018 to December 2021. This model was designed for English text and was a robust foundation for our SA endeavors. Additionally, we incorporated 'sentence-transformers/all-MiniLM-L6-v2,' a sophisticated tool that transforms tweets into a dense vector space of 384 dimensions. It transforms entire sentences into fixed-dimensional vectors while maintaining semantic information.

Although the code implementation for each transformer followed a similar structure, the choice of model brought diversity to our experimentation, enabling us to explore the impact of contextual embeddings on our text classification task. To tokenize and process our text data effectively, we employed the model's associated tokenizer, incorporating techniques such as padding and truncation to ensure consistent input lengths. The tokenized data was then efficiently processed on the GPU for optimal computational performance. We further harnessed the model's capabilities to extract the hidden states associated with the '[CLS]' token, which often encapsulates the comprehensive context of the text.

Unsupervised text labeling

In ML, labeling vast amounts of text data manually can be time-consuming. To address this challenge and expedite the labeling process, we leveraged lexicon-based methods, specifically TextBlob, VADER, and SentiWordNet, to automatically assign sentiment scores to tweets. Our sentiment classification scheme employed three categories: positive (assigned a value of 1), negative (assigned − 1), and neutral (assigned 0).

We used the BoW method implemented with the CountVectorizer from the scikit-learn library to convert text data from our benchmark dataset into a matrix of word frequencies. We conducted a comprehensive evaluation to determine the effectiveness of our unsupervised labeling approach. The performance of seven traditional base ML models was evaluated against sentiment scores derived from each of the three lexicon approaches: TextBlob, VADER, and SentiWordNet. Our evaluation unveiled that TextBlob consistently outperformed VADER and SentiWordNet regarding accuracy across all implemented ML models, as shown in Table 1. Hence, we used TextBlob-based labels for further benchmarking.

Models used

Traditional ML models

Our analysis encompasses a diverse set of models traditional ML models, including traditional base models, their stacked ensembles, and voting classifiers. The aim was to comprehensively evaluate the performance of these models using different text representations. The following models were used:

1. Random Forest (RF): RF is an ensemble learning method that aggregates the predictions of multiple decision trees. It is known for its robustness and ability to handle high-dimensional data.
2. Naive Bayes (NB): NB is a probabilistic classifier based on Bayes' theorem, assuming independence among features. It is particularly well-suited for text classification tasks.
3. Support Vector Machine (SVM): SVM is a powerful classifier that aims to find a hyperplane that best separates data points in a high-dimensional space. It is effective for both linear and non-linear classification.
4. Gradient Boosting Machine (GBM): GBM is an ensemble learning technique that builds decision trees sequentially, focusing on the mistakes of the previous trees. It often leads to strong predictive performance.
5. LightGBM (LGBM): LightGBM is a gradient-boosting framework for efficiency and speed. It uses a histogram-based approach for tree construction.
6. XGBoost: XGBoost is another popular gradient-boosting library known for its scalability and performance optimization. It has been widely used in various ML competitions.
7. CatBoost: CatBoost is a gradient-boosting library specializing in categorical feature support. It is known for its ability to tackle categorical data effectively.
8. LGBM + K-Nearest Neighbors (KNN) + Multi-Layer Perceptron (MLP): We explored an ensemble approach by combining LGBM with KNN and MLP to leverage the strengths of different algorithms.
9. RF + KNN + MLP: Similarly to the previous ensemble, we combined RF with KNN and MLP to diversify our modeling approach further.
10. GBM + RF Stacking Classifier: Stacking is an ensemble technique where multiple models' predictions are combined using another model. Here, we stack GBM and RF to improve predictive accuracy potentially.
11. GBM + RF Voting Classifier: Voting classifiers combine the predictions of multiple models by majority voting. We used this ensemble technique to take advantage of the collective wisdom of GBM and RF.We explored various combinations of word embeddings and text representations for each model, including BoW, TF-IDF, Word2Vec, and pre-trained transformer models for text tokenization. These different rep-

ML model	Accuracy of Lexicon based methods		
	TextBlob	**VADER**	**SentiWordNet**
RF	67%	65%	57%
NB	66%	59%	61%
SVM	73%	66%	72%
GBM	79%	71%	69%
LGBM	79%	73%	70%
XGBoost	76%	70%	71%
Catboost	74%	69%	68%

Table 1. Performance evaluation of unsupervised sentiment labeling approaches.

resentations allowed us to assess the impact of text preprocessing on model performance and gain insights into which models were most effective for sentiment classification.

Deep neural networks (DNNs)

To rigorously evaluate tweet sentiment classification, we employed a diverse set of DNN models, each with distinct architectural characteristics. We utilized the Keras Tuner for hyperparameter tuning across these models to ensure optimal performance. The search space included LSTM layer units ranging from 128 to 768, dense layer units from 64 to 512, dropout rates from 0.1 to 0.5, and learning rates from 1×10^{-5} to 1×10^{-2}. Using Keras Tuner's Random Search method, we identified the best parameters for each model, significantly enhancing their performance. This systematic exploration, coupled with comprehensive text representations and optimized hyperparameters, provided valuable insights into the performance of various DNN architectures.

Our initial model, the 'Single-Dense Layered Neural Network,' started with a transformer for feature extraction. A single dense layer with 512 units and ReLU activation captured high-level representations. The simplicity of this architecture allowed us to establish a baseline for performance comparison.

Building upon this foundation, we introduced the '3 dense layers of neural network. After global averaging of the transformer's outputs, three sequential dense layers were introduced, with decreasing units (512, 256, 128) to refine feature representations progressively. In particular, dropout regularization was applied after the first dense layer, enhancing model robustness.

We introduced the 'BiLSTM + 3 Hidden Dense Layers' model to explore the nuances of SA texts further. This architecture incorporated a BiLSTM layer, which is a special type of RNN, enabling the network to capture sequential dependencies in the input data. Following the BiLSTM layer, three additional dense layers (512, 256, and 128 units) were used to distill the features further. Dropout was applied to enhance model generalization.

Lastly, we explored hybrid architecture with the 'BiLSTM + CNN' model. Here, the model combined the strengths of a BiLSTM layer with convolutional layers. The convolutional layers, with 64 filters and varying kernel sizes, added a spatial perspective to feature extraction. Subsequently, two dense layers (128 and 64 units) were introduced to further process the extracted features.

Proposed TRABSA model

The proposed TRABSA model presents a systematic and effective architecture for SA, as shown in Fig. 5. The model utilizes the 'cardiffnlp/twitter-roberta-base-sentiment-latest' pre-trained transformer, capitalizing on its contextual understanding of the text. The architecture begins with input layers, including 'input_ids' and 'attention_mask,' where a maximum sequence length of 256 tokens is utilized. RoBERTa is used for its excellent performance in tasks involving natural language understanding. It encodes the input tweet text and generates contextual embeddings.

The input data is then organized into a TensorFlow dataset, shuffled, and batched for training. The batch size is set to 16 to facilitate efficient training. The labels are one-hot encoded to prepare them for multiclass classification. The dataset is split into training and validation sets for rigorous evaluation. In optimizing the TRABSA model, Keras Tuner plays a crucial role by systematically exploring a well-defined search space to fine-tune various hyperparameters. The search space includes the number of units in the BiLSTM layer, ranging from 128 to 512, which balances model complexity and efficiency. Additionally, three Dense layers are tuned

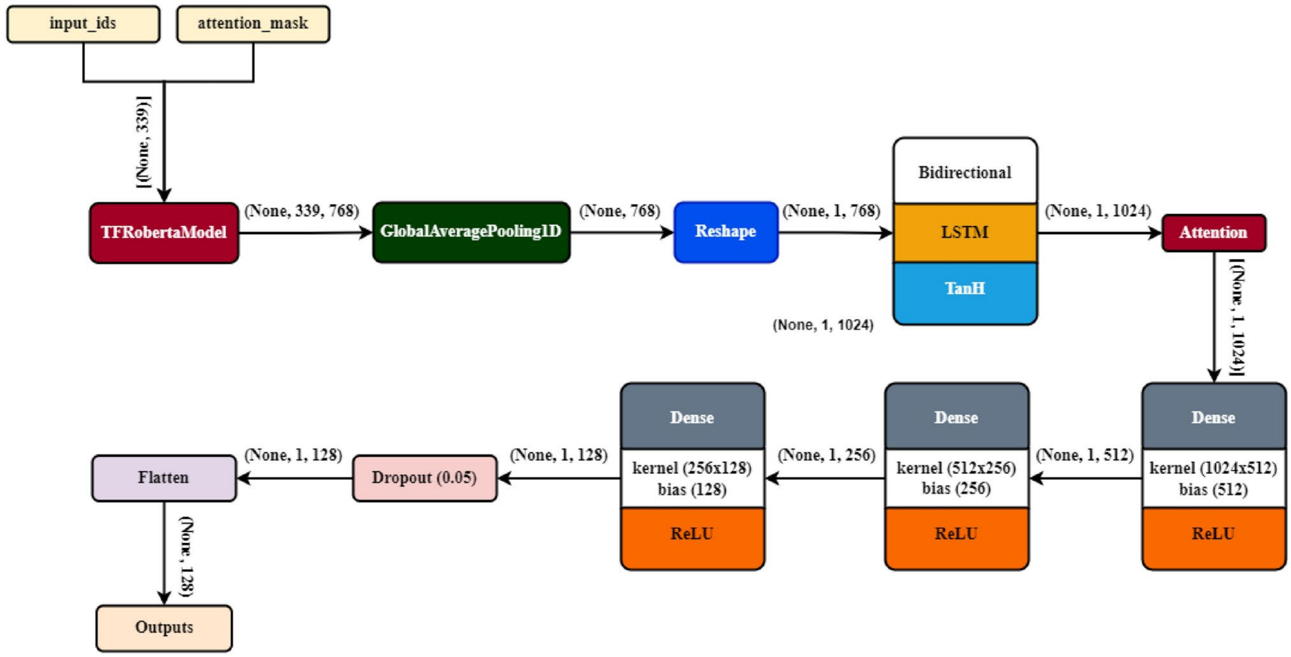


Fig. 5. Model architecture of the proposed TRABSA model.

with units ranging from 128 to 768, 64 to 512, and 32 to 256, respectively, affecting the model's capacity and computational demands. The dropout rate varies from 0.1 to 0.5 to prevent overfitting, while the learning rate is explored logarithmically between 1e−5 and 1e−3 to optimize convergence speed. The tuning process employs Random Search to sample various hyperparameter combinations, providing an efficient way to explore the space without exhaustive search. After running multiple trials, the tuner identifies the best hyperparameter settings (see Fig. 5) based on validation loss, ensuring an optimized balance between performance and computational efficiency.

The core architecture of the TRABSA model is based on the pre-trained RoBERTa-base model architecture with specific enhancements:

- *Input Layers:* Two input layers are defined—'input_ids' and 'attention_mask,' which receive the tokenized input tweet sequences and their corresponding attention masks, ensuring proper handling of padded fixed-length sequences.
- *Transformer embeddings:* The transformer produces contextual embeddings, which capture rich information about the text. These embeddings are then subjected to 'Global Average Pooling' to reduce the dimensionality while retaining essential features. A reshaping operation is applied to prepare the data for subsequent layers.
- *BiLSTM:* The BiLSTM layer is an advanced type of RNN designed to enhance sequence modeling by capturing contextual information from both directions in a sequence. In this model, the BiLSTM layer is configured with 512 units in each direction, totaling 512×2 units. The forward LSTM network processes the sequence from start to end, while the backward LSTM network processes it from end to start. This bidirectional approach allows the BiLSTM layer to integrate information from both past and future tokens, providing a more nuanced understanding of the text.
- *Self-attention mechanism:* An attention layer is incorporated, which applies self-attention to the output of the BiLSTM. This mechanism allows the model to weigh the importance of different parts of the input sequence, which can be critical for understanding the nuances of sentiment in tweets.
- *Dense layers:* A series of densely connected layers are added to capture complex patterns and relationships within the data. While there are multiple dense layers, their architecture plays a crucial role. A 512-unit dense layer with ReLU activation serves as the primary feature extractor, followed by two more dense layers (256 and 128 units) to refine representations progressively. A dropout layer with a 0.05 dropout rate contributes to regularization and helps prevent overfitting.
- *Flatten layer:* After processing through the dense layers, the output tensor is flattened to a 1D vector.
- *Classifier head:* The classifier head consists of a dense layer with three units, using the softmax activation function. It produces the input tweet's final sentiment classification probabilities (positive, negative, or neutral).The model is compiled using the Adam optimizer with a learning rate of 4×10^{-5} and categorical cross-entropy loss. Categorical accuracy is used as the evaluation metric. We included model checkpointing, early stopping, and callbacks to optimize model training. The early stopping mechanism monitors validation loss and restores the best weights to prevent overfitting. The training process involves fitting the model on the training dataset and validating it on the validation dataset for 50 epochs; however, due to the early stopping mechanism, the iterations stop after 23 epochs.

The learning rate scheduler callback function adjusted the learning rate during our model training. The function calculates the learning rate for each epoch based on an initial learning rate and an exponential decay factor, which controls the rate at which the learning rate decreases over epochs. In this specific implementation, the exponential decay formula is utilized, where the learning rate is multiplied by the exponent of a negative constant $k = 0.1$ multiplied by the epoch number. As the epoch increases, the learning rate exponentially decreases, allowing for a gradual reduction in the learning rate during training. This technique helps optimize the training process by fine-tuning the learning rate to improve model convergence and performance over successive epochs, as shown in Fig. 6.

Figure 7 comprehensively evaluates the TRABSA model's performance in tweet SA. The top left corner showcases the model's classification metrics plot, illustrating precision, recall, and F1-score metrics for each sentiment class. Moving clockwise, the confusion matrix provides a detailed analysis of the model's classification performance by comparing predicted sentiment labels with actual labels. The loss vs. epoch curve illustrates its training and validation loss over successive epochs, while the training and validation accuracy vs. epoch curve depicts the model's learning progress and convergence. Together, these visualizations offer insights into the TRABSA model's classification accuracy, convergence, and optimization process, aiding researchers in assessing its performance and identifying areas for improvement.

The TRABSA model's architecture combines the strengths of RoBERTa's contextual embeddings, BiLSTM's sequence modeling, attention mechanisms for capturing interdependencies, and a well-designed set of dense layers to improve SA accuracy for tweet data. This architecture demonstrated superior performance compared to other models, making it a noteworthy addition to the field of SA.

Results

Experimental setup

Table 2 overviews the hardware and software specifications used in our ML and DL experiments. It includes details on the CPU, GPU, TPU, RAM, Python version, and essential libraries utilized to conduct the research.

Evaluation metrics

We use a variety of assessment indicators in our research to determine our models' overall success. These measurements show how well the models correctly categorize sentiments across various classifications. The

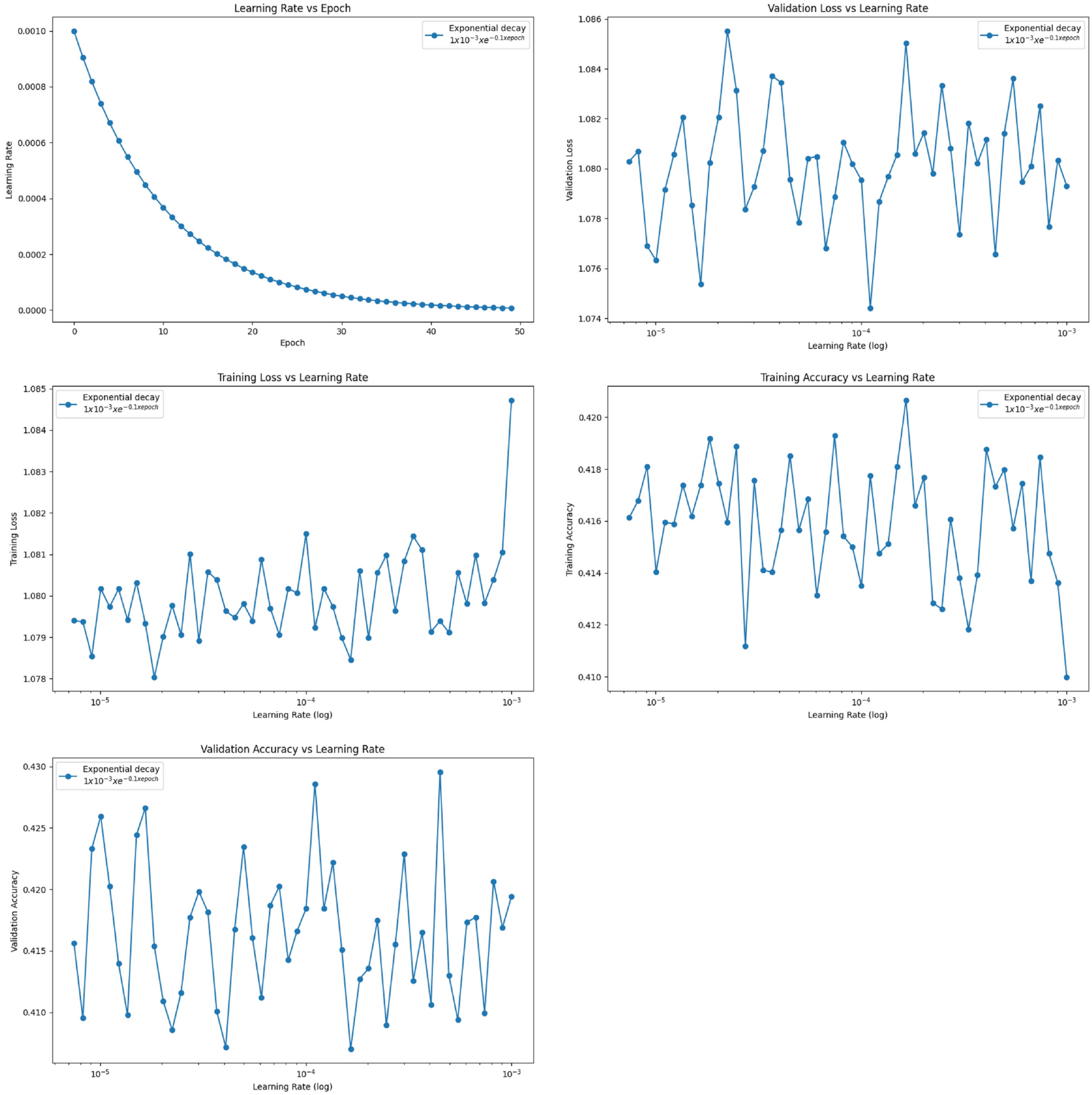


Fig. 6. This figure depicts the learning rate scheduler callback function utilized during model training. It visualizes the decayed learning rate as the epoch increases, alongside the corresponding training and validation accuracy, as well as training and validation loss, plotted against variable learning rates.

F1-score, accuracy, precision, recall, and macro-average metrics are among the crucial assessment measures employed. These are computed using the following definitions of true positives (TP), false positives (FP), and false negatives (FN):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

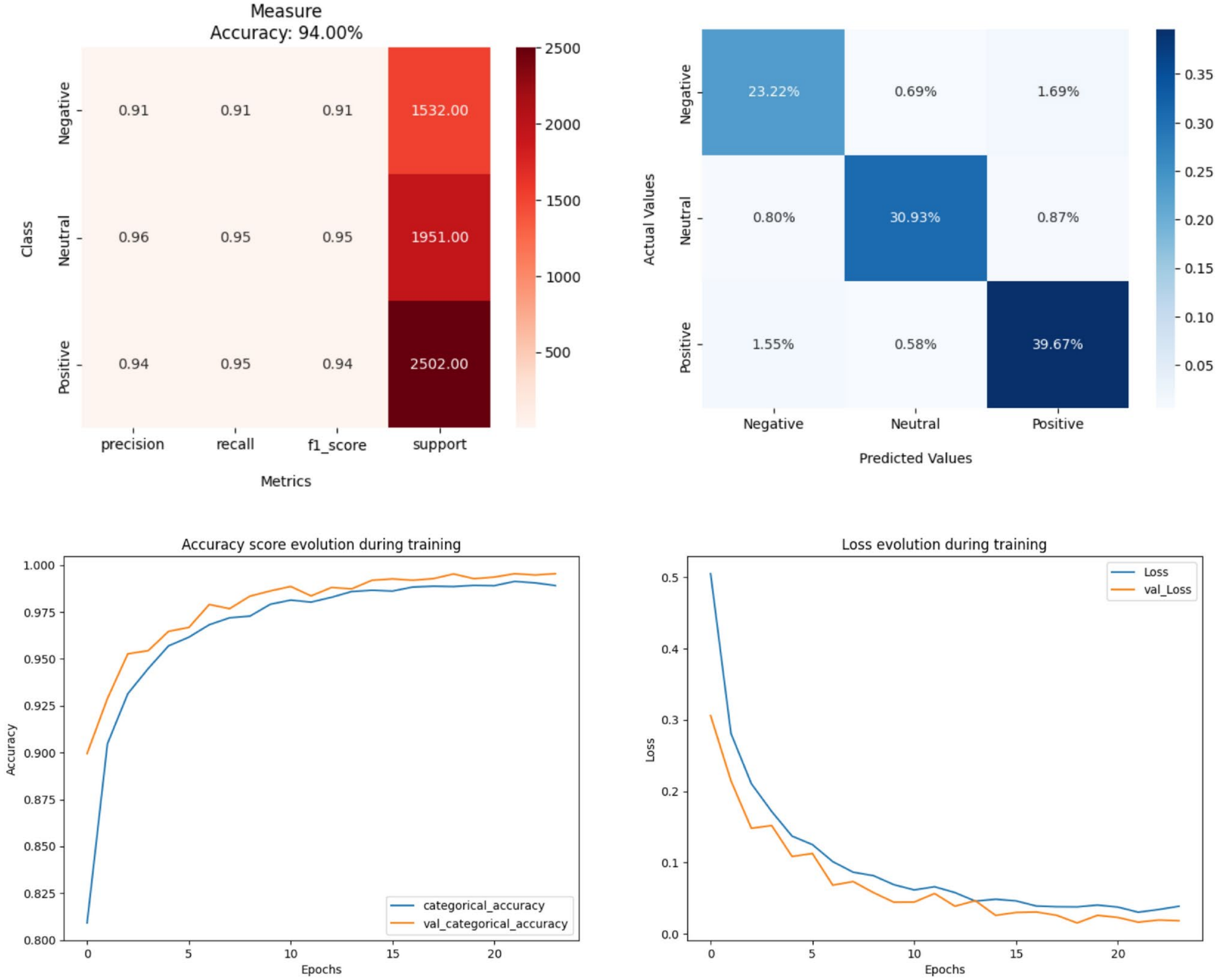


Fig. 7. Masterplot illustrating the TRABSA model's performance starting from the top right and proceeding clockwise including the (**a**) classification metrics plot, (**b**) confusion matrix, (**c**) loss vs Epoch curve, (**d**) training and validation accuracy vs Epoch curve.

Resources used	Specifications
Intel(R) Xeon(R) CPU	x86 with a clock frequency of 2 GHz, 4 vCPU cores, 18GB
NVIDIA T4 x2 GPU	2560 Cuda cores, 16 GB
Google TPU	8 TPU v3 cores. 128 GB
RAM	16 GB DDR4
Python	Version 3.10.12
Libraries	Numpy, pandas, matplotlib, seaborn, nltk, TensorFlow, keras, PyTorch, genism, scikit-learn, joblib, transformers, re, string, shap, scipy, lime

Table 2. Details of hardware and software specifications.

Recall, also known as sensitivity, quantifies the percentage of properly identified positive occurrences among all real positive instances, whereas precision describes the proportion of correctly classified positive instances among all cases projected as positive. The F1-score, which is derived from the harmonic mean of accuracy and recall, offers a fair indicator of model performance.

The accuracy metric quantifies the percentage of properly identified occurrences out of all instances, as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Where TN represents true negatives.

Furthermore, we calculate the F1-score, recall, and macro-average accuracy to present a comprehensive evaluation of the model's performance across all sentiment classes:

$$\text{Macro Average Precision} = \frac{1}{N}\sum_{i=1}^{N}\text{Precision}_i \tag{5}$$

$$\text{Macro Average Recall} = \frac{1}{N}\sum_{i=1}^{N}\text{Recall}_i \tag{6}$$

$$\text{Macro Average F1-score} = \frac{1}{N}\sum_{i=1}^{N}\text{F1-score}_i \tag{7}$$

Where N represents the total number of sentiment classes.

Together, these evaluation measures offer a thorough analysis of the model's performance employed in this research, assisting in the choice and improvement of SA models.

Results

Analyzing the performance of various models across different word embedding techniques reveals significant variations in their effectiveness for SA tasks. Among the traditional models, GBM and LightGBM consistently demonstrate strong performance across all word embedding techniques, achieving macro average F1-scores ranging from 79% to 83%. Pre-trained transformer models like BERT and RoBERTa exhibit competitive performance, with the RoBERTa model showcasing high accuracy across various configurations, especially when combined with advanced neural network architectures. Notably, the TRABSA model outperforms all other models across all evaluation metrics, demonstrating exceptional macro average precision (94%), macro average recall (93%), macro average F1-score (94%), and accuracy of 94% (see Table 3). This significant improvement underscores the efficacy of the TRABSA framework in SA tasks, surpassing even state-of-the-art transformer models like RoBERTa.

We used BoW, TFIDF, word2vec, BERT, SBERT, and RoBERTa as word embeddings with several ML and two DL models. When we noticed a significant improvement in the accuracy (above 80%) while using RoBERTa word embeddings with simple DL architectures, we decided to implement more complex hybrid DL models: BiLSTM+3 Hidden Layers NN, BiLSTM+CNN, and TRABSA.

Compared to the best-performing traditional models like GBM (81%), LightGBM (83%), stacked LGBM+KNN+MLP (84%), and advanced hybrid state-of-the-art BiLSTM+3 Hidden Layers NN (85%), the TRABSA model achieves an impressive accuracy of 94%, indicating a substantial improvement of at least 9% over the closest competitors. The TRABSA model consistently outperforms others in accuracy and macro average precision, recall, and F1-score, demonstrating its robustness and effectiveness across different evaluation criteria. Even when compared to sophisticated neural network architectures like Single Hidden Layer NN (84%) and 3 Hidden Layers NN (84%), the TRABSA model exhibits a remarkable performance boost of $\approx 10\%$, further emphasizing its superiority in SA tasks. The TRABSA model's outstanding performance underscores its hybrid architecture's efficacy, which integrates transformer-based mechanisms, attention mechanisms, and BiLSTM networks to capture nuanced sentiment patterns effectively. The findings suggest that while pre-trained transformer models like BERT and RoBERTa offer competitive performance, customized architectures like TRABSA tailored specifically for SA tasks can yield substantial accuracy and predictive power improvements.

The ablation study presented in Table 4 offers a detailed analysis of the proposed TRABSA model by examining the effects of removing or altering key components. The full model, not shown in this table, achieves the highest performance, with macro average precision and F1-score reaching 94%, macro average recall at 93%, and accuracy at 94%. This highlights the effectiveness of the BiLSTM, Attention, and Dense layers in capturing complex patterns from the data.

When the BiLSTM, Attention, and Dense layers are all removed, the model's performance drastically drops to 76% for precision, 75% for recall, 75% for F1-score, and 76% for accuracy. The reduction in training time ($\approx$ 1290 s) and inference time ($\approx$116 s) indicates the computational savings from eliminating these layers, but the performance decline reveals their critical role in generalizing well to the data.

In the model where only the Attention layer is removed, the performance sees a significant improvement compared to the full removal of BiLSTM and Dense layers. This configuration achieves an 85% macro average precision, 84% recall, and 84% F1-score, with an accuracy of 84%. The training time is $\approx$2250 s, and the inference time is $\approx$146 s, suggesting that while the Attention layer adds value, its absence doesn't drastically impair the model's ability to capture temporal dependencies, likely due to the strength of the BiLSTM.

Interestingly, replacing the BiLSTM with a unidirectional LSTM results in a slight drop in performance across all metrics, with precision at 83%, recall at 82%, F1-score at 82%, and accuracy at 83%. As expected, the training and inference times ($\approx$2381 s and $\approx$142 s, respectively) show that unidirectional LSTMs are slightly more efficient but at the cost of losing the bidirectional context offered by the BiLSTM.

Moreover, the model without the Dropout layer exhibits similar performance to the attention-removed configuration, maintaining 84% across all metrics, but with a slightly faster training time ($\approx$2201 s) and inference

Word embedding	Model	Macro average precision	Macro average recall	Macro average F1-score	Accuracy	Training time (s)	Inference time (s)
BoW	RF	75% ± 1%	61% ± 2%	61% ± 2%	67% ± 1%	6.73	0.03
BoW	NB	63% ± 1%	62% ± 2%	62% ± 2%	64% ± 1%	0.82	0.01
BoW	SVM	76% ± 2%	76% ± 2%	76% ± 2%	77% ± 1%	4.35	0.30
BoW	GBM	82% ± 1%	79% ± 1%	79% ± 1%	81% ± 1%	262.81	0.02
BoW	LGBM	83% ± 1%	81% ± 1%	81% ± 1%	83% ± 1%	35.54	0.01
BoW	XGBoost	77% ± 0%	75% ± 1%	75% ± 1%	78% ± 0%	86.50	0.02
BoW	Catboost	80% ± 0%	74% ± 1%	75% ± 1%	77% ± 1%	536.16	0.02
BoW	LGBM+KNN+MLP	82% ± 1%	81% ± 2%	81% ± 2%	84% ± 0%	1422.53	45.43
BoW	RF+KNN+MLP	74% ± 1%	73% ± 2%	74% ± 1%	75% ± 1%	1212.43	40.54
BoW	GBM+RF Stacking Classifier	78% ± 0%	76% ± 1%	76% ± 1%	78% ± 0%	1364.06	50.57
BoW	GBM+RF Voting Classifier	78% ± 0%	69% ± 1%	70% ± 1%	72% ± 1%	1250.37	48.30
BoW	Single Hidden Layer NN	71% ± 4%	69% ± 4%	69% ± 4%	72% ± 4%	646.34	24.53
BoW	3 Hidden Layers NN	68% ± 4%	67% ± 4%	67% ± 4%	70% ± 4%	904.64	25.35
TFIDF	RF	69% ± 1%	60% ± 3%	60% ± 3%	66% ± 2%	6.75	0.04
TFIDF	NB	65% ± 1%	56% ± 3%	56% ± 3%	61% ± 2%	0.89	0.02
TFIDF	SVM	74% ± 2%	70% ± 3%	70% ± 3%	73% ± 2%	5.15	0.46
TFIDF	GBM	80% ± 1%	77% ± 3%	77% ± 3%	79% ± 1%	273.51	0.05
TFIDF	LGBM	80% ± 0%	77% ± 3%	78% ± 3%	80% ± 1%	36.12	0.02
TFIDF	XGBoost	68% ± 1%	67% ± 1%	67% ± 1%	70% ± 1%	87.22	0.03
TFIDF	Catboost	74% ± 1%	72% ± 1%	72% ± 1%	74% ± 1%	542.44	0.03
TFIDF	LGBM+KNN+MLP	79% ± 0%	77% ± 1%	77% ± 1%	79% ± 0%	1532.37	46.24
TFIDF	RF Bagging	76% ± 0%	48% ± 1%	43% ± 1%	56% ± 1%	764.34	35.34
TFIDF	RF+KNN+MLP	75% ± 1%	71% ± 3%	72% ± 2%	74% ± 0%	1254.65	42.75
TFIDF	GBM+RF Stacking Classifier	77% ± 0%	76% ± 1%	76% ± 1%	78% ± 0%	1352.53	52.65
TFIDF	GBM+RF Voting Classifier	75% ± 0%	69% ± 2%	70% ± 1%	73% ± 0%	1283.23	50.75
TFIDF	Single Hidden Layer NN	68% ± 3%	67% ± 4%	67% ± 4%	69% ± 2%	650.34	22.43
TFIDF	3 Hidden Layers NN	93% ± 3%	92% ± 4%	92% ± 3%	93% ± 4%	954.64	27.53
word2vec	RF	51% ± 1%	49% ± 2%	49% ± 2%	56% ± 1%	6.82	0.05
word2vec	NB	41% ± 1%	34% ± 1%	19% ± 1%	36% ± 1%	1.03	0.04
word2vec	SVM	67% ± 0%	49% ± 2%	45% ± 3%	57% ± 1%	5.78	0.57
word2vec	GBM	51% ± 1%	50% ± 2%	50% ± 2%	56% ± 1%	282.10	0.08
word2vec	LGBM	53% ± 1%	49% ± 1%	47% ± 1%	57% ± 1%	37.41	0.03
word2vec	XGBoost	55% ± 1%	50% ± 2%	48% ± 2%	56% ± 1%	88.60	0.03
word2vec	Catboost	71% ± 0%	49% ± 1%	45% ± 1%	57% ± 1%	553.37	0.04
word2vec	LGBM+KNN+MLP	53% ± 1%	46% ± 2%	41% ± 3%	53% ± 1%	1448.76	47.34
word2vec	RF+KNN+MLP	55% ± 1%	49% ± 3%	43% ± 3%	57% ± 1%	1345.53	55.23
word2vec	GBM+RF Stacking Classifier	46% ± 2%	45% ± 2%	45% ± 2%	51% ± 1%	1412.64	53.73
word2vec	GBM+RF Voting Classifier	47% ± 1%	47% ± 1%	47% ± 1%	53% ± 1%	1350.23	28.78
word2vec	Single Hidden Layer NN	36% ± 5%	45% ± 4%	40% ± 4%	53% ± 3%	704.65	28.34
word2vec	3 Hidden Layers NN	38% ± 4%	48% ± 4%	42% ± 4%	56% ± 3%	1034.89	33.38
BERT	RF	53% ± 2%	51% ± 3%	49% ± 3%	58% ± 1%	805.43	45.53
BERT	NB	50% ± 3%	51% ± 2%	50% ± 2%	53% ± 1%	1.12	0.07
BERT	SVM	52% ± 2%	52% ± 2%	52% ± 2%	56% ± 1%	6.12	0.76
BERT	GBM	56% ± 1%	54% ± 2%	54% ± 2%	60% ± 2%	291.11	0.11
BERT	LGBM	58% ± 2%	57% ± 2%	57% ± 2%	61% ± 2%	37.66	0.03
BERT	XGBoost	56% ± 2%	56% ± 2%	56% ± 2%	60% ± 2%	89.10	0.04
BERT	Catboost	56% ± 2%	49% ± 2%	46% ± 2%	57% ± 2%	562.29	0.05
BERT	LGBM+KNN+MLP	61% ± 1%	59% ± 2%	59% ± 2%	62% ± 1%	1623.65	65.34
BERT	RF+KNN+MLP	60% ± 1%	58% ± 2%	57% ± 2%	62% ± 1%	1443.22	60.44
BERT	GBM+RF Stacking Classifier	55% ± 2%	54% ± 2%	54% ± 2%	56% ± 1%	1523.75	70.23
BERT	GBM+RF Voting Classifier	60% ± 1%	58% ± 2%	58% ± 2%	66% ± 0%	1452.45	67.85
BERT	Single Hidden Layer NN	61% ± 3%	58% ± 4%	59% ± 4%	62% ± 2%	945.53	35.64
BERT	3 Hidden Layers NN	62% ± 4%	60% ± 4%	60% ± 4%	64% ± 4%	1305.39	45.49
SBERT	RF	61% ± 0%	48% ± 3%	44% ± 4%	54% ± 2%	6.95	0.06
Continued							

Word embedding	Model	Macro average precision	Macro average recall	Macro average F1-score	Accuracy	Training time (s)	Inference time (s)
SBERT	NB	53% ± 2%	40% ± 3%	32% ± 4%	44% ± 2%	1.14	0.09
SBERT	SVM	56% ± 1%	57% ± 2%	56% ± 2%	58% ± 1%	6.28	0.81
SBERT	GBM	55% ± 2%	51% ± 2%	49% ± 2%	55% ± 2%	302.43	0.15
SBERT	LGBM	55% ± 2%	52% ± 2%	50% ± 2%	56% ± 2%	38.18	0.04
SBERT	XGBoost	56% ± 2%	56% ± 2%	56% ± 2%	57% ± 2%	90.53	0.06
SBERT	Catboost	69% ± 0%	48% ± 3%	42% ± 4%	53% ± 2%	577.54	0.05
SBERT	LGBM+KNN+MLP	55% ± 2%	49% ± 2%	47% ± 2%	53% ± 2%	1734.23	68.64
SBERT	RF+KNN+MLP	56% ± 1%	54% ± 2%	54% ± 2%	57% ± 1%	1522.42	63.43
SBERT	GBM+RF Stacking Classifier	44% ± 2%	44% ± 2%	43% ± 2%	48% ± 1%	1623.86	72.57
SBERT	GBM+RF Voting Classifier	53% ± 2%	50% ± 2%	50% ± 2%	54% ± 2%	1553.54	74.67
SBERT	Single Hidden Layer NN	55% ± 3%	55% ± 4%	54% ± 3%	59% ± 3%	954.64	38.48
SBERT	3 Hidden Layers NN	56% ± 4%	57% ± 3%	57% ± 3%	59% ± 2%	1402.54	48.48
RoBERTa	RF	62% ± 2%	62% ± 2%	62% ± 2%	64% ± 1%	7.23	0.08
RoBERTa	NB	63% ± 1%	55% ± 2%	52% ± 3%	54% ± 2%	1.16	1.05
RoBERTa	SVM	65% ± 1%	65% ± 1%	65% ± 1%	67% ± 0%	6.76	0.88
RoBERTa	GBM	63% ± 2%	62% ± 3%	63% ± 2%	65% ± 2%	314.44	0.17
RoBERTa	LGBM	64% ± 2%	63% ± 3%	64% ± 2%	66% ± 1%	38.89	0.05
RoBERTa	XGBoost	65% ± 1%	63% ± 2%	63% ± 2%	66% ± 1%	91.14	0.07
RoBERTa	Catboost	63% ± 2%	62% ± 3%	61% ± 4%	64% ± 2%	583.28	0.07
RoBERTa	LGBM+KNN+MLP	66% ± 2%	66% ± 2%	65% ± 3%	66% ± 2%	1823.93	72.23
RoBERTa	RF+KNN+MLP	58% ± 3%	55% ± 2%	55% ± 2%	60% ± 1%	1654.54	67.76
RoBERTa	GBM+RF Stacking Classifier	61% ± 3%	61% ± 3%	61% ± 3%	63% ± 2%	1705.36	75.96
RoBERTa	GBM+RF Voting Classifier	60% ± 2%	60% ± 2%	59% ± 3%	60% ± 2%	1653.78	73.47
RoBERTa	Single Hidden Layer NN	84% ± 4%	84% ± 4%	84% ± 4%	84% ± 4%	1349.46	154.39
RoBERTa	3 Hidden Layers NN	84% ± 3%	83% ± 4%	83% ± 4%	84% ± 3%	1898.05	153.64
RoBERTa	BiLSTM+3 Hidden Layers NN	84% ± 3%	84% ± 3%	84% ± 3%	85% ± 2%	3404.54	148.43
RoBERTa	BilSTM+CNN	83% ± 2%	81% ± 4%	82% ± 3%	83% ± 2%	5328.73	178.64
RoBERTa	Proposed TRABSA model	94% ± 1%	93% ± 2%	94% ± 1%	94% ± 1%	3675.21	147.14

Table 3. Comprehensive 10-fold cross-validated mean performance evaluation metrics with standard deviations for various models and embeddings, including time performance (training and inference times).

Word embedding	Model	Macro average precision	Macro average recall	Macro average F1-score	Accuracy	Training time (s)	Inference time (s)
RoBERTa	w/o[1] BiLSTM + Attention + 3 Dense Layers	76% ± 2%	75% ± 3%	75% ± 2%	76% ± 3%	1290.53	115.68
RoBERTa	w/o BiLSTM + Attention Layer	81% ± 1%	80% ± 1%	80% ± 1%	80% ± 2%	3500.23	169.34
RoBERTa	w/o Attention Layer	85% ± 2%	84% ± 3%	84% ± 2%	84% ± 2%	2250.15	145.85
RoBERTa	w/o 3 Dense Layers	82% ± 2%	81% ± 3%	81% ± 3%	82% ± 3%	2245.64	150.14
RoBERTa	w/o Dropout Layer	84% ± 1%	84% ± 1%	84% ± 1%	84% ± 2%	2200.70	140.45
RoBERTa	LSTM instead of BiLSTM Layer	83% ± 2%	82% ± 3%	82% ± 2%	83% ± 3%	2380.90	142.30

Table 4. Ablation test of the proposed TRABSA model. "w/o" stands for "without," indicating the absence of the specific component in the TRABSA model configuration

time (≈140 s). This suggests that Dropout contributes to regularization, but its absence does not significantly degrade performance, possibly due to the robustness of the RoBERTa embeddings and other layers.

The study underscores the importance of BiLSTM and Attention layers for optimal performance while also demonstrating the computational costs associated with these enhancements. The model without these components, while more computationally efficient, sacrifices accuracy, confirming the balance between complexity and performance in the proposed TRABSA model.

Robustness test of TRABSA model

The TRABSA model consistently demonstrates robustness and generalizability across datasets and DL architectures, as evidenced by its consistent performance metrics. Across various datasets, including the Global COVID-19 Dataset, the USA COVID-19 Dataset, the External Twitter Dataset, the Reddit Dataset, the Apple

Dataset, and the US Airline Dataset, the TRABSA model consistently achieves high macro average precision, recall, F1-score, and accuracy values. For instance, in the Global COVID-19 Dataset, the TRABSA model attains an impressive 98% macro average precision, recall, F1-score, and accuracy. Similarly, the USA COVID-19 dataset maintains high scores, obtaining 87% in terms of accuracy. The trend continues across External datasets, with the TRABSA model consistently performing exceptionally well, achieving an average accuracy of 97% on the Twitter Dataset, 95% on the Reddit Dataset, 90% on the Apple Dataset, and 96% on the US Airline Dataset (see Table 5). These consistent and high-performance metrics underscore the reliability and effectiveness of the TRABSA model across diverse datasets and DL architectures, reaffirming its robustness and generalizability in SA tasks.

The robustness and generalizability of our TRABSA model are evident through its superior performance compared to a wide range of state-of-the-art models used in multiclass sentiment analysis (SA) on Twitter data. Table 6 summarizes these comparisons, showcasing how TRABSA consistently outperforms models across multiple datasets. Notably, on the global COVID-19 dataset, TRABSA achieves an exceptional macro average precision, recall, F1-score, and accuracy of 98%, significantly surpassing models such as Jlifi et al.[58], which utilized the Ens-RF-BERT approach and achieved a macro average F1-score of 94.03% and accuracy of 93.01%. Similarly, the model by Sazan et al.[57], which employed RoBERTa+fastText, attained an F1-score of 92.05% but still falls short when compared to TRABSA's 95% F1-score on the US Airline dataset. Furthermore, the CNN-LSTM model proposed by Mohbey et al.[56] achieved 91.24% F1-score, showcasing a respectable result, but it is outperformed by TRABSA's 98% on the global COVID-19 dataset.

What sets TRABSA apart is its consistent performance across different datasets, including both domain-specific (e.g., the US Airline dataset, where it achieved 96% accuracy) and global datasets. In contrast, other models often exhibit variability in performance depending on the dataset or sentiment categories. This ability to generalize across diverse contexts, such as pandemic-related tweets and US airline sentiment, highlights TRABSA's robustness in handling complex multiclass SA tasks. The models compared in this table span various

			Evaluation metrics					
Dataset type	Dataset name	DL models	Macro average precision	Macro average Recall	Macro average F1-score	Accuracy	Training time (s)	Inference time (s)
Extended	Global COVID-19 Dataset	Single Hidden Layer NN	97% ± 0%	97% ± 0%	97% ± 0%	97% ± 1%	7365	491
		3 Hidden Layers NN	97% ± 0%	97% ± 0%	97% ± 0%	97% ± 1%	7755	517
		BiLSTM+3 Hidden Layers NN	97% ± 1%	97% ± 1%	97% ± 1%	97% ± 1%	5709	518
		BiLSTM+CNN	11% ± 5%	33% ± 4%	17% ± 5%	33% ± 4%	8789	536
		Proposed TRABSA Model	98% ± 0%	98% ± 0%	98% ± 0%	98% ± 1%	8288	518
Extended	USA COVID-19 Dataset	Single Hidden Layer NN	81% ± 3%	81% ± 3%	81% ± 3%	83% ± 3%	870	58
		3 Hidden Layers NN	85% ± 3%	83% ± 1%	84% ± 2%	85% ± 3%	696	58
		BiLSTM+3 Hidden Layers NN	85% ± 1%	85% ± 1%	85% ± 1%	86% ± 0%	1218	59
		BiLSTM+CNN	17% ± 5%	33% ± 5%	22% ± 4%	51% ± 1%	413	59
		Proposed TRABSA Model	87% ± 1%	86% ± 1%	86% ± 1%	87% ± 1%	1081	47
External	Twitter dataset	Single Hidden Layer NN	93% ± 3%	93% ± 3%	93% ± 3%	93% ± 3%	9891	495
		3 Hidden Layers NN	92% ± 1%	92% ± 1%	92% ± 1%	92% ± 1%	4608	288
		BiLSTM+3 Hidden Layers NN	92% ± 3%	92% ± 3%	92% ± 3%	92% ± 3%	8700	291
		BiLSTM+CNN	53% ± 2%	49% ± 4%	46% ± 3%	49% ± 4%	1752	292
		Proposed TRABSA model	97% ± 1%	97% ± 1%	97% ± 1%	97% ± 1%	6602	287
External	Reddit dataset	Single Hidden Layer NN	94% ± 3%	93% ± 3%	94% ± 3%	94% ± 3%	1494	90
		3 Hidden Layers NN	94% ± 1%	94% ± 1%	94% ± 1%	94% ± 2%	2415	119
		BiLSTM+3 Hidden Layers NN	94% ± 1%	94% ± 2%	94% ± 2%	94% ± 1%	2464	101
		BiLSTM+CNN	94% ± 1%	94% ± 0%	94% ± 0%	94% ± 0%	1944	119
		Proposed TRABSA Model	94% ± 1%	93% ± 0%	94% ± 0%	95% ± 1%	2200	94
External	Apple dataset	Single Hidden Layer NN	81% ± 1%	82% ± 2%	81% ± 1%	84% ± 3%	96	11
		3 Hidden Layers NN	83% ± 2%	81% ± 3%	82% ± 3%	85% ± 1%	55	10
		BiLSTM+3 Hidden Layers NN	87% ± 2%	85% ± 4%	86% ± 3%	89% ± 0%	140	12
		BiLSTM+CNN	85% ± 1%	83% ± 3%	84% ± 2%	87% ± 0%	130	11
		Proposed TRABSA Model	88% ± 1%	86% ± 2%	86% ± 2%	90% ± 0%	210	12
External	US Airline dataset	Single Hidden Layer NN	93% ± 3%	93% ± 3%	93% ± 3%	94% ± 2%	1201	48
		3 Hidden Layers NN	94% ± 2%	93% ± 3%	93% ± 3%	94% ± 2%	1166	53
		BiLSTM+3 Hidden Layers NN	94% ± 1%	93% ± 2%	94% ± 1%	94% ± 1%	1012	46
		BiLSTM+CNN	94% ± 3%	93% ± 4%	93% ± 4%	94% ± 3%	842	40
		Proposed TRABSA Model	95% ± 0%	95% ± 0%	95% ± 0%	96% ± 1%	897	39

Table 5. Generalizability and robustness of the proposed TRABSA model on both the extended and external datasets.

Study	Model used	Dataset	Macro average precision	Macro average recall	Macro average F1-score	Accuracy
Qi & Shabrina (2023)[16]	BoW+SVC	UK COVID-19 Twitter Dataset	69.66%	70.33%	69.66%	71.00%
Ours	TRABSA	UK COVID-19 Twitter Dataset	94.00%	93.00%	94.00%	94.00%
dos Santos Neto et al., (2024)[49]	BERT	TripAdvisor	87.70%	88.20%	87.90%	88.20%
Brum & Volpe Nunes (2018)[50]	BERT	TweetSentBR	73.27%	72.75%	72.96%	72.75%
De Souza et al. (2018)[51]	MultiFiT-Twitter LM	Twitter NPS	72.43%	72.46%	72.43%	72.46%
Pilar et al. (2023)[52]	Neighbor-sentiment	InterTASS	57.76%	51.39%	54.39%	61.35%
Su & Kabala (2023)[53]	GloVe100+LSTM	500k ChatGPT-related Tweets Jan-Mar 2023	81.10%	81.10%	81.10%	81.10%
Memiş et al. (2024)[54]	Multiclass CNN model with pre-trained word embedding	Turkish Financial Tweets	–	–	–	72.73%
Kp et al. (2024)[55]	Ensemble classifier	Twitter API Dataset	91.29%	89.65%	87.32%	93.42%
Mohbey et al. (2024)[56]	CNN-LSTM	Monkeypox Tweets	91.24%	91.24%	91.24%	91.24%
Sazan et al., 2024)[57]	RoBERTa+fastText	US Airline Dataset	92.08%	92.02%	92.05%	92.02%
Ours	TRABSA	US Airline Dataset	95.00%	95.00%	95.00%	96.00%
Jlifi et al. (2024)[58]	Ens-RF-BERT	Hashtag Covid19 Tweets	94.03%	93.05%	94.03%	93.01%
Bhardwaj et al. (2024)[59]	BoW+LR	COV19Tweets Dataset	82.00%	81.80%	81.60%	81.80%
Ours	TRABSA	Global COVID-19 dataset	98.00%	98.00%	98.00%	98.00%

Table 6. Summary of the proposed models in the state-of-the-art tweet sentiment analysis literature.

techniques, from traditional models such as BoW+SVC[16] to more modern DL architectures like CNN-LSTM[56] and transformer-based approaches such as BERT[49], yet none achieve the same level of performance as TRABSA.

Statistical validation

To assess the performance of our proposed TRABSA model against the other top-performing models benchmarked in this study, we performed a 10-fold cross-validated two-tailed paired t-test, each with 50 epochs, on key evaluation metrics: accuracy, macro average precision, recall, and F1-score. By top-performing model, we refer to the best combination of word embedding and the model obtained from Table 3. Our null hypothesis (H_0) stated that there is no significant difference between the performance of each model and the TRABSA model for the respective metrics. In contrast, the alternative hypothesis (H_1) posited that a significant difference does exist. We utilized a significance level of $\alpha = 5\%$, with a Bonferroni correction to account for multiple comparisons. The t-test results (see Table 7) revealed that for all metrics-accuracy, precision, recall, and F1-score-the TRABSA model showed statistically significant improvements compared to the other models ($p - values < 0.05$ after adjustment) by rejecting the H_0. For example, the accuracy of the TRABSA model was significantly higher than that of BoW+RF, with a t-value of 108.7332 and a p-value of 2.39×10^{-15}, suggesting a meaningful enhancement in performance. Similarly, the TRABSA model consistently demonstrated superior results with significant t-statistics and p-values for precision and recall. These findings robustly support the efficacy of the TRABSA model in delivering enhanced performance metrics compared to traditional models.

Interpretability analysis

This section discusses the interpretability analysis of the TRABSA model, employing SHAP and LIME techniques to enhance explainability.

SHAP

A useful tool for deciphering and understanding the results of ML models is the SHapley Additive exPlanations (SHAP) framework[60]. The computation and presentation of the relevance assigned to each characteristic in the prediction process are made easier by utilizing the SHAP Python package. Calculating SHAP values, which measure feature contribution and improve the interpretability of ML models, is essential to the SHAP framework. When the features (x) are unknown, a SHAP value specifies how to go from the expected or base value $E[f(x)]$ to the actual output f. Furthermore, by clarifying the direction of the link between features and the target variable, SHAP values shed light on how characteristics affect predictions. A characteristic with a SHAP value of 1 or -1, for example, significantly influences the prediction for a given data point favorably or negatively. On the other hand, a feature that approaches 0 in SHAP value has a negligible contribution to the prediction[60]. A range of graphs are provided by the SHAP framework to help in the understanding of feature contributions and to aid in the interpretation and justification of ML models. The following represents how SHAP values are calculated:

$$\phi_i(x) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}[f(S \cup \{i\}) - f(S)] \tag{8}$$

Where:

Model	t-value (Accuracy)	p-value (Accuracy)	t-value (Precision)	p-value (Precision)	t-value (Recall)	p-value (Recall)	t-value (F1 score)	p-value (F1 score)
BoW_RF	108.7332	2.39E−15	59.19424	5.65E−13	135.5109	3.30E−16	128.4593	5.33E−16
BoW_NB	161.109	6.95E−17	95.1292	7.95E−15	120.9821	9.15E−16	105.8549	3.04E−15
BoW_SVM	85.36398	2.10E−14	67.73762	1.68E−13	66.34988	2.03E−13	51.12919	2.10E−12
BoW_GBM	54.78734	1.13E−12	35.68181	5.28E−11	48.87382	3.15E−12	52.17979	1.75E−12
BoW_LGBM	61.19279	4.19E−13	29.05617	3.30E−10	41.47385	1.37E−11	49.3594	2.88E−12
BoW_XGBoost	91.83215	1.09E−14	62.80512	3.32E−13	101.6696	4.37E−15	96.5374	6.96E−15
BoW_Catboost	82.98394	2.71E−14	42.12589	1.19E−11	120.3859	9.56E−16	63.42425	3.04E−13
BoW_LGBM+KNN+MLP	56.52622	8.54E−13	39.37306	2.19E−11	53.2787	1.45E−12	44.7578	6.94E−12
BoW_RF+KNN+MLP	115.5203	1.39E−15	54.42813	1.20E−12	83.80527	2.48E−14	81.01391	3.37E−14
BoW_GBM+RF Stacking Classifier	95.87232	7.41E−15	76.63942	5.55E−14	125.9309	6.38E−16	104.5914	3.39E−15
BoW_GBM+RF Voting Classifier	126.3773	6.18E−16	67.23083	1.80E−13	89.36265	1.39E−14	74.88783	6.83E−14
RoBERTa_Single Hidden Layer NN	46.94341	4.52E−12	31.96107	1.41E−10	41.38057	1.40E−11	30.30619	2.27E−10
RoBERTa_3 Hidden Layers NN	51.66872	1.91E−12	33.33187	9.70E-11	97.90134	6.14E−15	39.84824	1.96E−11
RoBERTa_BiLSTM+3 Hidden Layers NN	43.79562	8.43E−12	26.53577	7.41E−10	56.63986	8.39E−13	46.28162	5.14E−12
RoBERTa_BiLSTM+CNN	57.76621	7.03E−13	40.85393	1.57E−11	45.50036	5.98E−12	60.10854	4.92E−13

Table 7. 10-fold cross-validated paired t-tests comparing macro-average precision, recall, F1 scores, and accuracy of the top-performing models against the TRABSA model.

$\phi_i(x)$ = the SHAP value for feature i in the context of input x,
N = the set of all features,
S = a subset of features excluding i,
$f(S)$ = the model's prediction with features S,
$f(S \cup \{i\})$ = the model's prediction with features S and feature i included.

This formulation encapsulates the incremental contribution of feature i towards the prediction, considering all possible combinations of features.

SHAP text plot

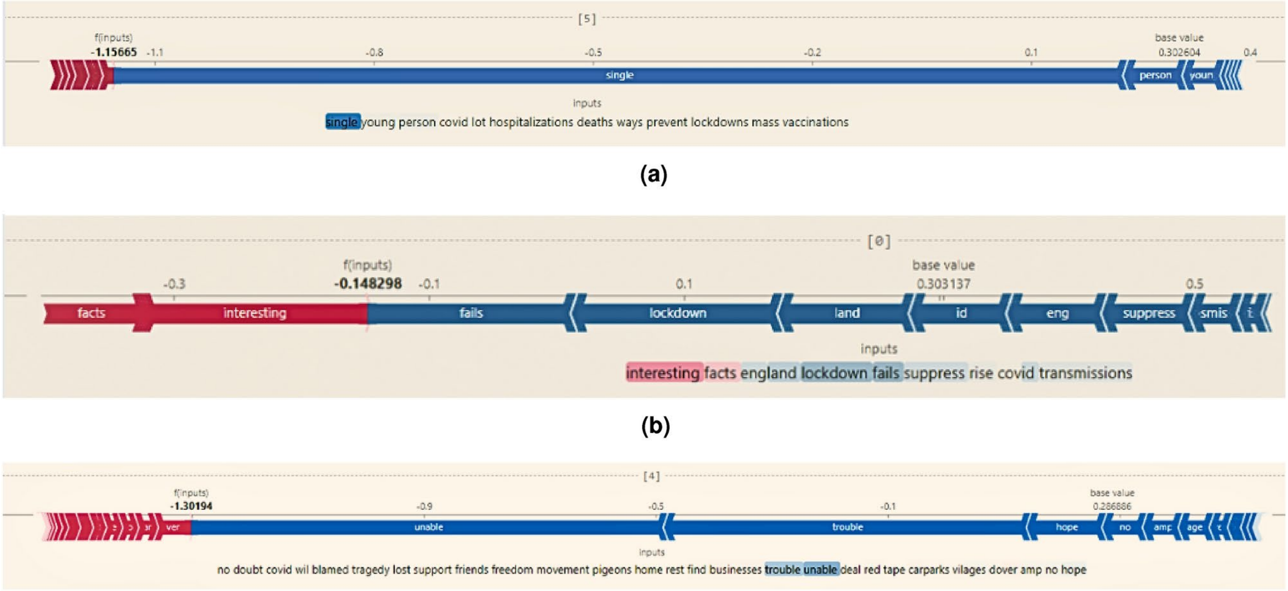
A thorough illustration of how specific tokens inside a text instance affect a TRABSA model's output may be seen in Fig. 8. The graphic illustrates sections that boost (in red) or reduce (in blue) the model's sentiment forecast by superimposing significance values over the original text. This makes it possible to comprehend how certain words or phrases fit into the larger feeling the text is trying to convey nuancedly. Furthermore, the hierarchical structure of significance values preserves the structural linkages between tokens, providing insights into intricate interactions within the text. A comprehensive picture of how the text's combined characteristics affect the model's output is given by the force plot that goes with it; positive features raise the prediction while negative features reduce it. By allowing users to investigate the connections between text segments and how those connections affect the model's predictions, interactive capability further improves interpretability. All things considered, the figure makes it easier to comprehend how the TRABSA model makes decisions and helps with text-based data interpretation and analysis.

SHAP bar plot

The NLP global summaries of the influence of tokens inside a dataset are shown in Fig. 9. Every bar in the graphic illustrates how significantly a token affects the model's predictions over the full dataset. Each bar's height represents the influence of the token; higher bars denote greater significance. The plot collects the individual contributions of tokens over numerous instances by compressing the Explanation object over its rows, usually by summing. This method generates a bar chart with as many columns as there were unique tokens in the original dataset by treating each kind of token as a feature. Big groups are split up, and each token gets an equal portion of the total group significance value if the Explanation object contains hierarchical values. Furthermore, arranging the bar chart in a descending sequence helps reveal which tokens have a major impact on the model's predictions.

LIME

The LIME method provides a methodical technique for evaluating individual predictions given by complicated ML models, which stands for Locally Interpretable Model-Agnostic Explanations[60]. This approach works by estimating the model's behavior around a given forecast. Fundamentally, LIME utilizes a local linear explanation model that follows Eq. (9) to the letter, making it an additive feature attribution method. LIME presents the idea of "interpretable inputs," which are condensed representations of the original inputs and are represented as x_0. A binary vector of interpretable inputs is mapped to the original input space via the transformation $x = h_x(x_0)$. Different kinds of mappings h_x are used for different input spaces. For example, when applied to bag-of-words text characteristics, h_x translates a binary vector (signaling the presence or absence of words) to the appropriate word count in the source text. LIME aims to minimize the objective function to determine the coefficients ϕ:



(a)

(b)

(c)

Fig. 8. The figure illustrates the importance of each token overlaid on the original text corresponding to that token. It showcases the significance of individual tokens in sentiment prediction, where red regions denote parts of the text increasing the model's output (positive sentiment), while blue regions indicate a decrease in the model's output (negative sentiment). **a** Positive sentiment. **b** Neutral sentiment. **c** Negative sentiment.

$$\xi = \arg\min_{g \in G} \left(L(f, g, \pi_{x_0}) + \Omega(g) \right) \tag{9}$$

The loss function in this case is represented by L, which expresses how loyal the explanation model $g(z_0)$ is to the original model $f(h_x(z_0))$. This evaluation is performed across a set of samples weighted by the local kernel π_{x_0} in the reduced input space. Furthermore, the penalty term Ω discourages the explanation model g from being overly complicated. Since g follows Equation 9 and L uses a squared loss formulation, using penalized linear regression methods is typically necessary to solve Eq. (9).

LIME text explainer plot

The LIME Text Explainer visualization is a useful tool for understanding how particular elements or tokens inside a text affect the model's predictions. Every text token is plotted along the horizontal axis in Fig. 10, and its contribution to the prediction is indicated along the vertical axis. Usually, the plot shows how important tokens are by emphasizing how they affect the model's output. Analyzing each token's contribution amount and direction is necessary to understand a LIME Text Explainer plot. Stronger impacts on the model's prediction are indicated by tokens with bigger positive or negative contributions. On the other hand, tokens with contributions closer to zero indicate a negligible influence on the prediction. Additionally, the plot could draw attention to particular words or phrases that greatly impact the model's ability to make decisions. Understanding these influential tokens can provide valuable insights into how the model processes and evaluates textual data.

Discussions

Our methodology and results demonstrate a significant advancement in SA compared to existing literature. While prior studies have explored diverse models and techniques, our TRABSA model introduces a unique hybrid approach combining transformer-based architectures, attention mechanisms, and BiLSTM networks. This innovative combination enables our model to effectively capture nuanced sentiment patterns, resulting in notably higher accuracy and performance across multiple evaluation metrics than traditional and state-of-the-art transformer models such as BERT and RoBERTa. We conducted a thorough analysis to assess the robustness of our TRABSA model across various datasets and scenarios, consistently observing superior performance across multiple datasets, including extended and external ones. Additionally, our model demonstrates resilience to variations in sentiment expression and context, reaffirming its reliability in diverse real-world scenarios.

This novel hybrid approach offers several benefits to the field, including unparalleled accuracy, robustness, and generalizability across diverse datasets and scenarios. By leveraging the strengths of each component, the TRABSA model can revolutionize SA applications, providing researchers, businesses, and policymakers with deeper insights into public opinion, consumer sentiment, and social trends. Its innovative architecture and superior performance represent a significant advancement in the quest for more accurate and reliable SA tools, with implications extending beyond academic research.

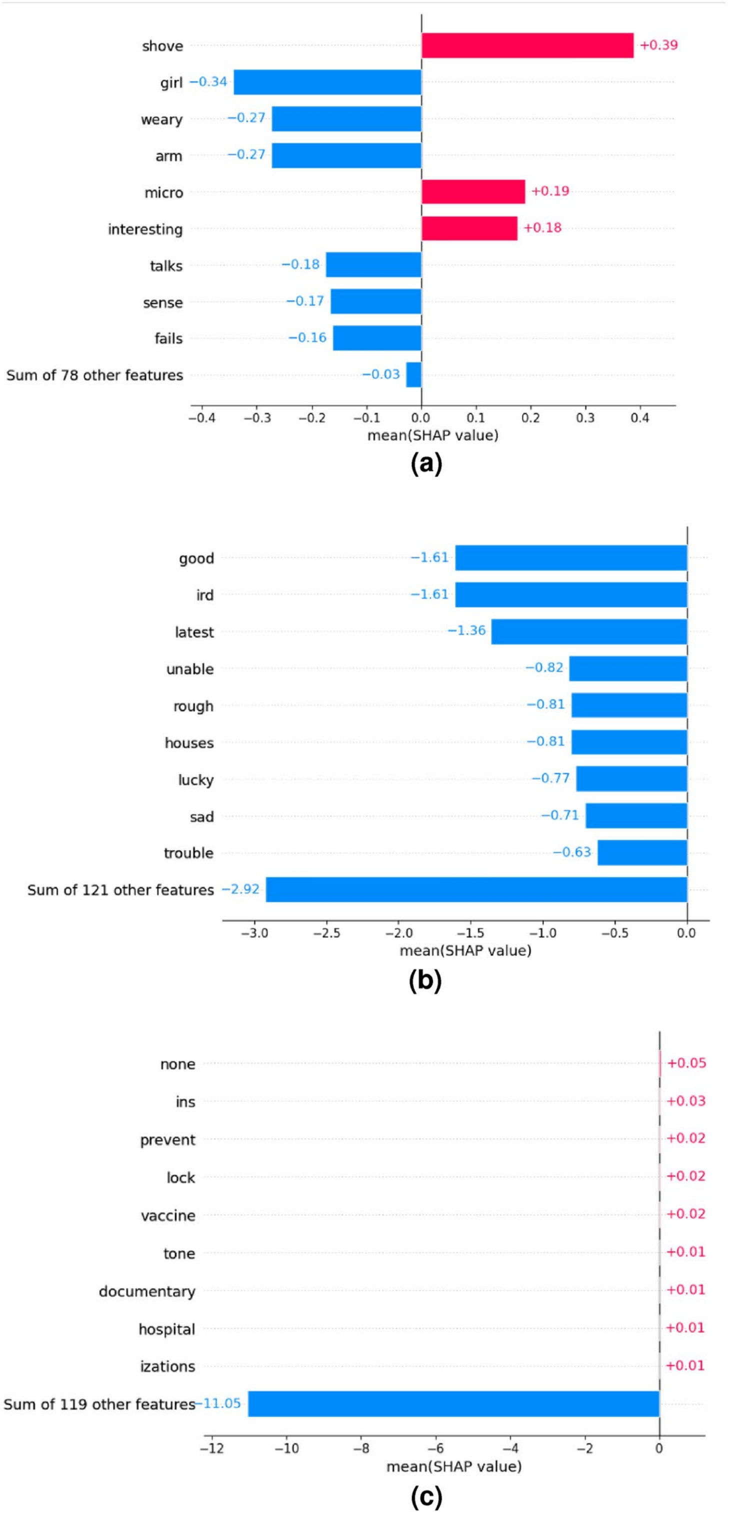
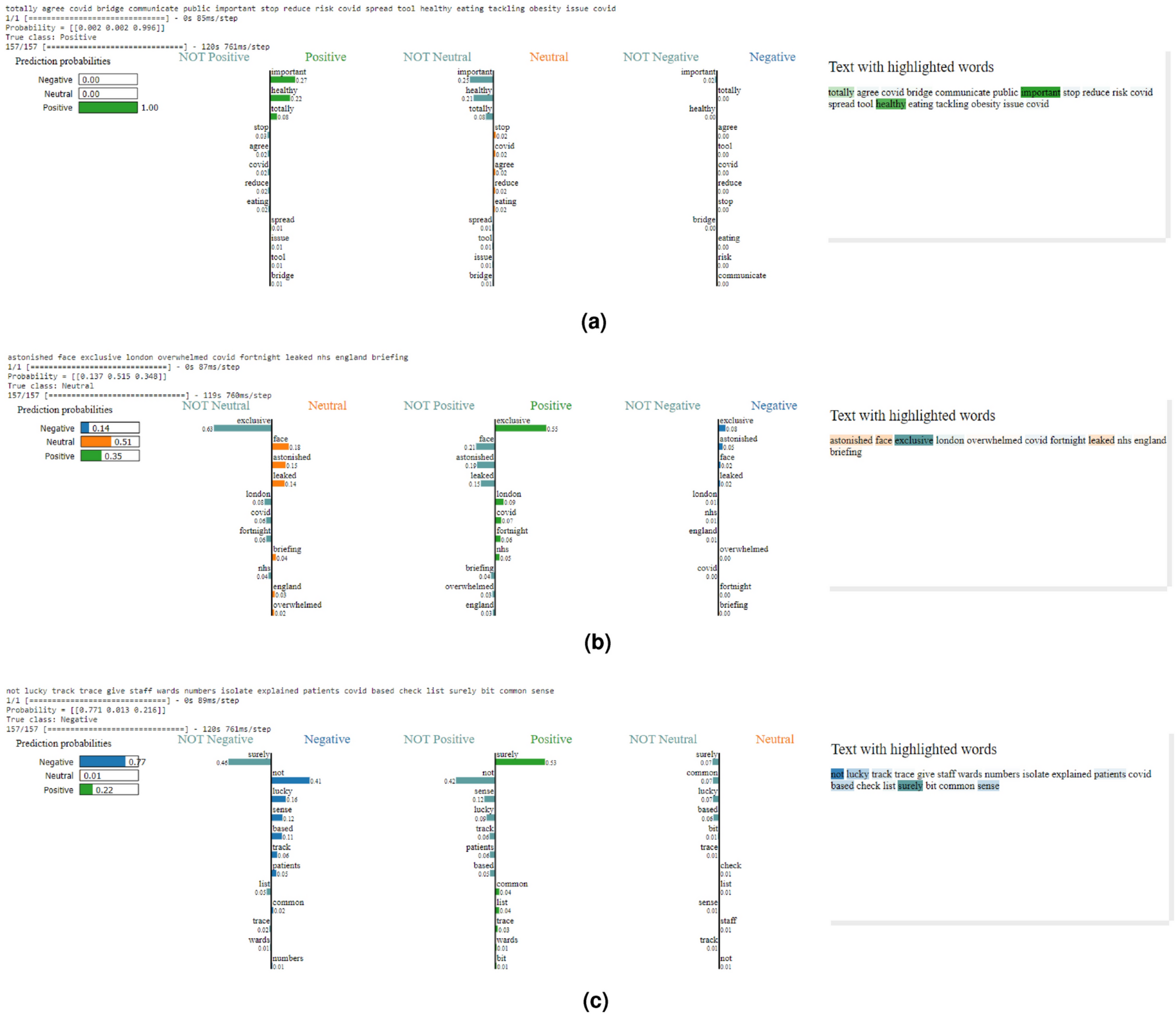


Fig. 9. Summarized importance of tokens in the dataset: (**a**) Neutral tokens displayed in their natural order, (**b**) Negative tokens sorted in descending order, and (**c**) Positive tokens sorted in ascending order. Each bar represents the overall importance of a token, with taller bars indicating greater influence. **a** Neutral tokens displayed in their natural order. **b** Negative tokens sorted in descending order. **c** Positive tokens sorted in ascending order.



```
totally agree covid bridge communicate public important stop reduce risk covid spread tool healthy eating tackling obesity issue covid
1/1 [==============================] - 0s 85ms/step
Probability = [[0.002 0.002 0.996]]
True class: Positive
157/157 [==============================] - 120s 761ms/step
```

Text with highlighted words

totally agree covid bridge communicate public important stop reduce risk covid spread tool healthy eating tackling obesity issue covid

```
astonished face exclusive london overwhelmed covid fortnight leaked nhs england briefing
1/1 [==============================] - 0s 87ms/step
Probability = [[0.137 0.515 0.348]]
True class: Neutral
157/157 [==============================] - 119s 760ms/step
```

Text with highlighted words

astonished face exclusive london overwhelmed covid fortnight leaked nhs england briefing

```
not lucky track trace give staff wards numbers isolate explained patients covid based check list surely bit common sense
1/1 [==============================] - 0s 89ms/step
Probability = [[0.771 0.013 0.216]]
True class: Negative
157/157 [==============================] - 120s 761ms/step
```

Text with highlighted words

not lucky track trace give staff wards numbers isolate explained patients covid based check list surely bit common sense

Fig. 10. LIME Text Explainer vertical bar plot in descending order of token contributions, illustrating the impact of each token on the TRABSA model's predictions. **a** Positive sentiment. **b** Neutral sentiment. **c** Negative sentiment.

The practical implications of the TRABSA model's advancements are profound, offering tangible benefits across various real-world applications. In market research, the model's ability to accurately analyze sentiment from social media, customer reviews, and other online sources empowers companies to gain valuable insights into consumer preferences, market trends, and brand sentiment. This knowledge informs strategic decision-making processes, product development strategies, and marketing campaigns, ultimately enhancing customer satisfaction and competitive advantage. Furthermore, in social media monitoring and reputation management, the TRABSA model equips organizations with tools to monitor public sentiment, identify emerging issues or crises, and proactively respond to customer feedback in real-time. Detecting and addressing potential issues early on enables businesses to safeguard their reputation and maintain positive relationships with their target audience. Additionally, in the context of public opinion analysis and political discourse, the TRABSA model provides policymakers and analysts with a powerful tool for gauging public sentiment, identifying key concerns, and tracking changes in public perception over time. This knowledge informs policy decisions, communication strategies, and crisis management efforts, ultimately contributing to more informed and responsive governance. The practical applications of the TRABSA model extend across a wide range of industries and domains, offering transformative benefits for businesses, governments, and society as a whole.

Conclusions and future directions

Our research has yielded significant findings and contributions to SA. We have achieved remarkable results by developing and evaluating the TRABSA model, a novel hybrid approach combining transformer-based architectures, attention mechanisms, and BiLSTM networks. Leveraging the latest RoBERTa-based transformer model and expanding the datasets, we have demonstrated the TRABSA model's exceptional accuracy and relevance, bridging existing gaps in SA benchmarks. Thorough comparisons of word embedding techniques and methodical labeling of tweets using lexicon-based approaches have further enhanced the effectiveness of SA methodologies. Our experiments and benchmarking efforts have highlighted the superiority of the TRABSA model over traditional and state-of-the-art models, showcasing its versatility and robustness across diverse datasets and scenarios. With macro-average precision of 94%, macro-average recall of 93%, macro-average F1-score of 94%, and accuracy of 94%, our model has proven its efficacy in capturing nuanced sentiment patterns. Additionally, exploring model interpretability techniques using SHAP and LIME has enhanced our understanding and trust in the TRABSA model's predictions, reinforcing its practical applicability.

Despite the significant advancements achieved in our research, several avenues remain for future exploration and improvement in interpretable SA. Firstly, there is scope for refining and expanding model interpretability techniques to provide deeper insights into the factors influencing sentiment predictions. Additionally, integrating multimodal data sources such as text, images, and audio could enhance the richness and accuracy of SA. Addressing ethical considerations regarding bias, fairness, and privacy in SA models is paramount for responsible deployment and usage. Furthermore, exploring the application of SA in emerging domains such as healthcare, finance, and politics could uncover new challenges and opportunities for research and innovation. Overall, continued research in interpretable SA holds the potential to drive meaningful advancements in AI technologies and contribute to more informed decision-making in various fields.

Data availability

The extended datasets, comprising the Global Twitter COVID-19 Dataset and the USA Twitter COVID-19 Dataset, are publicly available for download from the Extended Covid Twitter Datasets (https://data.mendeley.com/datasets/2ynwykrfgf/1) repository[47]. Additionally, the external datasets used in our research were sourced from Kaggle, including the Twitter and Reddit Dataset (https://www.kaggle.com/datasets/cosmos98/twitter-and-reddit-sentimental-analysis-dataset), Apple Dataset (https://www.kaggle.com/datasets/seriousran/appletwittersentimenttexts), and US Airline Dataset (https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment). The code for reproducibility is available in https://github.com/Abrar2652/nlp-roBERTa-biLSTM-attention.

Received: 18 May 2024; Accepted: 10 October 2024
Published online: 22 October 2024

References

1. Zhang, L. & Liu, B. Sentiment analysis and opinion mining. In *Encyclopedia of Machine Learning and Data Mining* (eds. Sammut, C. & Webb, G. I.) 1152–1161 (Springer US, 2017). https://doi.org/10.1007/978-1-4899-7687-1_907.
2. Chaturvedi, S., Mishra, V. & Mishra, N. Sentiment analysis using machine learning for business intelligence. In *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)* 2162–2166 (IEEE, 2017). https://doi.org/10.1109/ICPCSI.2017.8392100.
3. Taboada, M. Sentiment analysis: An overview from linguistics. *Annu. Rev. Linguist.* **2**, 325–347. https://doi.org/10.1146/annurev-linguistics-011415-040518 (2016).
4. Kallam, Y. R. et al. Advancements in sentiment analysis: A deep learning approach. In *2023 IEEE 15th International Conference on Computational Intelligence and Communication Networks (CICN)* 206–210 (IEEE, 2023). https://doi.org/10.1109/CICN59264.2023.10402154.
5. Kumar, V. Spatiotemporal sentiment variation analysis of geotagged COVID-19 tweets from India using a hybrid deep learning model. *Sci. Rep.* **12**, 1849. https://doi.org/10.1038/s41598-022-05974-6 (2022).
6. Jawale, S. & Sawarkar, S. Interpretable sentiment analysis based on deep learning: An overview. In *2020 IEEE Pune Section International Conference (PuneCon)* 65–70 (IEEE, 2020). https://doi.org/10.1109/PuneCon50868.2020.9362361.
7. Cambria, E. Affective computing and sentiment analysis. *IEEE Intell. Syst.* **31**, 102–107. https://doi.org/10.1109/MIS.2016.31 (2016).
8. D'Andrea, A., Ferri, F., Grifoni, P. & Guzzo, T. Approaches, tools and applications for sentiment analysis implementation. *Int. J. Comput. Appl.* **125**, 26–33. https://doi.org/10.5120/ijca2015905866 (2015).
9. Wang, Z., Ho, S.-B. & Cambria, E. Multi-level fine-scaled sentiment sensing with ambivalence handling. *Internat. J. Uncertain. Fuzziness Knowl.-Based Syst.* **28**, 683–697. https://doi.org/10.1142/S0218488520500294 (2020).
10. Yang, B., Shao, B., Wu, L. & Lin, X. Multimodal sentiment analysis with unidirectional modality translation. *Neurocomputing* **467**, 130–137. https://doi.org/10.1016/j.neucom.2021.09.041 (2022).
11. Ray, P. & Chakrabarti, A. A Mixed approach of Deep Learning method and Rule-Based method to improve Aspect Level Sentiment Analysis. *Appl. Comput. Inf.* **18**, 163–178. https://doi.org/10.1016/j.aci.2019.02.002 (2022).
12. Sarker, M. K., Zhou, L., Eberhart, A. & Hitzler, P. Neuro-symbolic artificial intelligence: Current trends. *AI Commun.* **34**, 197–209. https://doi.org/10.3233/AIC-210084 (2022).
13. Aqlan, A. A. Q., Manjula, B. & Lakshman Naik, R. A study of sentiment analysis: concepts, techniques, and challenges. In *Proceedings of International Conference on Computational Intelligence and Data Engineering, vol. 28, Lecture Notes on Data Engineering and Communications Technologies* (eds. Chaki, N., et al.) 147–162 (Springer Singapore, 2019). https://doi.org/10.1007/978-981-13-6459-4_16.
14. Ahmed, J. & Ahmed, M. Classification, detection and sentiment analysis using machine learning over next generation communication platforms. *Microprocess. Microsyst.* **98**, 104795. https://doi.org/10.1016/j.micpro.2023.104795 (2023).
15. Gaur, P., Vashistha, S. & Jha, P. Twitter sentiment analysis using naive bayes-based machine learning technique. In *Sentiment Analysis and Deep Learning, vol. 1423, Advances in Intelligent Systems and Computing*, vol. 1432 (eds. Shakya, S. et al.) 367–376 (Springer Nature Singapore, 2023). https://doi.org/10.1007/978-981-19-5443-6_27.
16. Qi, Y. & Shabrina, Z. Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach. *Soc. Netw. Anal. Min.* **13**, 31. https://doi.org/10.1007/s13278-023-01030-x (2023).
17. Al-sari, B. et al. Sentiment analysis for cruises in Saudi Arabia on social media platforms using machine learning algorithms. *J. Big Data* **9**, 21. https://doi.org/10.1186/s40537-022-00568-5 (2022).
18. Mukherjee, P. et al. Effect of negation in sentences on sentiment analysis and polarity detection. *Procedia Comput. Sci.* **185**, 370–379. https://doi.org/10.1016/j.procs.2021.05.038 (2021).
19. Noori, B. Classification of customer reviews using machine learning algorithms. *Appl. Artif. Intell.* **35**, 567–588. https://doi.org/10.1080/08839514.2021.1922843 (2021).

20. Zahoor, S. & Rohilla, R. Twitter sentiment analysis using machine learning algorithms: a case study. In *2020 International Conference on Advances in Computing, Communication & Materials (ICACCM)* 194–199 (IEEE, 2020). https://doi.org/10.1109/ICACCM50413.2020.9213011.
21. Samuel, J., Ali, G. G. M. N., Rahman, M. M., Esawi, E. & Samuel, Y. COVID-19 public sentiment insights and machine learning for tweets classification. *Information* **11**, 314. https://doi.org/10.3390/info11060314 (2020).
22. Kumar, S., Gahalawat, M., Roy, P. P., Dogra, D. P. & Kim, B.-G. Exploring impact of age and gender on sentiment analysis using machine learning. *Electronics* **9**, 374. https://doi.org/10.3390/electronics9020374 (2020).
23. Zarisfi-Kermani, F., Sadeghi, F. & Eslami, E. Solving the twitter sentiment analysis problem based on a machine learning-based approach. *Evol. Intell.* **13**, 381–398. https://doi.org/10.1007/s12065-019-00301-x (2020).
24. Truică, C.-O., Apostol, E.-S., Şerban, M.-L. & Paschke, A. Topic-based document-level sentiment analysis using contextual cues. *Mathematics* **9**, 145. https://doi.org/10.3390/math9212722 (2021).
25. Petrescu, A., Truica, C.-O., Apostol, E.-S. & Paschke, A. EDSA-Ensemble: an event detection sentiment analysis ensemble architecture. In *IEEE Transactions on Affective Computing, IEEE Transactions on Affective Computing* 1–18 (2024). https://doi.org/10.1109/TAFFC.2024.3434355.
26. Petrescu, A., Truică, C.-O. & Apostol, E.-S. Sentiment analysis of events in social media. In *2019 IEEE 15th International Conference on Intelligent Computer Communication and Processing (ICCP)* 143–149 (2019). https://doi.org/10.1109/ICCP48234.2019.8959677.
27. Truica, C.-O. & Leordeanu, C. A. Classification of an imbalanced data set using decision tree algorithms. *Univ. Politech. Bucharest Sci. Bull. Ser. C Electr. Eng. Comput. Sci.* **79**, 69–84 (2017).
28. Apostol, E.-S., Pisică, A.-G. & Truică, C.-O. ATESA-BÆRT: A heterogeneous ensemble learning model for aspect-based sentiment analysis (2023). https://doi.org/10.48550/arXiv.2307.15920. ArXiv:2307.15920 [cs].
29. Mitroi, M., Truică, C.-O., Apostol, E.-S. & Florea, A. M. Sentiment analysis using topic-document embeddings. In *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)* 75–82 (2020). https://doi.org/10.1109/ICCP51029.2020.9266181.
30. Bansal, V., Tyagi, M., Sharma, R., Gupta, V. & Xin, Q. A transformer based approach for abuse detection in code mixed indic languages *ACM Trans. Asian Low-Resourc. Lang. Inf. Process.* https://doi.org/10.1145/3571818 (2022).
31. Gupta, V. et al. An emotion care model using multimodal textual analysis on COVID-19. *Chaos, Solitons Fract.* **144**, 110708. https://doi.org/10.1016/j.chaos.2021.110708 (2021).
32. Gupta, V., Singh, V. K., Mukhija, P. & Ghose, U. Aspect-based sentiment analysis of mobile reviews. *J. Intel. Fuzzy Syst.* **36**, 4721–4730. https://doi.org/10.3233/JIFS-179021 (2019).
33. Gupta, V., Dass, P. & Arora, R. Pendulating or resonating? A case of echo-chambers in twitter. *J. Discrete Math. Sci. Cryptogr.* **25**, 231–240. https://doi.org/10.1080/09720529.2021.2019442 (2022).
34. Gupta, V. et al. Toward integrated cnn-based sentiment analysis of tweets for scarce-resource language-Hindi. *ACM Trans. Asian Low-Resourc. Lang. Inf. Process.* **20**, 801–8023. https://doi.org/10.1145/3450447 (2021).
35. Basiri, M. E., Nemati, S., Abdar, M., Asadi, S. & Acharrya, U. R. A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets. *Knowl.-Based Syst.* **228**, 107242. https://doi.org/10.1016/j.knosys.2021.107242 (2021).
36. Hayawi, K., Shahriar, S., Serhani, M. A., Taleb, I. & Mathew, S. S. ANTi-Vax: a novel Twitter dataset for COVID-19 vaccine misinformation detection. *Public Health* **203**, 23–30. https://doi.org/10.1016/j.puhe.2021.11.022 (2022).
37. Vishwamitra, N. et al. On Analyzing COVID-19-related Hate speech using BERT attention. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)* 669–676 (IEEE, 2020). https://doi.org/10.1109/ICMLA51294.2020.00111.
38. Chen, N., Zhong, Z. & Pang, J. An exploratory study of COVID-19 information on twitter in the greater region. *Big Data Cogn. Comput.* **5**, 586. https://doi.org/10.3390/bdcc5010005 (2021).
39. Kabir, M. Y. & Madria, S. EMOCOV: Machine learning for emotion detection, analysis and visualization using COVID-19 tweets. *Online Soc. Netw. Media* **23**, 100135. https://doi.org/10.1016/j.osnem.2021.100135 (2021).
40. Valdes, A., Lopez, J. & Montes, M. UACH-INAOE at SMM4H: a BERT based approach for classification of COVID-19 Twitter posts. In *Proceedings of the Sixth Social Media Mining for Health (#SMM4H) Workshop and Shared Task* (eds. Magge, A. et al.) 65–68 (Association for Computational Linguistics, 2021). https://doi.org/10.18653/v1/2021.smm4h-1.10.
41. Tziafas, G., Kogkalidis, K. & Caselli, T. Fighting the COVID-19 Infodemic with a Holistic BERT Ensemble (2021). ArXiv:2104.05745 [cs].
42. Sadia, K. & Basak, S. Sentiment analysis of COVID-19 tweets: How does BERT perform? In *Proceedings of International Joint Conference on Advances in Computational Intelligence* (eds. Uddin, M. S. & Bansal, J. C.) 407–416 (Springer, 2021). https://doi.org/10.1007/978-981-16-0586-4_33.
43. Song, X. et al. Classification aware neural topic model for COVID-19 disinformation categorisation. *PLOS ONE* **16**, e0247086. https://doi.org/10.1371/journal.pone.0247086 (2021).
44. Hossain, T. et al. COVIDLies: Detecting COVID-19 misinformation on social media. In *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020* (eds. Verspoor, K. et al.) (Association for Computational Linguistics, 2020). https://doi.org/10.18653/v1/2020.nlpcovid19-2.11.
45. Chintalapudi, N., Battineni, G. & Amenta, F. Sentimental analysis of COVID-19 tweets using deep learning models. *Infect. Dis. Rep.* **13**, 329–339. https://doi.org/10.3390/idr13020032 (2021).
46. Sloan, L. & Morgan, J. Who tweets with their location? Understanding the relationship between demographic characteristics and the use of geoservices and geotagging on twitter. *PLOS ONE* **10**, e0142209. https://doi.org/10.1371/journal.pone.0142209 (2015).
47. Jahin, M. A. Extended Covid twitter datasets. https://doi.org/10.17632/2ynwykrfgf.1 (2023).
48. Naseem, U., Razzak, I., Khushi, M., Eklund, P. W. & Kim, J. COVIDSenti: A large-scale benchmark twitter data set for COVID-19 sentiment analysis. *IEEE Trans. Comput. Social Syst.* **8**, 1003–1015. https://doi.org/10.1109/TCSS.2021.3051189 (2021).
49. dos Santos Neto, M. V., da Silva, N. F. F. & da Silva Soares, A. A survey and study impact of tweet sentiment analysis via transfer learning in low resource scenarios. *Lang. Resourc. Eval.* **58**, 133–174. https://doi.org/10.1007/s10579-023-09687-8 (2024).
50. Brum, H. & Volpe Nunes, M. d. G. Building a sentiment corpus of tweets in Brazilian Portuguese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)* (eds. Calzolari, N. et al.) (European Language Resources Association (ELRA), 2018).
51. De Souza, J. G. R., De Paiva Oliveira, A. & Moreira, A. Development of a Brazilian Portuguese Hotel's reviews corpus. In *Computational Processing of the Portuguese Language, vol. 11122 , Lecture Notes in Computer Science* (eds. Villavicencio, A. et al.) 353–361, https://doi.org/10.1007/978-3-319-99722-3_36 (Springer International Publishing, 2018).
52. Pilar, G.-D., Isabel, S.-B., Diego, P.-M. & José-Luis, G. A novel flexible feature extraction algorithm for Spanish tweet sentiment analysis based on the context of words. *Expert Syst. Appl.* **212**, 118817. https://doi.org/10.1016/j.eswa.2022.118817 (2023).
53. Su, Y. & Kabala, Z. J. Public perception of ChatGPT and transfer learning for tweets sentiment analysis using Wolfram mathematica. *Data* **8**, 180. https://doi.org/10.3390/data8120180 (2023).
54. Memiş, E., Akarkamçı-(Kaya), H., Yeniad, M., Rahebi, J. & Lopez-Guede, J. M. Comparative study for sentiment analysis of financial tweets with deep learning methods. *Appl. Sci.* **14**, 588. https://doi.org/10.3390/app14020588 (2024).
55. Kp, V., Ab, R., Hl, G., Ravi, V. & Krichen, M. A tweet sentiment classification approach using an ensemble classifier. *Int. J. Cogn. Comput. Eng.* **5**, 170–177. https://doi.org/10.1016/j.ijcce.2024.04.001 (2024).
56. Mohbey, K. K., Meena, G., Kumar, S. & Lokesh, K. A CNN-LSTM-based hybrid deep learning approach for sentiment analysis on monkeypox tweets. *N. Gener. Comput.* **42**, 89–107. https://doi.org/10.1007/s00354-023-00227-0 (2024).

57. Sazan, S. A., Ahmed, M., Saad, T. B. & Roy, M. Advanced natural language processing techniques for efficient sentiment analysis of US airline twitter data: a high-performance framework for extracting insights from tweets. In *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)* 01–06 (2024). https://doi.org/10.1109/ICEEICT62016.2024.10534511.
58. Jlifi, B., Abidi, C. & Duvallet, C. Beyond the use of a novel Ensemble based Random Forest-BERT Model (Ens-RF-BERT) for the Sentiment Analysis of the hashtag COVID19 tweets. *Soc. Netw. Anal. Min.* **14**, 88. https://doi.org/10.1007/s13278-024-01240-x (2024).
59. Bhardwaj, M., Mishra, P., Badhani, S. & Muttoo, S. K. Sentiment analysis and topic modeling of COVID-19 tweets of India. *Int. J. Syst. Assurance Eng. Manage.* **15**, 1756–1776. https://doi.org/10.1007/s13198-023-02082-0 (2024).
60. Jahin, M. A. et al. QAmplifyNet: pushing the boundaries of supply chain backorder prediction using interpretable hybrid quantum-classical neural network. *Sci. Rep.* **13**, 18246. https://doi.org/10.1038/s41598-023-45406-7 (2023).

Author contributions

M.A.J.: Conceptualization, Methodology, Data curation, Writing - Original Draft Preparation, Software, Visualization, Investigation. M.S.H.S.: Writing - Original Draft Preparation. M.F.M.: Conceptualization, Supervision, Reviewing, and Editing. M.R.I.: Conceptualization, Supervision, Reviewing, and Editing. Y.W.: Conceptualization, Supervision, Reviewing.

Competing interests

The authors have no conflict of interest to declare that are relevant to this article.

Additional information

Correspondence and requests for materials should be addressed to M.F.M. or M.R.I.

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.