The Scientific Temper (2024) Vol. 15 (spl): 114-120
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.14
https://scientifictemper.com/

RESEARCH ARTICLE

An ensemble-based approach for sentiment analysis of covid-19 Twitter data using machine learning and deep learning techniques

M. Jayakandan*, A. Chandrabose

Abstract
In the wake of the COVID-19 pandemic, social media platforms like Twitter have become critical channels for public expression, capturing a wide array of sentiments ranging from fear and anxiety to hope and optimism. This paper proposes an ensemble approach for automatic sentiment analysis of COVID-19-related tweets to extract valuable insights from large-scale data. The proposed method integrates multiple machine learning algorithms, including support vector machines (SVM), random forests, and deep learning models such as long short-term memory (LSTM) networks. By leveraging these diverse techniques, the ensemble model aims to improve classification accuracy and robustness in detecting positive, negative, and neutral sentiments. Feature extraction is optimized through natural language processing (NLP) techniques like term frequency-inverse document frequency (TF-IDF) and word embeddings. Experimental results on a publicly available COVID-19 Twitter dataset demonstrate the effectiveness of the proposed approach, showcasing its potential to contribute to public health monitoring, policy making, and understanding of public reactions during crises.

Keywords: Sentiment analysis, Natural language processing, Machine learning, Feature extraction, LSTM, TF-IDF.

Introduction

Sentiment analysis, or opinion mining, involves the computational study of people's emotions, attitudes, and opinions expressed in text. This field has seen significant growth in recent years due to the increase in online communication channels where individuals express their opinions, such as social media platforms, blogs, and forums. Twitter, in particular, has become one of the most popular platforms for sentiment analysis because of its wide user base and the large volume of real-time information it provides. Analyzing sentiment on Twitter during critical events can offer insights into public reactions, trends, and emotional responses on a global scale, Wang, Y., Guo, J., Yuan, C., & Li, B. (2022), Biradar, S. H., Gorabal, J. V., & Gupta, G. (2022).

The COVID-19 pandemic has brought unprecedented challenges, impacting virtually all aspects of daily life and societal structures. The sheer scale and intensity of the crisis spurred significant online discussions, with users from all walks of life taking to social media to share updates, fears, hopes, opinions, and frustrations. Twitter, with its dynamic and fast-paced nature, became a significant medium for disseminating information, government announcements, personal stories, and emotional responses. Consequently, analyzing COVID-19-related sentiment on Twitter presents a unique opportunity to understand how the pandemic affected people's emotional states and opinions over time, reflecting the pulse of society during this global crisis, Dangi, D., Dixit, D. K., & Bhagat, A. (2022), Leelawat, N., Jariyapongpaiboon, S., Promjun, A., Boonyarak, S., Saengtabtim, K., Laosunthara, A., ... & Tang, J. (2022).

Sentiment analysis on the COVID-19 Twitter dataset involves several technical challenges. Firstly, social media text often lacks structure, includes slang, abbreviations, and colloquial language, and is limited by Twitter's character limit of 280 characters. These factors require robust preprocessing techniques and advanced natural language processing (NLP)

Edayathangudy G.S Pillay Arts and Science College (Autonomous) (Affiliated to Bharathidasan University, Tiruchirappalli), Nagapattinam, Tamilnadu, India.

***Corresponding Author:** M. Jayakandan, Edayathangudy G.S Pillay Arts and Science College (Autonomous) (Affiliated to Bharathidasan University, Tiruchirappalli), Nagapattinam, Tamilnadu, India., E-Mail: dr.jayakandan@gmail.com

How to cite this article: Jayakandan, M., Chandrabose, A. (2024). An ensemble-based approach for sentiment analysis of covid-19 Twitter data using machine learning and deep learning techniques. The Scientific Temper, **15**(spl):114-120.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.14

Source of support: Nil

Conflict of interest: None.

Received: 18/09/2024 **Accepted:** 26/09/2024 **Published**: 16/10/2024

algorithms to accurately interpret the text. Additionally, COVID-19-specific terminology and emerging topics, such as new variants, lockdown policies, vaccines, and economic impacts, require models that can adapt to rapidly changing linguistic patterns. Techniques such as term frequency-inverse document frequency (TF-IDF), word embeddings, and deep learning-based representations (e.g., Word2Vec, GloVe, and BERT) have proven valuable in capturing these nuances, enabling more accurate sentiment classification, Kanakaraddi, S. G., Chikaraddi, A. K., Aivalli, N., Maniyar, J., & Singh, N. (2022, March), Müller, M., Salathé, M., & Kummervold, P. E. (2023) , Hall, K., Chang, V., & Jayne, C. (2022).

The dataset for COVID-19 Twitter sentiment analysis can be divided into different periods or themes, such as the initial outbreak, lockdown phases, vaccine rollout, and economic reopening. Tracking sentiments over these phases can highlight shifts in public attitudes and provide valuable insights for policymakers, healthcare providers, and researchers. The models developed for sentiment analysis on this dataset can classify tweets into categories such as positive, negative, neutral, and sometimes mixed sentiments. Advanced machine learning models and ensemble techniques are frequently employed for this purpose, combining the strengths of different classifiers to improve the overall accuracy of predictions, Yin, H., Song, X., Yang, S., & Li, J. (2022), Badi, H., Badi, I., El Moutaouakil, K., Khamjane, A., & Bahri, A. (2022), Alkhaldi, N. A., Asiri, Y., Mashraqi, A. M., Halawani, H. T., Abdel-Khalek, S., & Mansour, R. F. (2022, May).

The goal of sentiment analysis on COVID-19 Twitter data extends beyond merely understanding general emotions. By analyzing specific emotions—such as fear, anxiety, hope, and anger—across various demographics and geographic locations, researchers can uncover deeper trends and potential areas of concern. For example, a rise in anxiety-related tweets during vaccine distribution or anger-related tweets during lockdown enforcement periods could provide actionable insights for public health messaging. The analysis could also be used to track misinformation, as rapid sentiment changes may correlate with the spread of false information or rumors.

Background Study On Natural Language Processing

NLP is a subfield of artificial intelligence (AI) focused on the interaction between computers and human language. It combines linguistics, computer science, and AI to enable machines to understand, interpret, and generate human language in ways that are both meaningful and useful. Over the years, NLP has evolved substantially, with advancements in algorithms, computational power, and the availability of large datasets leading to remarkable breakthroughs in areas such as translation, sentiment analysis, and text generation, Fanni, S. C., Febi, M., Aghakhanyan, G., & Neri, E. (2023), Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023).

The initial stages of NLP, dating back to the 1950s, were marked by rule-based approaches. Early NLP systems relied heavily on predefined rules and manually crafted grammar structures to analyze language. Programs like the Georgetown-IBM experiment in 1954 demonstrated the potential of NLP by translating simple Russian sentences into English. Despite the progress, these systems had limited real-world applications due to the immense complexity of language and the limitations of computing power and algorithms at the time.

In the late 1980s and 1990s, the availability of large datasets and increased computational resources led to the emergence of statistical approaches in NLP. This period saw a shift from rigid rule-based systems to probabilistic and statistical models, which offered greater flexibility in handling language variability. Techniques like n-grams, Hidden Markov Models (HMM), and Maximum Entropy Models allowed NLP systems to better model language by estimating the likelihood of certain sequences of words, thus enabling more accurate predictions.

The introduction of machine learning further revolutionized NLP by allowing models to "learn" language patterns from large amounts of data rather than relying on hand-crafted rules. Supervised learning algorithms such as Naïve Bayes, support vector machines (SVM), and Decision Trees became popular for tasks like text classification, sentiment analysis, and named entity recognition. However, these early machine-learning approaches still faced limitations, especially in terms of understanding the context and meaning of words, Kochmar, E. (2022).

The advent of deep learning in the 2010s marked a transformative era for NLP, as neural networks demonstrated their capability to handle complex language tasks with high accuracy. RNNs, and later LSTM networks, became instrumental in processing sequential data like text. These architectures allowed models to better understand the sequential nature of language by "remembering" previous information, which was a significant advancement for tasks requiring context, such as machine translation and sentiment analysis, Durga, P., & Godavarthi, D. (2023), Hindarto, D. (2023).

However, RNNs and LSTMs also had limitations, particularly with long-range dependencies in text, where earlier information in a sentence or document might be forgotten. This led to the development of the Transformer model by Vaswani et al. in 2017, which introduced the concept of self-attention mechanisms. Transformers revolutionized NLP by allowing models to process entire sequences of text in parallel and capture dependencies across long spans. This architecture became the foundation for many state-of-the-art NLP models, such as BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pretrained Transformer), and T5 (Text-To-Text Transfer Transformer).

Frequency-Inverse Document Frequency (Tf-Idf) Based Feature Extraction

Frequency-Inverse Document Frequency (TF-IDF) is a widely-used feature extraction technique in Natural Language Processing (NLP) for evaluating the importance of words within a collection of documents, making it especially suitable for text-based applications like sentiment analysis. When applied to COVID-19 Twitter data, TF-IDF aids in highlighting terms that uniquely characterize the emotions, topics, or sentiments expressed within tweets related to the pandemic. This technique is instrumental in sentiment analysis as it filters out common words (e.g., "the," "is") while emphasizing terms that are more distinctive within the dataset, thus improving the accuracy and relevance of sentiment classification. TF-IDF is a composite metric derived from two main components, Dey, R. K., & Das, A. K. (2023), Addiga, A., & Bagui, S. (2022), Brindha, K., & Ramadevi, E. (2023):

Term Frequency (TF)

This measures the frequency of a term in a document, giving insight into how often a word appears in each individual document (in this case, a tweet). It is calculated as: Here, a higher term frequency indicates that a word is more prominent within a particular tweet.

$$TF(t,d) = \frac{Number\ of\ times\ term\ t\ appears\ in\ document\ d}{Total\ Number\ of\ terms\ in\ document\ d}$$

Inverse Document Frequency (IDF)

This assesses how unique or rare a term is across all documents. Words that appear frequently across many tweets (such as "COVID" or "pandemic" in a COVID-19 dataset) will have a low IDF score because they are less unique. It is calculated as:

$$IDF(t,D) = log \frac{Total\ number\ of\ documents\ in\ the\ corpus}{Number\ of\ documents\ where\ the\ term\ t\ appears}$$

TF-IDF Calculation

By multiplying the TF and IDF components, TF-IDF emphasizes terms that are frequent in a particular document but rare across the corpus, making them more informative for identifying distinct topics or sentiments. The TF-IDF value for a term ttt in document ddd is given by:

$$TF - IDF(t,d) = TF(t,d) \times IDF(t,D)$$

Applying TF-IDF to COVID-19 Twitter Sentiment Analysis

In sentiment analysis on COVID-19 Twitter data, TF-IDF-based feature extraction helps highlight keywords that capture public emotions and attitudes towards various aspects of the pandemic, such as government policies, vaccine distribution, or personal anxieties and experiences. Here's a breakdown of how TF-IDF is applied to this specific dataset:

Data Preprocessing

Before computing TF-IDF, the raw tweet text needs to be preprocessed to enhance the quality of the extracted features. This step typically includes:

- *Tokenization*

Splitting tweets into individual words or tokens.

- *Removing Stop Words*

Filtering out commonly used words (like «the,» «is,» «and») that do not contribute to the sentiment.

- *Stemming and Lemmatization*

Reducing words to their root forms (e.g., «vaccinating» becomes «vaccinate») to group related terms.

Calculating TF-IDF Scores

Once the text is preprocessed, the TF-IDF scores are computed for each term across all tweets. Terms with higher TF-IDF scores are considered more relevant for distinguishing the content or sentiment of individual tweets. For example, in the COVID-19 dataset, terms such as «lockdown,» «vaccine,» «positive,» or «fear» might have higher TF-IDF scores when appearing in tweets expressing specific sentiments.

Feature Vector Representation

Each tweet is represented as a vector of TF-IDF scores. These vectors become the input features for machine learning or deep learning models used in sentiment classification. The TF-IDF vectors capture the importance of words while retaining the contextual weightage, making the sentiment analysis model more responsive to words with distinctive sentiment-representative properties.

Step by Step Procedure for TF-IDF to COVID-19 Twitter Sentiment Analysis

Step 1: Data Collection

Collect the text data (e.g., tweets) related to COVID-19. This can be done using Twitter's API or other datasets publicly available, ensuring the data aligns with the sentiment analysis goal.

Step 2: Data Preprocessing

Preprocess the text data to prepare it for analysis. Preprocessing steps typically include:

- *Tokenization*

Break down each tweet into individual words or tokens.

- *Lowercasing*

Convert all text to lowercase to avoid treating «Vaccine» and «vaccine» as different terms.

- *Removing Stop Words*

Remove common words (like «the,» «is,» «and») that don't contribute much to the sentiment.

- *Removing Punctuation and Special Characters*

Clean up unnecessary punctuation, hashtags, URLs, and special characters that may not contribute to sentiment.

- *Stemming/Lemmatization*

Reduce words to their base forms, e.g., converting «running» to «run» or «studies» to «study.»

Step 3: Create Document-Term Matrix (DTM)

Construct a document-term matrix where each row represents a document (tweet) and each column represents a term (word). Each cell in this matrix contains the frequency count of each term within the corresponding document.

- In Python, tools like CountVectorizer from scikit-learn or TfidfVectorizer can be used to create the DTM.

Step 4: Compute Term Frequency (TF)

Calculate the Term Frequency for each term within each document. Term Frequency (TF) measures how often a term appears in a document relative to the total number of terms in that document:

Step 5: Compute Inverse Document Frequency (IDF)

Calculate the Inverse Document Frequency (IDF) for each term. IDF gives higher weight to terms that are rare across all documents and lower weight to common terms:

- In most implementations, a smoothing term (e.g., adding 1 to the denominator) is added to prevent division by zero when terms appear in all documents.

Step 6: Calculate TF-IDF Scores

Compute the TF-IDF score for each term in each document by multiplying the TF and IDF values: This assigns a score to each term in each document, indicating its importance within that document relative to the entire corpus.

Step 7: Construct the TF-IDF Matrix

Build the TF-IDF matrix, where each row represents a document and each column represents a term, with the values in the matrix corresponding to the TF-IDF scores. This matrix serves as the feature representation of the dataset, with each tweet transformed into a vector of TF-IDF values.

Step 8: Dimensionality Reduction (Optional)

If the TF-IDF matrix is high-dimensional (common in text datasets), apply dimensionality reduction techniques like:

- *Principal Component Analysis (PCA)*

Reduces dimensionality by projecting the data onto principal components.

Truncated Singular Value Decomposition (SVD)

Commonly used for reducing the dimensionality of TF-IDF matrices.

Long-Short Term Memory Based Classification Method

LSTM networks are a type of RNN designed to handle the challenges of learning and remembering long-term dependencies in sequential data, such as text. LSTM-based models are widely used for text classification tasks, including sentiment analysis, due to their ability to capture the context of words and phrases across sequences of text. For COVID-19 Twitter data, LSTM models can effectively capture the sentiments and evolving opinions of users by processing each tweet as a sequence of words.

Key Features of LSTMs

Cell State

The core component of an LSTM is its cell state, which serves as a long-term memory that carries information across sequences. The cell state can be modified by information flowing through various gates, allowing the model to maintain or forget information as needed.

Gates

LSTMs use three types of gates to control the flow of information:

- *Forget Gate*

Decides what information to discard from the cell state. It takes the previous hidden state and the current input, applying a sigmoid activation function to produce values between 0 and 1, where 0 indicates «forget this» and 1 indicates «keep this.»

- *Input Gate*

Determines what new information to store in the cell state. It consists of a sigmoid layer (to decide which values to update) and a tanh layer (to create new candidate values).

- *Output Gate*

Controls what part of the cell state will be outputted as the hidden state for the next time step. It uses the previous hidden state and the current input to decide what to output.

Handling Long Sequences

Unlike traditional RNNs, which struggle with long sequences due to gradient issues, LSTMs maintain their ability to learn from earlier time steps thanks to their cell state and gating mechanisms. This makes them effective for tasks where context and order matter, such as language modeling or sentiment analysis.

The COVID-19 pandemic has triggered significant public discourse on platforms like Twitter, where users express various sentiments regarding health policies, vaccine developments, and personal experiences. Accurately analyzing this sentiment is crucial for understanding public perception and informing policy decisions. This proposal outlines a hybrid approach combining term frequency-inverse document frequency (TF-IDF) for feature extraction and LSTM networks for classification and prediction of sentiment in tweets related to COVID-19.

- To classify sentiments expressed in COVID-19-related tweets into categories such as positive, negative, and neutral.

- To leverage the strengths of both TF-IDF and LSTM in processing textual data effectively.
- To provide insights into public sentiment trends during the pandemic.

Proposed Methodology

Step 1: Data Collection

- Collect a dataset of tweets related to COVID-19 using Twitter's API, focusing on a specific time frame or relevant hashtags (e.g., #COVID19, #pandemic, #vaccine).
- Ensure that the dataset contains labeled sentiments, either through manual annotation or using pre-existing sentiment labels.

Step 2: Data Preprocessing

- *Text Cleaning*

Remove URLs, special characters, hashtags, and punctuation. Convert all text to lowercase.

- *Tokenization*

Split each tweet into individual words or tokens.

- *Stop Word Removal*

Filter out common stop words that do not contribute to sentiment.

- *Stemming/Lemmatization*

Reduce words to their root forms to normalize the dataset.

Step 3: Feature Extraction using TF-IDF

- *Calculate TF-IDF Scores*

Create a document-term matrix (DTM) from the pre-processed tweets, where each row corresponds to a tweet, and each column corresponds to a term.

- *Feature Representation*

Transform the text data into a TF-IDF matrix, where each tweet is represented as a vector of TF-IDF scores.

Step 4: Splitting the Dataset

Split the TF-IDF matrix and sentiment labels into training, validation, and test sets (e.g., 70% training, 15% validation, 15% testing).

Step 5: Building the LSTM Model

- *Input Layer*

Specify the shape of the input data, typically the number of features in the TF-IDF matrix.

- *LSTM Layer*

Add one or more LSTM layers to capture temporal dependencies in the sequence of words.

- *Dense Layer*

Use a dense layer with a softmax activation function for multi-class classification (for example, positive, negative, neutral).

- *Dropout Layers*

Include dropout layers to prevent overfitting

Step 6: Model Training

- Train the LSTM model using the training set, monitoring performance on the validation set.
- Adjust hyperparameters, including the number of epochs, batch size, and learning rate.

Result And Discussion

The dataset used in this research work to evaluate the performance of the proposed TF-IDF with LSTM (https://www.kaggle.com/datasets/gpreda/covid19-tweets). The performance of the LSTM with TF-IDF approach is evaluated with other pre-processing techniques like Bag of Words, Word Embeddings, Count Vectorization, N-grams, and Sentiment Lexicons using various metrics like Accuracy, Precision, Recall, and F1-Score.

Table 1 depicts the classification accuracy (in %) obtained by the TF-IDF, Bag of Words, Word Embeddings, Count Vectorization, N-grams, and Sentiment Lexicons using LSTM, DBN, and ANN. From Table 1, TF-IDF achieved the highest accuracy (89.5%) when used with the LSTM model, indicating its effectiveness in capturing important features for sentiment analysis. The LSTM model consistently outperformed both DBN and ANN across most preprocessing techniques, showcasing its strength in handling sequential data. Word Embeddings also performed well, but they did not surpass the accuracy of TF-IDF with LSTM. The Bag of Words and Count Vectorization techniques exhibited the lowest accuracies, demonstrating their limitations in capturing meaningful relationships in the text data.

Table 2 depicts the precision (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN. From Table 2, TF-IDF achieved the highest precision (88.3%) when used with the LSTM model, demonstrating its capability to identify true positive sentiments effectively. The LSTM model generally outperformed both DBN and ANN across all preprocessing techniques, emphasizing its strength in accurately capturing relevant sentiment features.

Table 1: Classification accuracy (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN

Preprocessing technique	*Classification accuracy (in %)*		
	LSTM	*DBN*	*ANN*
Bag of Words	75.2	70.5	72.8
TF-IDF	89.5	83.1	80.2
Word Embeddings	82.4	77.6	79
Count Vectorization	73	68.2	70.5
N-grams	78.9	72.1	75.4
Sentiment Lexicons	76.5	71.8	73.9

Table 2: Precision (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN

Preprocessing Technique	*Precision (in %)*		
	LSTM	*DBN*	*ANN*
Bag of words	74.1	69.3	71.5
TF-IDF	88.3	81	79.7
Word embeddings	80.2	75.5	77.1
Count vectorization	72.5	66.8	69
N-grams	77.6	70.9	73.4
Sentiment Lexicons	75.8	70.2	72.5

Table 3: Recall (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN

Preprocessing Technique	*Recall (in %)*		
	LSTM	*DBN*	*ANN*
Bag of Words	72.8	68.1	70.4
TF-IDF	87	80.2	78.3
Word Embeddings	79	74	76.5
Count Vectorization	70	64.5	67
N-grams	76.5	69.2	72.1
Sentiment Lexicons	74.3	68.7	70.9

Word Embeddings also yielded high precision scores but did not surpass the performance of TF-IDF with LSTM. The bag of words and count vectorization methods showed lower precision scores, indicating their limited effectiveness in distinguishing between different sentiment classes.

Table 3 depicts the recall (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN. From Table 3, TF-IDF achieved the highest recall (87.0%) when utilized with the LSTM model, indicating its effectiveness in correctly identifying all relevant positive cases in the sentiment classification task. The LSTM model consistently outperformed both DBN and ANN across various pre-processing techniques, showcasing its robustness in capturing the nuances of sentiment. Word Embeddings also produced strong recall scores, though they did not exceed the performance of TF-IDF with LSTM. The bag of words and count vectorization techniques demonstrated the lowest recall, suggesting their limitations in accurately capturing all relevant sentiments.

Table 4 depicts the F1-score (in %) obtained by the TF-IDF, Bag of Words, Word Embeddings, Count Vectorization, N-grams, and Sentiment Lexicons using LSTM, DBN, and ANN. From Table 4, TF-IDF achieved the highest F1-Score (88.1%) when used with the LSTM model, demonstrating its effectiveness in balancing precision and recall for sentiment classification. The LSTM model consistently outperformed both DBN and ANN across all pre-processing techniques, reinforcing its robustness in sentiment analysis tasks. Word Embeddings also produced commendable F1-scores but did not surpass the performance of TF-IDF with LSTM. The Bag of Words and count vectorization techniques yielded the lowest F1-scores, highlighting their limitations in effectively capturing sentiment nuances.

Table 4: F1-score (in %) obtained by the TF-IDF, bag of words, word embeddings, count vectorization, N-grams, and sentiment Lexicons using LSTM, DBN, and ANN

Preprocessing Technique	*F1-Score (in %)*		
	LSTM	*DBN*	*ANN*
Bag of Words	73.5	68	70.6
TF-IDF	88.1	80.5	78.8
Word Embeddings	80.5	75.6	77.8
Count Vectorization	71.2	65.1	68.4
N-grams	77.2	70.1	72.7
Sentiment Lexicons	75	69.5	71.7

Conclusion

The results of our analysis highlight the effectiveness of the TF-IDF feature extraction method when combined with the LSTM model for sentiment classification on the COVID-19 Twitter dataset. The performance metrics demonstrated that TF-IDF outperformed other pre-processing techniques across several key indicators, including accuracy, precision, recall, F1-Score, and false positive rate. Specifically, the TF-IDF with LSTM approach achieved the highest accuracy (89.5%), precision (88.3%), recall (87.0%), and F1-score (88.1%), indicating its superior capability in effectively capturing the nuanced sentiments expressed in social media data.

The LSTM model's ability to handle sequential data, combined with TF-IDF's focus on important terms, allowed for a robust representation of text, enabling the model to discern meaningful patterns in sentiment more accurately. Other techniques, such as Word Embeddings, N-grams, and Bag of Words, although effective to some extent, did not provide the same level of performance as TF-IDF.

In conclusion, the combination of TF-IDF with LSTM not only enhances the accuracy of sentiment classification but also ensures a well-balanced evaluation of precision and recall, making it a highly effective approach for sentiment analysis in the context of rapidly evolving topics like COVID-19. This study reinforces the importance of selecting appropriate feature extraction methods and machine learning models to improve the performance of natural language processing tasks, particularly in analyzing real-time social media data. Future research could explore the integration of advanced techniques, such as Transformers, along with TF-IDF to further enhance classification outcomes and provide deeper insights into public sentiment.

References

Addiga, A., & Bagui, S. (2022). Sentiment analysis on Twitter data using term frequency-inverse document frequency. *Journal of Computer and Communications*, 10(8), 117-128.

Alkhaldi, N. A., Asiri, Y., Mashraqi, A. M., Halawani, H. T., Abdel-Khalek, S., & Mansour, R. F. (2022, May). Leveraging tweets for artificial intelligence driven sentiment analysis on the COVID-19 pandemic. *In Healthcare* (Vol. 10, No. 5, p. 910). MDPI.

Badi, H., Badi, I., El Moutaouakil, K., Khamjane, A., & Bahri, A. (2022). Sentiment analysis and prediction of polarity vaccines based on Twitter data using deep NLP techniques. *Radioelectronic and Computer Systems*, (4), 19-29.

Biradar, S. H., Gorabal, J. V., & Gupta, G. (2022). Machine learning tool for exploring sentiment analysis on twitter data. *Materials Today: Proceedings*, 56, 1927-1934.

Brindha, K., & Ramadevi, E. (2023). Twitter Sentiment Analysis For Feature Extraction Using Support Vector Machine (SVM) With TF-IDF. *Journal of Survey in Fisheries Sciences*, 3575-3583.

Dangi, D., Dixit, D. K., & Bhagat, A. (2022). Sentiment analysis of COVID-19 social media data through machine learning. *Multimedia tools and applications*, 81(29), 42261-42283.

Dey, R. K., & Das, A. K. (2023). Modified term frequency-inverse document frequency based deep hybrid framework for sentiment analysis. *Multimedia Tools and Applications*, 82(21), 32967-32990.

Durga, P., & Godavarthi, D. (2023). Deep-Sentiment: An Effective Deep Sentiment Analysis Using a Decision-Based Recurrent Neural Network (D-RNN). *IEEE Access*.

Fanni, S. C., Febi, M., Aghakhanyan, G., & Neri, E. (2023). Natural language processing. *In Introduction to Artificial Intelligence* (pp. 87-99). Cham: Springer International Publishing.

Hall, K., Chang, V., & Jayne, C. (2022). A review on Natural Language Processing Models for COVID-19 research. *Healthcare Analytics*, 2, 100078.

Hindarto, D. (2023). Comparison of RNN Architectures and Non-RNN Architectures in Sentiment Analysis. *Sinkron: jurnal dan penelitian teknik informatika*, 7(4), 2537-2546.

Kanakaraddi, S. G., Chikaraddi, A. K., Aivalli, N., Maniyar, J., & Singh, N. (2022, March). Sentiment analysis of covid-19 tweets using machine learning and natural language processing. *In Proceedings of Third International Conference on Intelligent Computing, Information and Control Systems: ICICCS 2021* (pp. 367-379). Singapore: Springer Nature Singapore.

Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: state of the art, current trends and challenges. *Multimedia tools and applications*, 82(3), 3713-3744.

Kochmar, E. (2022). Getting started with natural language processing. *Simon and Schuster*.

Leelawat, N., Jariyapongpaiboon, S., Promjun, A., Boonyarak, S., Saengtabtim, K., Laosunthara, A., ... & Tang, J. (2022). Twitter data sentiment analysis of tourism in Thailand during the COVID-19 pandemic using machine learning. *Heliyon*, 8(10).

Müller, M., Salathé, M., & Kummervold, P. E. (2023). Covid-twitter-bert: A natural language processing model to analyse covid-19 content on twitter. *Frontiers in artificial intelligence*, 6, 1023281.

Wang, Y., Guo, J., Yuan, C., & Li, B. (2022). Sentiment analysis of Twitter data. *Applied Sciences*, 12(22), 11775.

Yin, H., Song, X., Yang, S., & Li, J. (2022). Sentiment analysis and topic modeling for COVID-19 vaccine discussions. *World Wide Web*, 25(3), 1067-1083.