

Lexicon and Deep Learning-Based Approaches in Sentiment Analysis on Short Texts

Taminul Islam^{1,2} , Md. Alif Sheakh¹, Md. Rezwane Sadik³, Mst. Sazia Tahosin¹,
Md. Musfiqur Rahman Foysal¹, Jannatul Ferdush⁴ , Mahbuba Begum^{5*} 

¹Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

²Department of Computer Science, Southern Illinois University, Carbondale, IL, USA

³Department of Economics & Decision Sciences, University of South Dakota, Vermillion, SD, USA

⁴Department of Computer Science and Engineering, Jashore University of Science and Technology, Jashore, Bangladesh

⁵Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Tangail, Bangladesh

Email: *mahbub327@yahoo.com, *mahbubacse@mbstu.ac.bd

How to cite this paper: Islam, T., Sheakh, Md.A., Sadik, Md.R., Tahosin, Mst.S., Foysal, Md.M.R., Ferdush, J. and Begum, M. (2024) Lexicon and Deep Learning-Based Approaches in Sentiment Analysis on Short Texts. *Journal of Computer and Communications*, 12, 11-34.

<https://doi.org/10.4236/jcc.2024.121002>

Received: December 6, 2023

Accepted: January 5, 2024

Published: January 8, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Social media is an essential component of our personal and professional lives. We use it extensively to share various things, including our opinions on daily topics and feelings about different subjects. This sharing of posts provides insights into someone's current emotions. In artificial intelligence (AI) and deep learning (DL), researchers emphasize opinion mining and analysis of sentiment, particularly on social media platforms such as Twitter (currently known as X), which has a global user base. This research work revolves explicitly around a comparison between two popular approaches: Lexicon-based and Deep learning-based Approaches. To conduct this study, this study has used a Twitter dataset called sentiment140, which contains over 1.5 million data points. The primary focus was the Long Short-Term Memory (LSTM) deep learning sequence model. In the beginning, we used particular techniques to preprocess the data. The dataset is divided into training and test data. We evaluated the performance of our model using the test data. Simultaneously, we have applied the lexicon-based approach to the same test data and recorded the outputs. Finally, we compared the two approaches by creating confusion matrices based on their respective outputs. This allows us to assess their precision, recall, and F_1 -Score, enabling us to determine which approach yields better accuracy. This research achieved 98% model accuracy for deep learning algorithms and 95% model accuracy for the lexicon-based approach.

Keywords

Opinion Mining, Lexicon Analysis, Twitter Data, LSTM, Machine Learning

1. Introduction

Opinion mining, also known as sentiment analysis, has gained significant attention in recent years as a result of the exponential growth of user-generated content on social media platforms such as Twitter. Estimating and categorizing people's expressed thoughts, sentiments, and attitudes in the text is known as sentiment analysis [1]. This information can be useful for various objectives, including market research, brand management, public opinion analysis, and decision-making. Sentiment analysis can be performed using either a lexicon-based or a deep learning-based method. The lexicon-based technique is based on established sentiment dictionaries or lexicons that contain words and their associated sentiment polarity [2]. It assigns sentiment labels to text depending on the presence and context of certain terms. On the other hand, deep learning-based techniques use artificial neural networks, notably recurrent neural networks (RNNs) and variants such as Long Short-Term Memory (LSTM), to learn sentiment patterns and characteristics automatically from enormous volumes of training data [3].

This research focuses on analyzing and comparing the performance of these two approaches in sentiment analysis using Twitter data. By analyzing both approaches which are lexicon-based and deep learning-based, we can get insights into their strengths, limits, and potential applications in extracting sentiment information from social media content [4]. Twitter has become a popular forum for users to share their thoughts on various issues, including products, services, events, and current events [5]. Its microblogging nature, with restricted characters in each tweet, poses a unique obstacle for sentiment research. Because of the brevity of tweets, specialized procedures are required to capture the sentiment nuances effectively.

The lexicon-based approach provides a simple and interpretable mechanism for sentiment analysis. By correlating words with sentiment polarity in a predetermined lexicon, it can instantly classify the sentiment of a tweet based on the presence of positive or negative phrases [6]. However, lexicons may not capture the context-dependent mood, irony, or slang, resulting in erroneous results. Deep learning-based techniques, notably LSTM-based models, have shown considerable potential in capturing contextual information and dependencies inside text sequences [7]. These models can effectively capture the sentiment represented in lengthier and more complex tweets. They can learn from enormous amounts of data and adapt to varied language patterns and user expressions. Deep learning models, on the other hand, necessitate significant computational resources and a large amount of training data to reach optimal performance [8] [9]. This study compares the strengths and disadvantages of lexicon-based and deep learning-based approaches to sentiment analysis on Twitter data [10]. It will evaluate their accuracy, efficiency, adaptation to diverse domains and languages, and capacity to deal with obstacles such as noisy data, sarcasm, and context-dependent sentiment. The findings of this study will help researchers, practitioners, and de-

cision-makers choose the best strategy for sentiment analysis tasks using Twitter data.

Furthermore, this research will help to expand sentiment analysis techniques and increase the researcher's knowledge of the sentiments and opinions expressed by people on social media sites. The findings may have repercussions in a variety of fields, including marketing, public opinion analysis, customer feedback analysis, and political attitude analysis. Ultimately, the goal is to employ sentiment analysis to gather important insights, make educated decisions, and improve user experiences in the age of social media and online communication. Study contributions are listed below:

- This study compares lexicon-based and deep learning-based sentiment analysis methods for sentiment analysis, specifically Twitter posts.
- A transparent and reproducible method for data collection, labeling, preprocessing, and model evaluation is described.
- Showed that deep learning model can achieve substantial performance for sentiment analysis on Twitter, reaching 98% accuracy.
- We used accuracy, recall, F_1 -Score, and confusion matrix to compare deep learning with lexicon-based methods on a large Twitter dataset.
- Showed long short-term memory (LSTM) networks as an effective deep learning architecture for sentiment classification of short social media posts.
- Analyzed trends and performance trade-offs between the two approaches—deep learning had higher recall while lexicon was more precise.

The structure of the paper is as follows: It contains an overview of relevant research and current knowledge in the field in the Literature Review section. The methodology part is divided into ten sections, which are: Proposed System, Data Description, Data Pre-Processing, Data Visualization, Label Encoding, Feature Ex-traction using Word Embedding, Model Training, Optimization Algorithm, Callbacks, and Lexicon Approach. Following are the results: the Confusion Matrix, Accuracy, Precision, Recall, and F_1 -score are all listed in the Experimental Results section. During the Discussion part, the comparison analysis's primary results are examined. As a final touch in conclusion, the paper summarizes the main study results and essential insights.

2. Literature Review

This section contains an overview of the relevant works and studies associated to the comparative analysis.

M. Sani *et al.* [11] analyze Hausa tweets' emotions using machine learning. The research examines how machine learning and lexicon-based techniques might improve Hausa text categorization. The BBC Hausa Twitter API provided the paper's dataset. Two machine learning-based classification techniques were Multinomial Naive Bayes (MNB) and Logistic Regression (LR) employing Count Vectorizer and TF-IDF. LR classifies Hausa texts 86% better than MNB. The paper examines how Hausa's informality and lack of structure make sentiment

analysis challenging. The paper's focus on Hausa tweet sentiment analysis limits its applicability to other text data.

M. ur-Rehman *et al.* [12] evaluate actual time sentiment analysis with SVM, K-SVM, and Multinomial Naive Bayes. Twitter and contested territory news websites are used for the study. The work examines important supervised machine learning classifiers for real-time sentiment analysis and suggests a technique for new researchers to investigate real-time sentiment correlation between pairs of countries to predict conflicts with substantial casualties. This study employs SVM, Kernel SVM, and Multinomial Naive Bayes for sentiment analysis. The Kaggle dataset trains real-time sentiment classifiers. K-SVM's real-time positive sentiment recognition is 80% accurate. Multinomial Naive Bayes and Real-time SVMs performed poorly. This study examines how real-time data distorts sentiment analysis. Update language models, add data sources, and adapt sentiment analysis algorithms to language use.

N. Imanina Zabha *et al.* [13] they have worked on sentiment analysis by focusing on lexicon methods for the cross-lingual sentiment analysis from Twitter data. Their final result is revealed that the classifier was able to determine the sentiments finally. A. Mitra *et al.* [14] worked on a movie data set with the help of lexicon approaches and tried to find out which movie is best or what type of categories movie people would like to watch, which will enhance the movie producer or theater owner to adopt that kind of categories movies for public to make a good business out of it.

M. S. Hajrahimova and M. I. Ismaylova *et al.* [15] analyze Twitter users' COVID-19 attitudes using machine learning. This article analyses pandemic tweets' emotional "color" using Support Vector Machine, Naive Bayes, Random Forest, and Neural Network methods. Python and scikit-learn are used to test COVID-19 tweets from Kaggle. The Random Forest classifier, with 79.37% accuracy, performs best in experiments. The paper uses machine learning to study Twitter users' COVID-19 pandemic reactions. The article only analyses COVID-19 tweets and excludes other factors that may affect Twitter users' sentiments.

N. M. Sham and A. Mohamed *et al.* [16] want to find the best sentiment analysis approach for tweets on climate change and related topics by comparing multiple methodologies. The paper compares seven lexicon-based methods, three machine learning classifiers, and hybrid algorithms using climate change tweets. The hybrid technique outperformed the other two with an F_1 -Score of 75.3%. Lemmatization improved machine learning and hybrid techniques' accuracy by 1.6%, whereas TF-IDF feature extraction was marginally better than BoW, increasing the Logistic Regression classifier by 0.6%. Near the end of the paper, the authors recommend additional research on deep learning algorithms and domain-specific sentiment in social media.

B. S. Ainapure *et al.* [17] examined Indian tweets about the COVID-19 pandemic and vaccine campaign. Deep learning and lexicons sort sentiments. Training a recurrent neural network with Bi-LSTM and GRU techniques yields 92.70%

and 91.24% on the COVID-19 dataset and 92.48% and 93.03% on vaccination tweet categorization. Healthcare workers and governments can use the models to make pandemic decisions. The study only considers Indian opinions.

R. Srivastava *et al.* [18] compare unsupervised lexicon-based system and machine learning for sentiment analysis. The dataset included 20,000 TripAdvisor hotel reviews. The authors evaluated their performance after training and testing classifiers on cleaned and preprocessed data. The Support Vector Machine model achieved 96.3 percent TFIDF accuracy and 88.7 percent VADER lexical correctness. Machine learning outperforms un-supervised lexicon-based sentiment analysis in this paper's comparison. Its single dataset may make extrapolating its conclusions to other datasets difficult.

Zvonarev *et al.* [19] compare Logistic Regression (LR), Convolutional Neural Network (CNN), XGBoost classifier, and models for Russian tweet tone analysis. The study compares two binary classification-based text tone analysis methods. The dataset is Russian tweets, and models are compared for accuracy and F₁-Score. CNN has the highest accuracy (79.5%) but the longest computation duration. The study also indicates how Russian language unpredictability requires rigorous data pre-processing by any model. The report advises that scientists fine-tune the hyperparameters of boosting type models and broaden the methodologies used to improve the strategy.

N. Braig *et al.* [20] analyze sentiment analysis literature using COVID-19 Twitter data. This research informs policymakers and public health experts on how to use sentiment analysis to stop COVID-19. Researchers examined 40 publications and five databases from October 2019 to January 2022. BERT and RoBERTa models perform best on Twitter data, although ensemble models with various machine learning classifiers perform best overall. The report includes a summary of canonical ML classification techniques and a complete list of characteristics. Lexicon-based sentiment analysis approaches, which were not included in the study, could improve the research.

Although previous research has looked at sentiment analysis methods that utilized lexicons or machine learning, there hasn't been any direct comparison of different methods on the same dataset to measure how they differ in terms of accuracy, precision, recall, and other metrics. A lot of research has limited generalizability by focusing on specific languages or domains, such as Hausa or COVID-19 tweets. Very few studies compared different methods simultaneously. Furthermore, researchers were unable to compare state-of-the-art methods because most studies lacked a reliable experimental methodology. By comparing deep learning with lexicon-based sentiment analysis on a massive English Twitter dataset, this study hopes to fill up some of those gaps. The benefits and drawbacks can be better understood with the use of a quantitative and systematic evaluation approach that examines not only accuracy but also other indicators. To serve as a standard for the community, the study also releases the code for data preprocessing, model implementation, and evaluation. This work adds to our scientific knowledge of the best methods for social media text by thoroughly

comparing the top approaches on standard sentiment analysis datasets. **Table 1** presents the comparison at a glance between different research experiments on sentiment analysis based on short texts.

3. Methodology

This part detailed the proposed approach, the system model, and how we completed this task.

3.1. Proposed System

In this section, we outline the methodology workflow, and the overall workflow is visually presented in **Figure 1**.

- Obtain the dataset from the targeted Twitter platform.
- Label the collected data for subsequent processing.
- Cleanse the dataset through various pre-processing steps.
- Divide the dataset into two segments: Training and Testing sets.
- Employ deep learning techniques, specifically Long Short-Term Memory (LSTM), to train the model using the training set.
- Assess sentiment detection accuracy using the testing set with the trained deep learning model.
- Apply the lexicon-based approach to test sentiment detection accuracy using the same testing set.
- Conduct a comparative analysis of the accuracies obtained from the deep

Table 1. Comparison table with the existing works related to short text and short texts.

Ref	Contribution	Dataset	Algorithms	Best Accuracy
This Work	An examination at how lexicon-based and deep learning-based approaches to Twitter sentiment analysis vary	Sentiment140 dataset containing 1.6 million tweets.	Lexicon-based approach and Long Short-Term Memory (LSTM) in deep learning	98% (LSTM)
[11]	Twitter sentiment analysis in Hausa with the use of machine learning and lexicon-based techniques	BBC Hausa Twitter API.	Logistic Regression, Multinomial Naive Bayes (MNB) with Count Vectorizer and TF-IDF	86% (LR)
[12]	Real-time sentiment analysis using SVM, K-SVM, and Multinomial Naive Bayes.	Twitter data and contested territory news websites.	Support Vector Machine (SVM), Kernel SVM (K-SVM), Multinomial Naive Bayes	80% (K-SVM)
[13]	Comparative Research of Lexicon and Machine Learning Techniques for Sentiment Analysis	TF-IDF	Logistic regression, support vector machine (SVM), logistic regression, AFINN, and Vader lexicon	96.3% (SVM)
[15]	Analysis of Twitter users' COVID-19 attitudes using machine learning.	COVID-19 tweets from Kaggle.	Naive Bayes, Random Forest, Support Vector Machine, Neural Network	79.37% (Random Forest)
[17]	Using deep learning and lexicons, Indian tweets on COVID-19 and vaccination are analyzed.	COVID-19 and vaccination tweets from India.	Recurrent Neural Network (RNN) with Bi-LSTM and GRU techniques	93.03% (GRU)

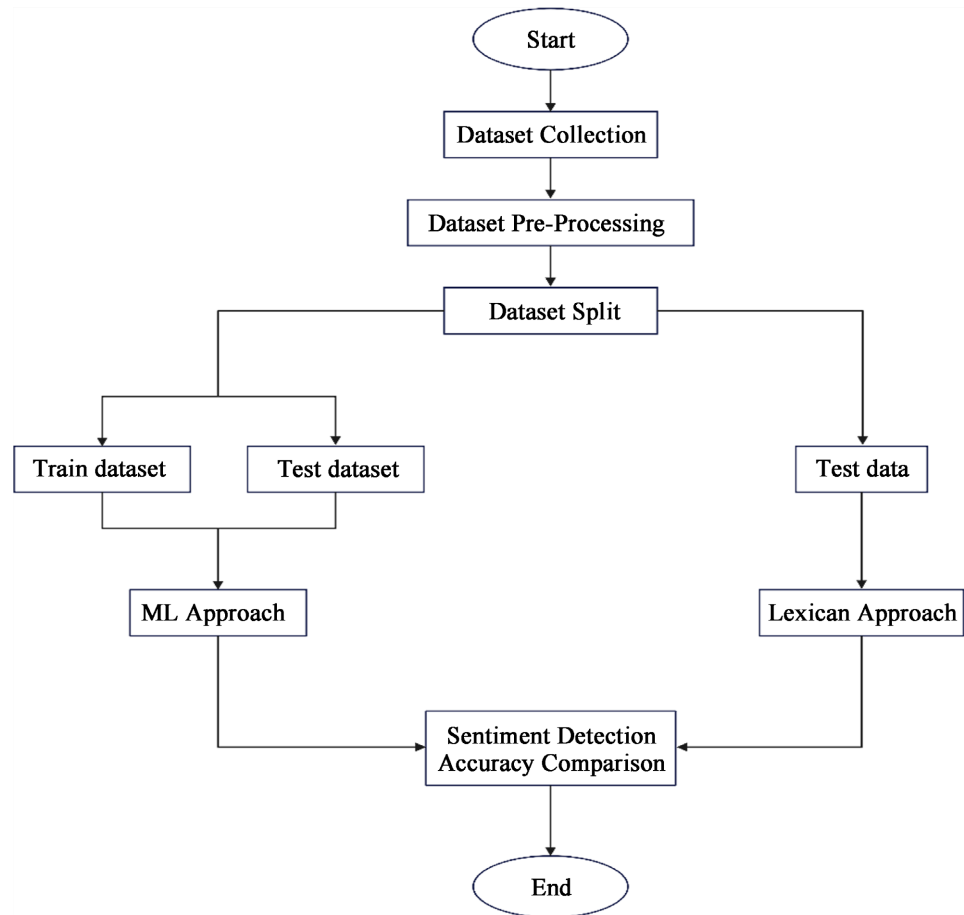


Figure 1. Methodology workflow diagram.

learning model and the lexicon approach.

3.2. Data Description

The Sentiment140 dataset [21], managed by Stanford University, was used to train the algorithm. There are 1.6 million tweets in it. The polarity, tweet ID, date, username, and tweet text are all included in this dataset. The polarity of the tweet and its language are the two most important columns for the goals of this project.

There were no tweets with neutral designations despite the suggestion of a neutral class. This dataset consists of two equally balanced types (positive and negative) with no skewness. There was no need to use any target class balancing strategies because the dataset was balanced correctly.

The columns of the dataset are shown in **Figure 2**. Features like tweet IDs, tweet posting dates, and usernames have been dropped because they don't improve categorization. It is clear from the sentiment column that 0 denotes a negative and 4 denotes a positive. According to the general labeling guidelines of sentimental analysis, the positive class, formerly represented by 4, is now mapped to 1. With a split ratio of 0.8, 0.1, and 0.1, the dataset was divided into train, validation, and test sets for the experiment. Here the K-fold, cross-validation

is applied for the training to strengthen the approached model of deep learning. This is a good split ratio, according to AY Ng; the main test set size is significantly much less than from the train set. From Reddit the test data are scraped which is for some months and can close to almost 100,000 comments, accurately similar to the Senti-ment140 dataset, this fits in exceptionally well with the project’s framework. The size of the validation and test sets is the same. The optimal approach is shuffling the entire dataset before splitting it, which strengthens the model. For our research work, we found a good amount of data to train our model. In our dataset, there were no null values, which proves that there is no missing data on this Dataset. To check if our dataset is balanced or not, we used the data distribution plot shown in **Figure 3**. Equal distribution of data is a sign of a good dataset.

	sentiment	id	date	query_string	user	text
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all...
...
1599995	4	2193601966	Tue Jun 16 08:40:49 PDT 2009	NO_QUERY	AmandaMarie1028	Just woke up. Having no school is the best fee...

Figure 2. Main data in sentiment140 dataset.

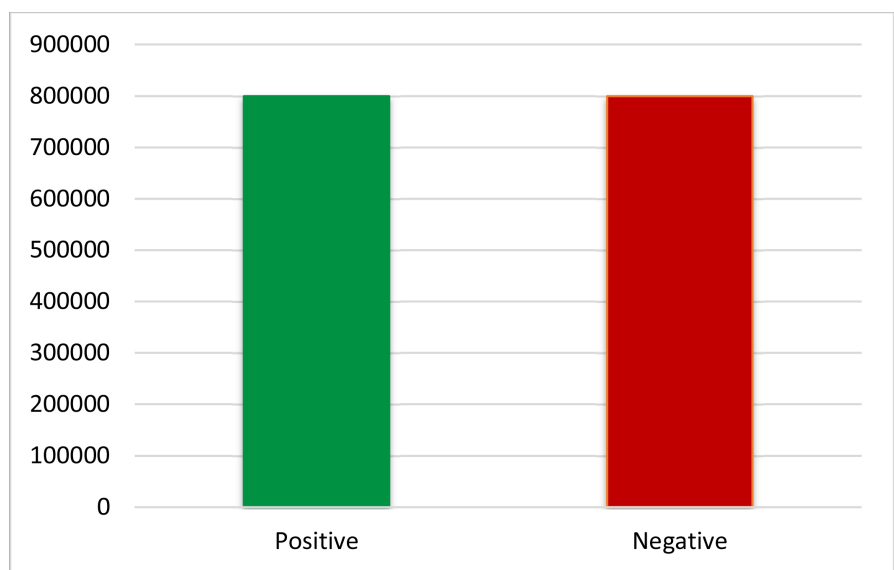


Figure 3. Sentiment data distribution.

3.3. Data Pre-Processing

Text model processing is an important part of working with Twitter data. Because Twitter tweets are limited in length and contain noise, it is important to deal with data spread, use the right methods, and put an emphasis on cleaning. Twitter users can only use a limited number of characters to say what they want to say, so preparing short and frequent data for labeling is necessary.

Prior processing gets rid of unnecessary and duplicate data so that the focus can be on the message itself. This not only makes it easier to pull out meaningful material, but it also makes computing simpler.

As shown in **Figure 4**, the dataset now has column names for “sentiment,” “id,” “date,” “query,” “user_id,” and “text.” After that, columns like “id,” “date,” “query,” and “user_id” that were not needed for mood analysis were taken out of the dataframe. Also, “0” and “1” were changed to “Negative” and “Positive,” respectively, for the sentiment numbers.

3.3.1. Cleaning Dataset

Now the “text” column contains neat text. Tweet content frequently includes references to other users, linked text, emojis, and punctuation. We cannot allow certain texts to be used for training a model to learn utilizing a language model. We must thus use a variety of preprocessing and cleansing techniques to clear the text data.

The following actions were taken to sanitize the data:

- HTML decoding was used to generate the text in the tweets. It is considerably easier to decode the text using the Python module BeautifulSoup.

0	1	2	3	4	5
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_ @switchfoot http://twitpic.com/2y1z1 - Awww, t...
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton is upset that he can't update his Facebook by ...
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus @Kenichan I dived many times for the ball. Man...
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli @nationwideclass no, it's not behaving at all....
5	0	1467811372	Mon Apr 06 22:20:00 PDT 2009	NO_QUERY	joy_wolf @Kwesidel not the whole crew
6	0	1467811592	Mon Apr 06 22:20:03 PDT 2009	NO_QUERY	mybirch Need a hug
7	0	1467811594	Mon Apr 06 22:20:03 PDT 2009	NO_QUERY	coZZ @LOLTrish hey long time no see! Yes.. Rains a...
8	0	1467811795	Mon Apr 06 22:20:05 PDT 2009	NO_QUERY	2Hood4Hollywood @Taliana_K nope they didn't have it
9	0	1467812025	Mon Apr 06 22:20:09 PDT 2009	NO_QUERY	mimismo @twittera que me muera ?
10	0	1467812416	Mon Apr 06 22:20:16 PDT 2009	NO_QUERY	erinx3leannexo spring break in plain city... It's snowing
11	0	1467812579	Mon Apr 06 22:20:17 PDT 2009	NO_QUERY	pardonlauren I just re-pierced my ears
12	0	1467812723	Mon Apr 06 22:20:19 PDT 2009	NO_QUERY	TLeC @caregiving I couldn't bear to watch IL And ...
13	0	1467812771	Mon Apr 06 22:20:19 PDT 2009	NO_QUERY	robobbierobert @octolinz16 It it counts, idk why I did either...
14	0	1467812784	Mon Apr 06 22:20:20 PDT 2009	NO_QUERY	bayofwolves @smarrison I would've been the first, but I di...

Figure 4. Dataset all columns.

- The word “@” is frequently used in the text, but it serves no purpose in this context and can be removed because it adds nothing meaningful to the tweet.
- URL addresses are also unnecessary and can be disregarded for this purpose.
- Contradiction Words “is not” from “it’s not”; “doesn’t” from the “does not” have been mapped using mapping.

When the punctuation has been removed; then some of the newly acquired words lose all meaning. For instance, “isn’t” would become “isn’t” once the punctuation is removed. A dictionary is constructed to deal with such negative words, as seen in **Figure 5**, where the original words serve as the keys and the new words as the values. Now the dictionary replaces every negative word that fits the key with its corresponding values—these aid in improved sentiment recognition.

By changing the word’s case to lowercase, the punctuation (unique letters) is deleted, and the case insensitivity issue is resolved. The words in these sentences are later rejoined after being tokenized. All 1.6 million tweets go through this five-step process in groups. There are 16 batches total, with 100,000 tweets in each set. **Figure 6** shows the entire cleaning procedure in pictorial form. After cleaning, the data frame had roughly 3500 null elements removed because they weren’t helpful for the analysis. In the above figure, we can see the whole cleaning

```
#All the negative words are mapped to "*** not" as it is easy to identify if it comes under negative category
negations_dic = {"isn't": "is not", "aren't": "are not", "wasn't": "was not", "weren't": "were not",
                 "haven't": "have not", "hasn't": "has not", "hadn't": "had not", "won't": "will not",
                 "wouldn't": "would not", "don't": "do not", "doesn't": "does not", "didn't": "did not",
                 "can't": "can not", "couldn't": "could not", "shouldn't": "should not", "mightn't": "might not",
                 "mustn't": "must not" }
```

Figure 5. Dictionary built to handle negative words.

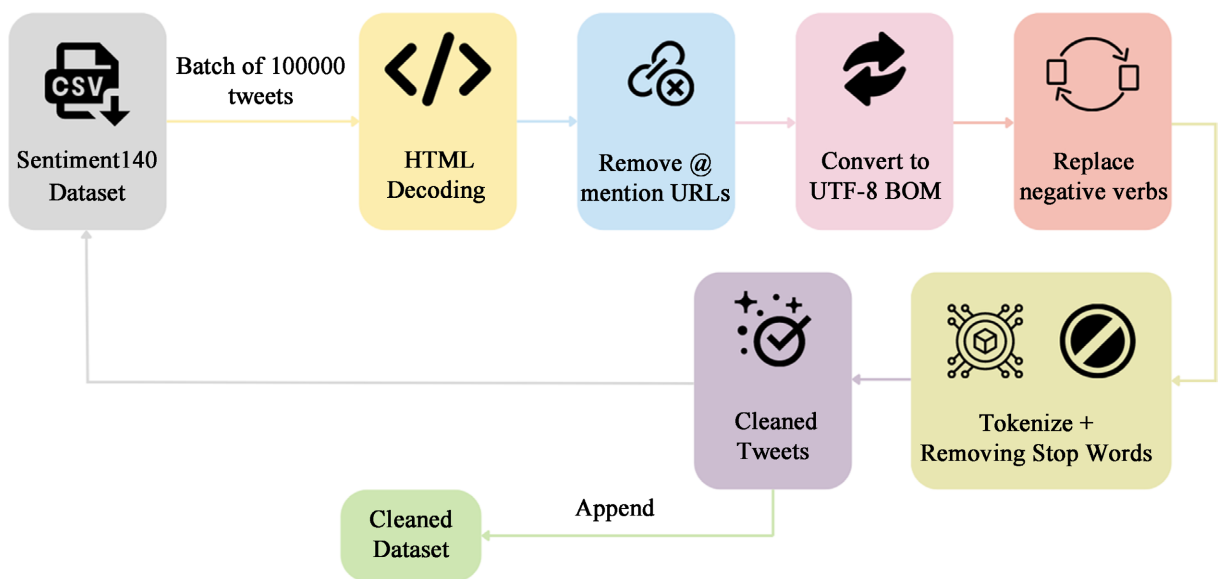


Figure 6. Data cleaning process.

process of data that we have done for implementing our work.

3.3.2. Text Pre-Processing

Short text content frequently includes references to other users, links, emojis, and punctuation [22]. We cannot allow specific texts to be used for training a model to learn utilizing a language model. Thus, we must use various preprocessing and cleansing techniques to clear the text data.

For many grammatical reasons, this research has employed several word forms, which can be like write, and writing, and writes. There were also some words of families with analogous derivations and similar meanings. Stemming often refers to a technique that removes derivational affixes from words and chops off their ends in the hopes of attaining the goal most of the time.

Lemmatization and stemming techniques are utilized to decrease the number of words that convey comparable sentimental connotations and to convert words from their complete form to their truncated parent form. Lemmatization is a method for reducing words to their lemma using a linguistic dictionary [23]. Linguistic accuracy preservation is a common practice that involves the utilization of language-specific lemmatization algorithms or rule-based methodologies.

The stemming algorithm produces the stem form of a given word. Non-STEM words are subjected to stemming and subsequently substituted with their corresponding stem words. For example, 'like', 'liken', 'likewise', 'liking', and 'likelihood' has been substituted with the term 'lik'. It helps to improve model performance, grouping similar words, easier to analyze and understand [24].

3.3.3. Mention with Hyperlinks

In the context of sentiment analysis of textual data, it is frequently imperative to perform text cleaning procedures, which involve the elimination of hyperlinks, mentions, and punctuation marks. This procedure facilitates the extraction of the factual information conveyed by users, with a specific emphasis on the text pertinent to sentiment analysis. To determine the optimal sentiment combination, it is crucial to perform text cleaning procedures such as hyperlink removal [25].

On social media, people post videos and blogs on the internet and mention tag other users. As a result, many tweets contain hyperlinks and Twitter mentions. Examples cited by Twitter users include @switchfoot and @Kenichan.

Examples of hyperlinks <https://www.youtube.com>, <https://www.google.com>, we used a regular expression to remove hyperlinks and mentions from the text [26]. Since we only need the actual texts that users express from their minds for sentiment analysis, the second step of my text cleaning is to remove all mentions from tweets. In my code, "HTTPS?: S+" and "HTTP. I've used the regular expression "@S+" to remove mentions. I've used the word "[A-Za-z0-9]+" to remove punctuation.

3.3.4. Stop-Words

Stop-words are often used for 'in, the, I' etc. which have minimal meaning, and

also do not help decipher the spirit of the statement. By using some built-in design for the English language, such terms can be eliminated from the vocabulary [27].

Making a list is an additional strategy. Sorting the terms in decreasing order of frequency will do this. The vocabulary size can be reduced by eliminating stop words that don't contribute to the analysis. Stop words are frequently used terms in English that have no semantic context. As a result, we got rid of them before classifying them. In **Figure 7**, some stop words have been shown for our dataset.

3.4. Visualization of Data

After cleaning the data, we visualized it by dividing it into two separate word clouds, one for positive words shown in **Figure 8** and the other for negative words shown in **Figure 9**, based on the words that occurred most frequently in the dataset. Here are some positive words we have found, which are mostly used in the positive posts.

Here are some negative words we have found, which is mostly used in negative posts.

```

['i',
'me',
'my',
'myself',
'we',
'our',
'ours',
'ourselves',
'you',
"you're",
"you've",
"you'll",
"you'd",
'your',
'yours',
'yourself',
'yourselves',
'he',
'him',
'his',
'himself',
'she',
"she's",
'her',
'hers',
'herself',
'it',
"it's",
'its',
'itself',
'they',
'them',
'their',

'are',
'was',
'were',
'be',
'been',
'being',
'have',
'has',
'had',
'having',
'do',
'does',
'did',
'doing',
'a',
'an',
'the',
'and',
'but',
'if',
'or',
'because',
'as',
'until',
'while',
'of',
'at',
'by',
'for',
'with',
'about',
'against',
'between',

'during',
'before',
'after',
'above',
'below',
'to',
'from',
'up',
'down',
'in',
'out',
'on',
'off',
'over',
'under',
'again',
'further',
'then',
'once',
'here',
'there',
'when',
'where',
'why',
'how',
'all',
'any',
'both',
'each',
'few',
'more',
'most',

'same',
'so',
'than',
'too',
'very',
's',
't',
'can',
'will',
'just',
'don',
"don't",
'should',
"should've",
'now',
'd',
'll',
'm',
'o',
're',
've',
'y',
'ain',
'aren',
"aren't",
'couldn',
"couldn't",
'didn',
"didn't",
'doesn',
"doesn't",
'hadn',
"hadn't",

```

Figure 7. Example of Stop words of this study.

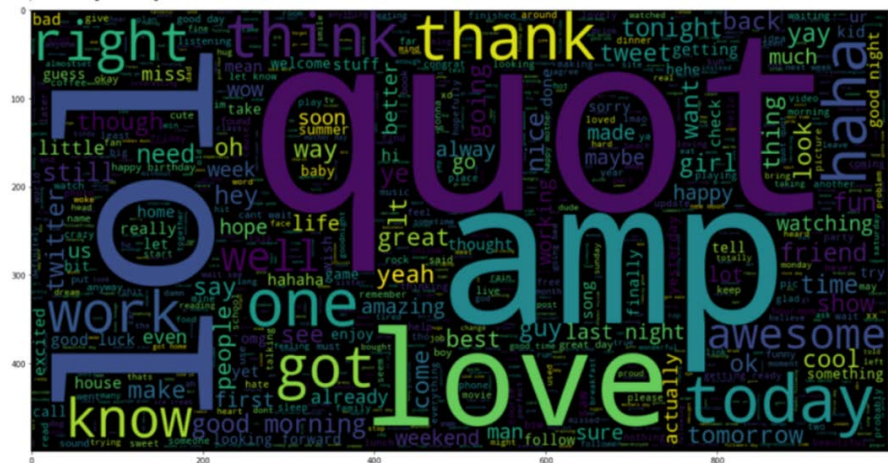


Figure 8. Positive words visualization.

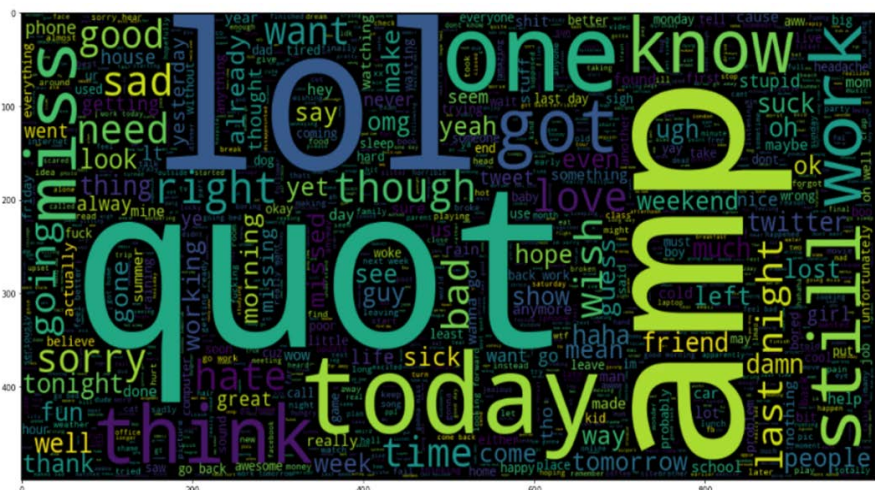


Figure 9. Negative words visualization.

After the pre-processing, we divided the train/test ratio into 80 percent and 20 percent. Therefore, the amount of the data for the train is 1.2 million, whereas the size of the data for the test is 320 thousand.

Tokenization

Tokenization is the process of breaking down a character's sequence into smaller sections known as tokens, as well as occasionally eliminating some characters such as punctuation. The example of tokenization [28] is shown in Figure 10.

Tokenizer generates tokens for each word in the data corpus and then maps them to an index using a dictionary [29]. The index for each word is included in the word index. Vocabulary size represents the total number of words in the data corpus. Following that, the vocabulary size increased to 290,575. We already have a tokenizer object, which can convert any word into a dictionary key (number). We must provide it with a set of numbers. Additionally, we must make sure that the input shapes of the sequences are uniform. They ought to all be the same length. However, tweets' messages vary in their word counts. We

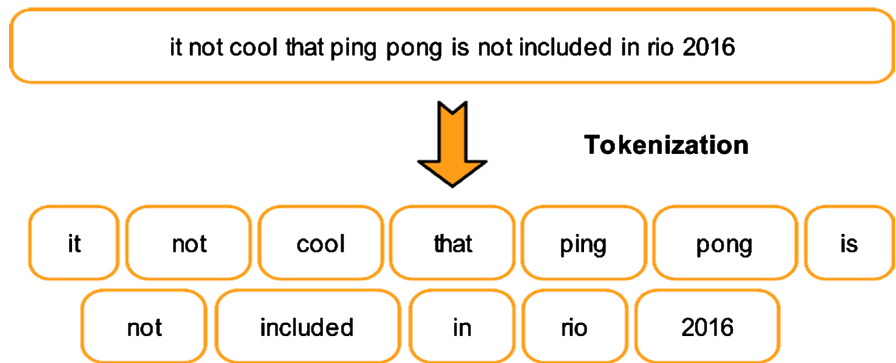


Figure 10. Tokenization example of this research.

asked Pad Sequence for a little assistance to complete my work to prevent this. The MAX SEQUENCE LENGTH fixed length will be used for the whole sequence. We are developing the encoded class prediction model (0 or 1 as this is a binary classification). As a result, we turned my training labels into encodings.

3.5. Feature Extraction Using Word Embedding

We use the Word Embedding technique to represent words as variables in text processing. This helps us to analyze patterns and context nuances in the language to gain new understanding. Embedding is a popular way to define a word in a document. This allows us to understand context, semantics, syntactic similarities, and word relationships in a document. We could relearn the pattern by using Word Embedding in our NLP application, but this would be time-consuming and inefficient. So, we use pre-built Word Embedding, such as GloVe or Word2Vec, to give extra meaning to words. These pre-models can be very useful for tasks such as classification. In our specific case, we use Glove Embedding from Stanford AI to optimize and contextualize text. That way we can improve our NLP model without reinventing the wheel.

3.6. Model Training-Long Short-Term Memory (LSTM)

The word cloud reveals that some terms appear often in both good and negative tweets. This might be a problem when employing training data such as Naive Bayes, SVD, or others. As a result, we use sequence models.

Sequence Model

In this research, we have implemented a sequence model shown in **Figure 11**. Recurrent neural networks can analyze data sequences and learn input sequence patterns to create output that can be a video sequence or a scalar value.

In our instance, the neural network predicts a scalar value. Here is the model architecture we have used:

- **Embedding Layer:** For every input we generate Embedding Vector sequence is what we use for model architecture. Conv1D Layer: It is used to divide large feature vectors of data into smaller ones.
- **Long Short-Term Memory (LSTM):** An RNN variant [30] includes a memory

```

Model: "model"
-----
Layer (type)                Output Shape                Param #
-----
input_1 (InputLayer)        [(None, 30)]                0
embedding (Embedding)       (None, 30, 300)            87172500
spatial_dropout1d (SpatialD  (None, 30, 300)            0
ropout1D)
conv1d (Conv1D)             (None, 26, 64)             96064
bidirectional (Bidirectiona  (None, 128)                 66048
l)
dense (Dense)               (None, 512)                 66048
dropout (Dropout)          (None, 512)                 0
dense_1 (Dense)            (None, 512)                 262656
dense_2 (Dense)            (None, 1)                   513
-----
Total params: 87,663,829
Trainable params: 491,329
Non-trainable params: 87,172,500

```

Figure 11. Sequence model.

state cell that could be used to remember the context of letters farther into the text in the ability to talk contextual meaning instead of always close words as in RNN [31].

- Dense: Classification using dense connected fully layers.

This research used Adam optimization for Gradient Descent. Callbacks are special functions that are triggered after each epoch to accomplish certain tasks [32]. Two callbacks were used in this scenario. To improve the results, the LR-Scheduler callback alters the learning rate at a specified epoch. In this notebook, the learning rate remained constant for the first ten epochs before falling exponentially. Model Checkpoint saves the best model during training depending on specific metrics. The model is preserved with the lowest validation loss in this situation. The model generates prediction scores ranging from 0 to 1, which are then classified into two classes based on a threshold value. If the score is greater than 0.5, it is classified as positive, and this serves as the categorization threshold.

3.7. Lexicon Approach

The Lexicon-based approach [33] is a widespread sentiment analysis technique that depends on the existence of specific words or phrases in a document to infer sentiment. This method employs a sentiment lexicon, which is a pre-defined

dictionary comprising words or attributes along with their associated sentiment values. Based on this lexicon, each term in the paper is correlated with sentimental polarity and conviction. To compute the overall polarity score of a document, the lexicon is matched against the document, and the occurrence of positive and negative words is calculated. The sentiment orientation of the document is then assigned using the following formulas [34]:

$$x_1(s) = 1 \text{ if } \text{sum}(l, s) > 0 \quad (1)$$

$$x_1(s) = 0 \text{ if } \text{sum}(l, s) = 0 \quad (2)$$

$$x_1(s) = -1 \text{ if } \text{sum}(l, s) < 0 \quad (3)$$

Here, $\text{sum}(l, s)$ represents the difference between the total number of positive and negative sentiment words discovered in the document. A positive sum implies a favorable sentiment, whereas a negative sum suggests a negative sentiment. However, this strategy has limits. While lexical algorithms can produce practically faultless results, they are dependent on the availability of a lexicon, which may not be available in all languages. Furthermore, the Lexicon-based approach may struggle with sarcasm, context-dependent mood, and language nuances.

To generate sentiment lexicons, researchers use a variety of methodologies, including manual compilation, lexical methods, and corpus-based approaches. The lexicon can be enlarged by including synonyms, WordNet, SenticNet, or utilizing emoticons and emojis as sentiment indicators. In summary, the Lexicon-based approach to sentiment analysis uses sentiment lexicons to match words and phrases in a document, allowing for the computation of an overall polarity score. While it provides a simple and effective solution, it does have limitations in terms of language availability and contextual nuances. Researchers continue to improve sentiment lexicons to improve accuracy and capture sentiment in text more thoroughly.

A Python of package 2, package 3 called Text-Blob, is used for the process of textual data [35]. By using a straightforward API for getting started with some typical NLP activities, which can be a part-of-speech for tagging, which can be noun phrases for extraction, sentiment analysis for data, classification for the data, translation for the data, and other things. Text-Blob plays well with NLTK and pattern while standing on their enormous shoulders.

4. Experimental Result

Training and extrapolating results are fraught with errors. Complex models reduce training errors because the error rate decreases with complexity. Bias-variance decomposition (Bias + Variance) reduces the frequency of erroneous generalizations. "Overfitting" occurs when a drop in training error rates increases test error rates. Accuracy, accuracy, recall rate, and F_1 -Score can be used to evaluate categorization strategies. Writers used many approaches to evaluate their models. Most studies looked at many success markers, but some used just one. We assess the model accuracy, precision, recall, and F_1 -Score. This

four-factor structure works effectively for analyzing prediction data. The ability to notice and classify events is linked to accuracy. Equation (4) [36] gives the formula:

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{Total Number of Tuples}} \quad (4)$$

Precision is the degree to which observed events match predicted ones in statistics. Equation (5) [37] expresses precision mathematically. The formula is:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (5)$$

The recall of a classification model represents the fraction of relevant instances that the model correctly identifies. It is a measure of the model's ability to find all the positive samples in the dataset. Here is the formula (6) [38] for determining recall:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (6)$$

This technique provides a balance between precision and memorability, hence the term "harmonic mean". Formula (7) [39] can be used to calculate the F₁-Score.

$$F_1\text{-Score} = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

4.1. Performance Evaluation

Within this phase, we will assess our work performance, which will explain the precision of our work. A deep learning model (DL) and a lexicon-based approach were compared in terms of performance. Confusion matrices were constructed for both methods, with **Figure 12** displaying the deep learning model's confusion matrix and **Figure 13** indicating the lexicon-based approach's confusion matrix. The confusion matrix evaluates the models' accuracy and efficacy by demonstrating how well they predict distinct classes. True positives and true negatives are accurately classified cases, whereas false positives and false negatives represent misclassifications.

The performance evaluation also included a comparison of the classification reports from the deep learning (ML) model and the lexicon-based technique. The classification reports provide a full examination of the models' performance, including recall, precision, and F₁-Score for each class (negative and positive), as well as accuracy and other aggregated metrics.

4.2. Analysis of the Classification Report

The result analysis included comparing the classification reports from the deep learning (ML) model and the lexicon-based technique. The classification reports provide a full examination of the models' performance, including precision, recall, and F₁-Score for each class (negative and positive), as well as accuracy and

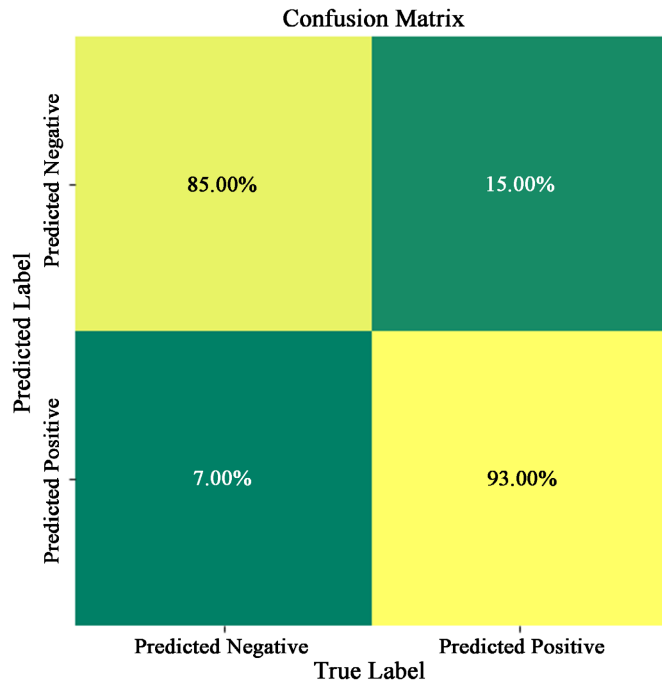


Figure 12. Confusion matrix of deep learning model.

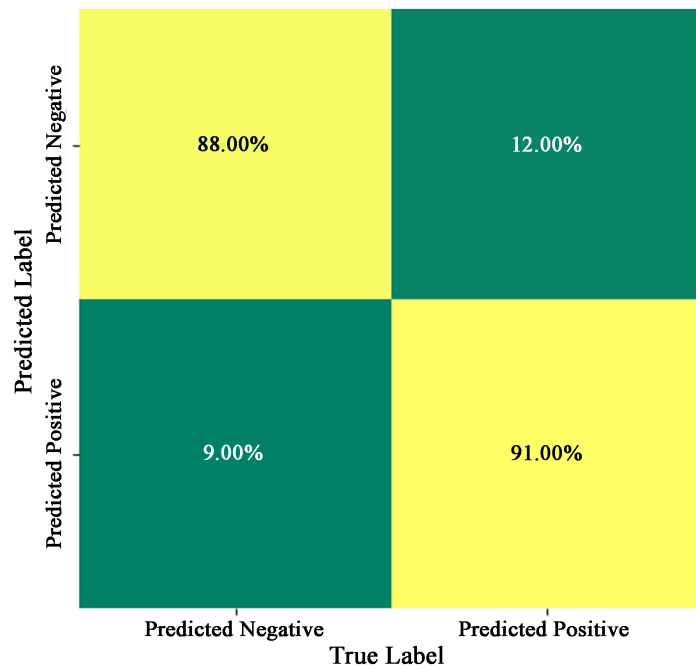


Figure 13. Confusion matrix of lexicon-based technique.

other aggregated metrics.

Table 2 shows the categorization report for the deep learning model (DL). It gives an overview of the model’s precision, recall, and F₁-Score performance for both the negative and positive classes.

The total accuracy of the deep learning model is stated to be 0.98. The macro average for both courses is 0.86, recall is 0.96, and F₁-Score is 0.87. The weighted

Table 2. Classification report of deep learning model.

	Precision	Recall	F ₁ -Score
Negative	0.85	0.93	0.88
Positive	0.88	0.9	0.82
Accuracy			0.98
Macro avg	0.86	0.96	0.87
Weighted avg	0.84	0.98	0.86

Table 3. Classification report of lexicon-based model.

	Precision	Recall	F ₁ -Score
Negative	0.90	0.92	0.85
Positive	0.93	0.94	0.8
Accuracy			0.95
Macro avg	0.84	0.91	0.88
Weighted avg	0.82	0.9	0.84

average has a precision of 0.84, a recall of 0.98, and an F₁-Score of 0.86.

Table 3 shows the categorization report for the lexicon-based technique. Precision, recall, and F₁-Score for negative and positive classes are similar to the deep learning model's classification report.

The total accuracy of the deep learning model is stated to be 0.95. Furthermore, the macro average (across both classes) shows an average precision of 0.84, recall of 0.91, and F₁-Score of 0.88. The weighted average accounts for class imbalance and yields precision of 0.82, recall of 0.90, and F₁-Score of 0.84.

Comparing the deep learning model with lexicon-based approach's classification reports indicates their performance. The deep learning model had 0.85 precision, while the lexicon-based technique had 0.90. However, the deep learning model had stronger recall (0.93) than the lexicon-based approach (0.92). The deep learning model had a higher F₁-Score (0.88) in the negative class than the lexicon-based strategy (0.85). Precision (0.93 vs. 0.88) and recall (0.94 vs. 0.90) were better for the positive class using the lexicon-based strategy. In the positive class, the deep learning (0.82) outperformed the lexicon-based technique (0.80). The deep learning model outperformed the deep learning model with 0.98 accuracy. Also, the deep learning model performed better across both classes in precision, recall, and F₁-Score criteria.

5. Discussion

The discussion section seeks to go deeper into the sentiment analysis research using lexicon-based and machine-learning approaches, with a particular focus on Twitter data. The findings will be thoroughly analyzed in this section, along with the study's significance, the results' ramifications, and prospective future

research directions. The sentiment140 Twitter dataset was utilized in the study to compare two widely used approaches: the lexicon-based method and the deep learning method. With more than 1.6 million data points in the dataset, there is a significant amount of data for analysis and inference.

The study's findings showed that the Long Short-Term Memory (LSTM) model, used especially in the deep learning approach, produced an amazing accuracy score of almost 98%. On the other hand, the lexicon-based strategy produced an accuracy score that was 95%. These results demonstrate the superiority of the deep learning strategy over the lexicon-based approach in sentiment analysis tasks, demonstrating its capacity to collect and understand sentiment accurately. The lexicon-based method's intrinsic constraints can be blamed for the considerable accuracy gap between the two approaches. Lexicons are dictionaries or databases that include word or phrase sentiment scores that have already been determined. Although lexicons are a useful tool for sentiment analysis, they frequently miss the subtleties and context-specific meanings of words, producing less accurate results. Deep learning models, like LSTM, on the other hand, have the advantage of learning directly from the data and identifying intricate patterns and connections in the text. With more training data and this ability, they can adapt to various situations and gain accuracy over time.

The results of this study have several ramifications for sentiment analysis. They emphasize the necessity of using deep learning techniques, especially deep learning models like LSTM, for more precise sentiment classification. These algorithms are capable of accurately capturing the fine details and situation-specific moods expressed in social media messages. Second, the study emphasizes the importance of using expansive datasets for training and assessing sentiment analysis models, such as sentiment140. The sheer amount of data makes it possible to train robust models and conduct thorough evaluations, producing outcomes that are more trustworthy and generalizable.

The study also supports the idea that sentiment analysis on social media sites like Twitter is essential for determining the attitudes and sentiments of the general population towards certain issues. For companies, marketers, and decision-makers to make wise choices, comprehend client preferences, and raise customer happiness, accurate sentiment research can offer useful information. Despite the noteworthy results, this study has certain restrictions. The attitudes expressed in other domains or on other social media platforms may not be fully reflective of the analysis because it was specifically focused on the sentiment140 dataset and Twitter data. Furthermore, the study mainly concentrated on categorizing sentiment as either positive or negative; however, future research may explore more nuanced sentiment analysis.

Future studies should look into the possibility of merging lexicon-based and machine-learning methodologies to maximize the benefits of each approach. Furthermore, investigating additional robust deep learning architectures and combining domain-specific expertise could improve the precision and efficacy of

sentiment analysis models.

6. Conclusion

We emphasized the lexicon-based technique and deep learning methods in this research. Our main contribution is to the study of popular sentiment-related posts from Twitter. We have explored many topics that are mostly very usable and that people want to discuss with each other. People want to share their opinions on some of the specific topics. Mostly they spend very busy time in their life but there are some specific criteria because people usually love to enjoy spending time on social media. For this work, we need a huge amount of data which might help us to get the desired accuracy rate. For that, we started to find the dataset from Kaggle. Likely we found a dataset made during the COVID-19 situation when most people have used social media a lot. This dataset carries almost 1.5 million data. So, this was the actual key point for our work. We collect the dataset which already has been in labeled condition. After that, we ran the clean process and made it ready to apply the model and the different approaches. For applying the model, we needed to split the dataset into train and test data. We split the dataset into 0.8 and 0.2 percent. To train our model we have used the train part data. After that, we used the test data to measure the output accuracy for the deep learning model. On the other hand, we have used the same test part data for measuring the output for the lexicon approach. Finally, we have compared the two different parts' outputs. We have analyzed many related works and discovered that many people currently using the lexicon approach for sentiment analysis works. But based on our work we can find out that the deep learning approach can give a better accuracy for the sentiment analysis work. So, if the researcher works on a deep learning approach with various models, then with sentiment analysis, he will be able to get a better accuracy than other approaches. For this reason, our work will get better responses as proof of a better accuracy method.

Author Contributions

Conceptualization, T.I., M.A.S, M.S.T and M.M.R.F; methodology, T.I., M.R.S. and M.M.R.F; visualization and data collection, M.S.T., M.A.S.; validation, T.I. and M.A.S; investigation, M.R.S. and M.M.R.F; resources, M.A.S; data curation, T.I. and M.B; writing—original draft preparation, T.I, M.M.R.F and M.A.S; editing, M.S.T, T.I and M.A.S.; supervision, J.F and M.B.; project administration, T.I. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Kwon, H.J., Ban, H.J., Jun, J.K. and Kim, H.S. (2021) Topic Modeling and Senti-

- ment Analysis of Online Review for Airlines. *Information*, **12**, Article No. 78. <https://doi.org/10.3390/info12020078>
- [2] Catelli, R., Pelosi, S. and Esposito, M. (2022) Lexicon-Based vs. Bert-Based Sentiment Analysis: A Comparative Study in Italian. *Electronics*, **11**, Article No. 374. <https://doi.org/10.3390/electronics11030374>
- [3] Rahman, M.H., Islam, T., Rana, M.M., Tasnim, R., Mona, T.R. and Sakib, M.M. (2023) Machine Learning Approach on Multiclass Classification of Internet Firewall Log Files. *Proceedings of International Conference on Computational Intelligence and Sustainable Engineering Solution, CISES 2023*, Greater Noida, 28-30 April 2023, 358-364. <https://doi.org/10.1109/CISES58720.2023.10183601>
- [4] Islam, M.T., Ahmed, T., Raihanur Rashid, A.B.M., Islam, T., Rahman, M.S. and Tarek Habib, M. (2022) Convolutional Neural Network Based Partial Face Detection. *2022 IEEE 7th International Conference for Convergence in Technology, I2CT*, Mumbai, 7-9 April 2022, 1-6. <https://doi.org/10.1109/I2CT54291.2022.9825259>
- [5] Khan, H.U., Nasir, S., Nasim, K., Shabbir, D. and Mahmood, A. (2021) Twitter Trends: A Ranking Algorithm Analysis on Real Time Data. *Expert Systems with Applications*, **164**, Article ID: 113990. <https://doi.org/10.1016/j.eswa.2020.113990>
- [6] Birjali, M., Kasri, M. and Beni-Hssane, A. (2021) A Comprehensive Survey on Sentiment Analysis: Approaches, Challenges and Trends. *Knowledge-Based Systems*, **226**, Article ID: 107134. <https://doi.org/10.1016/j.knosys.2021.107134>
- [7] Bhowmik, N.R., Arifuzzaman, M. and Mondal, M.R.H. (2022) Sentiment Analysis on Bangla Text Using Extended Lexicon Dictionary and Deep Learning Algorithms. *Array*, **13**, Article ID: 100123. <https://doi.org/10.1016/j.array.2021.100123>
- [8] Talukder, M.S.H., Bin Sulaiman, R., Chowdhury, M.R., Nipun, M.S. and Islam, T. (2023) PotatoPestNet: A CTInceptionV3-RS-Based Neural Network for Accurate Identification of Potato Pests. *Smart Agricultural Technology*, **5**, Article ID: 100297. <https://doi.org/10.1016/j.atech.2023.100297>
- [9] Tahosin, M.S., Sheakh, M.A., Islam, T., Lima, R.J. and Begum, M. (2023) Optimizing Brain Tumor Classification through Feature Selection and Hyperparameter Tuning in Machine Learning Models. *Informatics in Medicine Unlocked*, **43**, Article ID: 101414. <https://doi.org/10.1016/j.imu.2023.101414>
- [10] Gulati, K., Saravana Kumar, S., Sarath Kumar Boddu, R., Sarvakar, K., Kumar Sharma, D. and Nomani, M.Z.M. (2022) Comparative Analysis of Machine Learning-Based Classification Models Using Sentiment Classification of Tweets Related to COVID-19 Pandemic. *Materials Today: Proceedings*, **51**, 38-41. <https://doi.org/10.1016/j.matpr.2021.04.364>
- [11] Sani, M., Ahmad, A. and Abdulazeez, H.S. (2022) Sentiment Analysis of Hausa Language Tweet Using Machine Learning Approach. *Journal of Research in Applied Mathematics*, **8**, 7-16.
- [12] ur Rehman, M. and Bashir, M. (2023) Sentiment Analysis on Disputed Territory Discrepancies Using Machine Learning-Based Text Mining Approach. *VFAST Transactions on Software Engineering*, **11**, 17-25.
- [13] Imanina Zabha, N., Ayop, Z., Anawar, S., Hamid, E. and Zainal Abidin, Z. (2019) Developing Cross-Lingual Sentiment Analysis of Malay Twitter Data Using Lexicon-Based Approach. *International Journal of Advanced Computer Science and Applications*, **10**, 346-351. <http://www.ijacsa.thesai.org>
<https://doi.org/10.14569/IJACSA.2019.0100146>
- [14] Mitra, A. (2020) Sentiment Analysis Using Machine Learning Approaches (Lexicon Based on Movie Review Dataset). *Journal of Ubiquitous Computing and Commu-*

- nication Technologies, **2**, 145-152. <https://doi.org/10.36548/jucct.2020.3.004>
- [15] Hajrahimova, M.S. and Ismaylova, M.I. (2021) Machine Learning-Based Sentiment Analysis of Twitter Data.
- [16] Sham, N.M. and Mohamed, A. (2022) Climate Change Sentiment Analysis Using Lexicon, Machine Learning and Hybrid Approaches. *Sustainability*, **14**, Article No. 4723. <https://doi.org/10.3390/su14084723>
- [17] Ainapure, B.S., et al. (2023) Sentiment Analysis of COVID-19 Tweets Using Deep Learning and Lexicon-Based Approaches. *Sustainability*, **15**, Article No. 2573. <https://doi.org/10.3390/su15032573>
- [18] Srivastava, R., Bharti, P.K. and Verma, P. (2022) Comparative Analysis of Lexicon and Machine Learning Approach for Sentiment Analysis. *International Journal of Advanced Computer Science and Applications*, **13**, 71-77. <https://doi.org/10.14569/IJACSA.2022.0130312>
- [19] Zvonarev, A. (2019) A Comparison of Machine Learning Methods of Sentiment Analysis Based on Russian Language Twitter Data.
- [20] Braig, N., Benz, A., Voth, S., Breitenbach, J. and Buettner, R. (2023) Machine Learning Techniques for Sentiment Analysis of COVID-19-Related Twitter Data. *IEEE Access*, **11**, 14778-14803. <https://doi.org/10.1109/ACCESS.2023.3242234>
- [21] Go, A., Bhayani, R. and Huang, L. (2009) Twitter Sentiment Classification Using Distant Supervision. <https://www-cs-faculty.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf>
- [22] Islam, T., et al. (2023) Review Analysis of Ride-Sharing Applications Using Machine Learning Approaches: Bangladesh Perspective. In: Harjule, P., Rahman, A., Agarwal, B. and Tiwari, V., Eds., *Computational Statistical Methodologies and Modeling for Artificial Intelligence*, CRC Press, Boca Raton, 99-122. <https://doi.org/10.1201/9781003253051-7>
- [23] Shaukat, S., Asad, M. and Akram, A. (2023) Developing an Urdu Lemmatizer Using a Dictionary-Based Lookup Approach. *Applied Sciences*, **13**, Article No. 5103. <https://doi.org/10.3390/app13085103>
- [24] Almuzaini, H.A. and Azmi, A.M. (2020) Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization. *IEEE Access*, **8**, 127913-127928. <https://doi.org/10.1109/ACCESS.2020.3009217>
- [25] HaCohen-Kerner, Y., Miller, D. and Yigal, Y. (2020) The Influence of Preprocessing on Text Classification Using a Bag-of-Words Representation. *PLOS ONE*, **15**, e0232525. <https://doi.org/10.1371/journal.pone.0232525>
- [26] Sharif, O., Hasan, M.Z. and Rahman, A. (2022) Determining an Effective Short-Term COVID-19 Prediction Model in ASEAN Countries. *Scientific Reports*, **12**, Article No. 5083. <https://doi.org/10.1038/s41598-022-08486-5>
- [27] Bezdan, T., et al. (2021) Hybrid Fruit-Fly Optimization Algorithm with K-Means for Text Document Clustering. *Mathematics*, **9**, Article No. 1929. <https://doi.org/10.3390/math9161929>
- [28] Cagri, T., Halit, Y., Furkan, Ş. and Oguzhan, O. (2023) Impact of Tokenization on Language Models: An Analysis for Turkish. *ACM Transactions on Asian and Low-Resource Language Information Processing*, **22**, Article No. 116. <https://doi.org/10.1145/3578707>
- [29] Sharif, O., et al. (2022) Analyzing the Impact of Demographic Variables on Spreading and Forecasting COVID-19. *Journal of Healthcare Informatics Research*, **6**, 72-90.

- <https://doi.org/10.1007/s41666-021-00105-8>
- [30] Islam, T., Kundu, A., Ahmed, T. and Khan, N.I. (2022) Analysis of Arrhythmia Classification on ECG Dataset. 2022 *IEEE 7th International conference for Convergence in Technology, I2CT*, Mumbai, 7-9 April 2022, 1-6.
<https://doi.org/10.1109/I2CT54291.2022.9825052>
- [31] Fischer, F., Birk, A., Somers, P., Frenner, K., Tarín, C. and Herkommer, A. (2022) FeaSel-Net: A Recursive Feature Selection Callback in Neural Networks. *Machine Learning and Knowledge Extraction*, **4**, 968-993.
<https://doi.org/10.3390/make4040049>
- [32] Song, C., Wang, X.K., Cheng, P.F., Wang, J.Q. and Li, L. (2020) SACPC: A Framework Based on Probabilistic Linguistic Terms for Short Text Sentiment Analysis. *Knowledge-Based Systems*, **194**, Article ID: 105572.
<https://doi.org/10.1016/j.knosys.2020.105572>
- [33] Li, H., Chen, Q., Zhong, Z., Gong, R. and Han, G. (2022) E-Word of Mouth Sentiment Analysis for User Behavior Studies. *Information Processing & Management*, **59**, Article ID: 102784. <https://doi.org/10.1016/j.ipm.2021.102784>
- [34] Hota, H.S., Sharma, D.K. and Verma, N. (2021) Lexicon-Based Sentiment Analysis Using Twitter Data: A Case of COVID-19 Outbreak in India and Abroad. In: Kose, U., et al., Eds., *Data Science for COVID-19*, Elsevier, Amsterdam, 275-295.
<https://doi.org/10.1016/B978-0-12-824536-1.00015-0>
- [35] Abiola, O., Abayomi-Alli, A., Tale, O.A., Misra, S. and Abayomi-Alli, O. (2023) Sentiment Analysis of COVID-19 Tweets from Selected Hashtags in Nigeria Using VADER and Text Blob Analyser. *Journal of Electrical Systems and Information Technology*, **10**, Article No. 5. <https://doi.org/10.1186/s43067-023-00070-9>
- [36] Islam, T., Kundu, A., Islam Khan, N., Chandra Bonik, C., Akter, F. and Jihadul Islam, M. (2022) Machine Learning Approaches to Predict Breast Cancer: Bangladesh Perspective. *Smart Innovation, Systems and Technologies*, **302**, 291-305.
https://doi.org/10.1007/978-981-19-2541-2_23
- [37] Islam, T., Hosen, M.A., Mony, A., Hasan, M.T., Jahan, I. and Kundu, A. (2022) A Proposed Bi-LSTM Method to Fake News Detection. 2022 *International Conference for Advancement in Technology*, Goa, 21-22 January 2022, 1-5.
<https://doi.org/10.1109/ICONAT53423.2022.9725937>
- [38] Sheakh, M.A., Sazia Tahosin, M., Hasan, M.M., Islam, T., Islam, O. and Rana, M.M. (2023) Child and Maternal Mortality Risk Factor Analysis Using Machine Learning Approaches. *ISDFS 2023—11th International Symposium on Digital Forensics and Security*, Chattanooga, 11-12 May 2023, 1-6.
<https://doi.org/10.1109/ISDFS58141.2023.10131826>
- [39] Hasan, M., Tahosin, M.S., Farjana, A., Sheakh, M.A. and Hasan, M.M. (2023) A Harmful Disorder: Predictive and Comparative Analysis for Fetal Anemia Disease by Using Different Machine Learning Approaches. *ISDFS 2023—11th International Symposium on Digital Forensics and Security*, Chattanooga, 11-12 May 2023, 1-6.
<https://doi.org/10.1109/ISDFS58141.2023.10131838>