**International Journal of Data Informatics and Intelligent Computing (IJDIIC)**
Vol. 3, No. 2, June 2024, pp. 42~51
DOI: 10.59461/ijdiic.v3i2.115

# Sentiment Analysis and Topic Classification with LSTM Networks and TextRazor

**Jency Jose[1], Simritha R[1]**
[1]Department of Computer Science, Mount Carmel College, Bengaluru, India

**Article Info**

***Article history:***

Received April 05, 2024
Revised May 30, 2024
Accepted June 09, 2024

***Keywords:***

Sentiment analysis
LSTM
TextRazor
Twitter

**ABSTRACT**

In the ever-evolving landscape of social media, where user-generated content shapes digital discourse, the need for nuanced sentiment analysis and topic extraction is paramount. This paper presents a comprehensive approach utilizing advanced Natural Language Processing (NLP) techniques to enhance user experience and foster a healthier digital environment. Leveraging Long Short-Term Memory (LSTM) networks for sentiment analysis and TextRazor for topic extraction, the system provides insights into emotional tones and key themes within social media discussions. Through intuitive visualizations, users gain awareness of sentiment trends and topic distributions, empowering informed engagement. The results demonstrate high accuracy in sentiment classification with 86% and effective topic identification, contributing to the mitigation of misinformation and negativity online. This research underscores the potential of advanced NLP methods in cultivating constructive digital spaces and sets the stage for further innovation in the field.



**Corresponding Author:** Simritha R (e-mail: simritha14@gmail.com)

## 1. INTRODUCTION

In the realm of social media, a multifaceted digital landscape thrives on user-generated content and interactive features that foster engagement, interaction, and connectivity. Through platforms like Twitter, Facebook, and Instagram, individuals share their thoughts, opinions, and experiences, contributing to vibrant online communities. Despite its role in community-building and information dissemination, social media grapples with inherent challenges. On the one hand, it serves as a conduit for meaningful dialogue, enabling users to connect over shared interests, discuss current events, and express themselves creatively. On the other hand, it is susceptible to the spread of misinformation, the proliferation of hate speech, and the amplification of negativity. For businesses and organizations, social media has emerged as a pivotal component of marketing campaigns, offering unparalleled opportunities for brand promotion, customer engagement, and market research. From targeted advertising to influencer collaborations, companies leverage social media platforms to cultivate brand identity, foster customer loyalty, and drive sales. Amidst the bustling digital landscape, users navigate through a myriad of content and trends, encountering topics that evoke a spectrum of emotions. However, alongside uplifting and inspiring content, social media users often come across discussions permeated with negativity, ranging from political controversies to personal grievances.

In response to this challenge, there arises a pressing need for innovative solutions that prioritize user well-being and emotional health. One such solution is the development of a sophisticated system capable of automatically segmenting tweets by topics and analyzing the underlying sentiment. By leveraging advanced natural language processing techniques, this system can categorize sentiments as positive, negative, or neutral, providing users with insights into the emotional tone of discussions.

Moreover, through intuitive visualization techniques, such as color-coded sentiment labels for each topic, highlighting positive sentiment in green, negative sentiment in red, and neutral sentiment in grey, users can navigate social media content with heightened emotional awareness. By empowering users to identify trends characterized by negative sentiment, this solution promotes mental well-being and fosters a healthier digital environment.

Ultimately, the primary objective of this project is to enhance the social media experience by offering users the tools and insights needed to make informed decisions about their engagement. By facilitating a deeper understanding of the emotional nuances underlying online discussions, this approach encourages mindful interaction and promotes a more positive and constructive online community.

To achieve this, we leverage advanced natural language processing techniques, including Long Short-Term Memory (LSTM) networks for sentiment analysis and TextRazor for topic extraction. By contributing to sentiment analysis and topic extraction methods, this research has practical implications for various domains, including marketing, brand management, and public opinion research. Moreover, it sets the stage for further advancements in natural language processing.

## 2. LITERATURE REVIEW

According to the study [1], employing big data techniques for sentiment analysis on Twitter data presents significant opportunities for extracting valuable insights from social media discussions. Utilizing these techniques, businesses and organizations can tap into the wealth of Twitter data to gain deeper insights into public sentiment and improve decision-making processes. The authors reported an approximate accuracy of 75% for sentiment classification.

In the recent paper [2] provided a comprehensive review of contemporary research employing deep learning techniques in sentiment analysis, focusing on sentiment polarity. The study covered a variety of models utilizing methods like TF-IDF and word embedding across different datasets. Additionally, a comparative analysis of these models was conducted, concluding that while the convolutional neural network (CNN) model offers the best balance between processing time and accuracy, the recurrent neural network (RNN) model achieves the highest level of precision. This comparative assessment provides valuable insights for practitioners in the field of sentiment analysis.

The study [3] emphasizes the significance of understanding public opinion, particularly on platforms like Twitter, within contemporary decision-making processes. They highlight the challenges associated with analyzing sentiments in Twitter data, such as its massive volume, linguistic diversity, and the brevity of tweets. To address these challenges, the authors propose a novel approach that gamifies sentiment analysis, actively involving users in classifying tweet polarity and sentiment. A web-based game was developed, and participants were enlisted to play, utilizing a dataset comprising 52,877 tweets. The study's results, validated through ground-truth comparisons and manual assessments, demonstrate the game's efficacy in accurately measuring sentiments. Participant feedback underscores the enjoyment derived from the gaming experience, affirming the viability of this innovative method in sentiment analysis.

The researcher [4] explores the challenge of applying topic models to the brief and sparse texts commonly found in social media micro-blogs, such as Twitter. A comparison of three models is carried out, which include the standard Latent Dirichlet Allocation (LDA), the Gibbs Sampler Dirichlet Multinomial Model (GSDMM), and the Gamma Poisson Mixture Model (GPM), the latter two being specifically tailored for sparse data. Proposing a novel evaluation approach involving the simulation of pseudo-documents, the authors focus their analysis on tweets related to the COVID-19 pandemic. Surprisingly, they find that traditional coherence scores, typically utilized for evaluating topic models, exhibit inadequacy as evaluation metrics. Through their simulation-based methodology, the authors suggest that both the GSDMM and GPM models may offer improved topic generation compared to the standard LDA model. These findings underscore the necessity of considering alternative evaluation methods and specialized models when grappling with short and sparse text data within social media micro-blogs.

## 3. METHOD

In the realm of natural language processing (NLP) and text mining, understanding the intricacies of textual data is paramount for deriving meaningful insights. The journey from raw text to actionable insights typically involves several key steps. Here as seen in Figure 1, we outline the core components of a comprehensive text analysis framework, each serving a crucial purpose in unlocking the latent value within textual data.

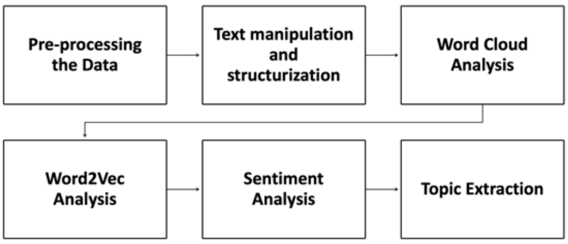



Figure 1. Overview of Text Analysis Framework

### 3.1. Preprocessing the data

Text data often comes in raw, unstructured formats, making it challenging to derive meaningful insights directly [6]. Therefore, a series of preprocessing steps are typically applied to clean, organize, and prepare the data for further analysis. The dataset used for the paper is from Hussein, Sherif [5], which comprises 107,765 tweets categorized into three sentiments: negative, neutral, and positive. Each entry in the dataset contains two fields: the tweet text and its corresponding sentiment label. The following steps outline a systematic approach to processing text data.

#### 3.1.1. Missing values

Identifies and handles any missing values in the dataset to prevent errors and ensure data quality.

#### 3.1.2. Label encoding

Converts categorical sentiment labels into numerical values for model training, facilitating compatibility with machine learning algorithms.

### 3.2. Text manipulation and structurization

Text manipulation involves transforming raw text into a structured format that is amenable to analysis. This step includes removing irrelevant elements such as URLs, emojis, and punctuation marks, tokenizing the text into individual words or phrases, and converting text to lowercase for consistency given in Figure 2. By structuring the text appropriately, we lay the foundation for more advanced analyses.

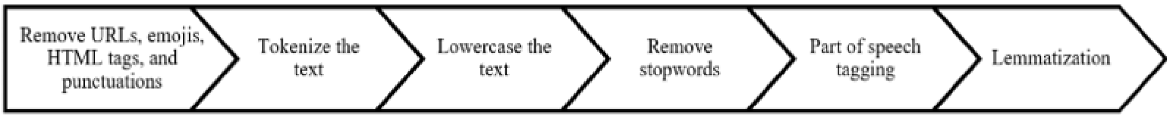



Figure 2. Text Manipulation and Structurization Process

To preprocess text data for sentiment analysis, the first step is to remove URLs, emojis, HTML tags, and punctuations, which eliminates irrelevant or noisy elements, allowing the focus to remain on meaningful content. The text is then tokenized, breaking it down into individual words or tokens, facilitating further analysis and manipulation. Normalizing the text by converting all characters to lowercase helps avoid duplicate entries and improves consistency. Removing stop words, which are common words that do not carry significant meaning, reduces noise and enhances the accuracy of the analysis. Part of speech tagging assigns grammatical categories such as nouns, verbs, and adjectives to each word, allowing for the extraction of insights based on linguistic patterns. Finally, lemmatization converts words to their base or root forms, standardizing the vocabulary and improving interpretability. This comprehensive preprocessing pipeline ensures that the text data is clean, consistent, and ready for effective analysis.

### 3.3. Word Cloud Analysis

A word cloud analysis also referred to as a tag cloud or text cloud, is a visual depiction of text data. The size, and sometimes colour, represents how often each word occurs. This method allows for visual exploration and emphasis on the most important terms or concepts within a text. It's commonly used in text mining, natural language processing, and qualitative data analysis to swiftly uncover dominant themes, keywords, or patterns in textual data. Figure 3 gives the word cloud for the tweets from the dataset.

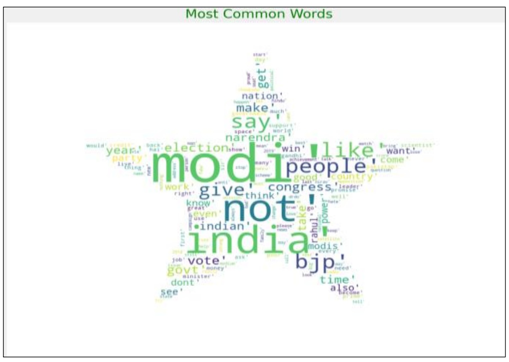



Figure 3. Word Cloud displays the most common words

### 3.4. Word2Vec Analysis

Word2Vec, a prominent method in the field of Natural Language Processing (NLP), is used to create word embeddings [7]. These dense vector representations of words in a continuous space were introduced by Tomas Mikolov and his team at Google in 2013. Word2Vec has become a popular choice due to its ability to capture the semantic relationships between words. It offers a robust method to represent and analyze text data in NLP tasks. By learning distributed representations of words, Word2Vec procures rich semantic information that can boost performance across a variety of NLP applications. Figure 4 explains the process of performing Word2Vec analysis.

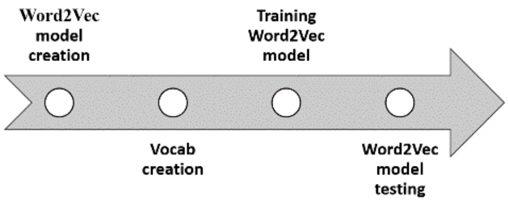



Figure 4. Word Cloud displays the most common words

#### 3.4.1. Word2Vec model creation

Constructs a Word2Vec model, a neural network-based technique, to learn distributed representations (word embeddings) of words based on their context in the corpus.

#### 3.4.2. Vocab creation

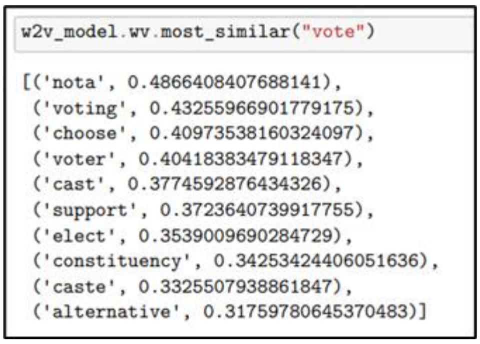
Builds a vocabulary comprising unique words from the corpus to map them to vector representations in the Word2Vec model.

#### 3.4.3. Training Word2Vec model

Trains the Word2Vec model on the corpus to learn the vector representations of words that capture semantic similarities and relationships.

#### 3.4.4. Word2Vec model testing

Evaluates the trained Word2Vec model to ensure that it generates meaningful word embeddings that reflect semantic properties. Test the trained Word2Vec model to display words similar to 'vote' as shown in Figure 5.



```
w2v_model.wv.most_similar("vote")

[('nota', 0.4866408407688141),
 ('voting', 0.43255966901779175),
 ('choose', 0.40973538160324097),
 ('voter', 0.40418383479118347),
 ('cast', 0.3774592876434326),
 ('support', 0.3723640739917755),
 ('elect', 0.3539009690284729),
 ('constituency', 0.34253424406051636),
 ('caste', 0.3325507938861847),
 ('alternative', 0.31759780645370483)]
```

Figure 5. Test the trained Word2Vec model to display words similar to 'vote.'

### 3.5. Sentiment Analysis Using Deep-Learning Model

Sentiment analysis is a natural language processing (NLP) technique used to determine the sentiment or opinion expressed in a piece of text. It involves identifying and categorizing opinions, emotions, attitudes, and feelings expressed by individuals towards a particular subject, product, service, or topic. Sentiment analysis aims to understand the overall sentiment conveyed in the text, whether it is positive, negative, or neutral [8].

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture that is well-suited for sequence prediction problems due to its ability to capture long-term dependencies in data. When combined, sentiment analysis using LSTM involves utilizing LSTM networks to analyze and classify the sentiment of textual data [9]. LSTM networks are particularly effective for sentiment analysis tasks because they can effectively model the sequential nature of language and capture dependencies between words in a sentence. This capability allows LSTM models to understand the context and nuances of language, making them well-suited for sentiment analysis tasks where the sentiment expressed may depend on the overall context of the text.

The word "satisfaction" in the sentence "Nothing beats the satisfaction of taking that first bite of a perfectly swirled soft serve cone on a hot summer day" implies Happiness, indicating a positive sentiment. For example: Is the product receiving positive or negative feedback? Do consumers feel satisfied with the product? Are feelings that are more positive than negative predominating? The questions similar to the above will help in sentiment analysis [10]. The customer feedback can be taken for sentiment analysis for an organization/company to get input about how the customers feel about a business's goods or services. This section outlines the key steps involved in creating, training, evaluating, and utilizing LSTM models for sentiment analysis, as shown in Figure 6.

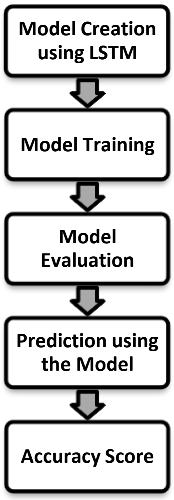
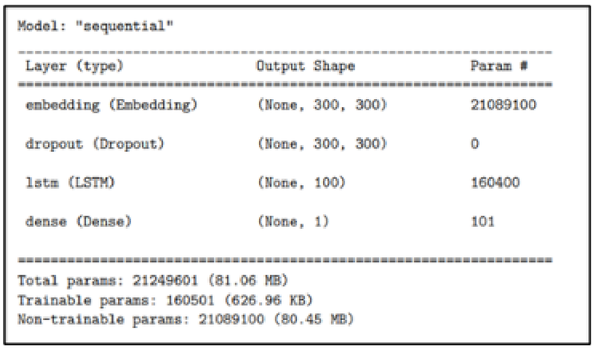



Figure 6. Steps involved in developing the LSTM model

### 3.5.1. Model Creation using LSTM

In this step, we create a deep-learning model based on LSTM architecture, as seen in Figure 7. LSTM networks are particularly effective for sequential data like text due to their ability to capture long-range dependencies [11]. The model architecture includes an embedding layer, an LSTM layer, a dropout layer for regularization, and a dense output layer with sigmoid activation for binary classification. The embedding layer converts text data into dense vector representations, facilitating the learning process for the subsequent LSTM layer.



Model: "sequential"

| Layer (type) | Output Shape | Param # |
| --- | --- | --- |
| embedding (Embedding) | (None, 300, 300) | 21089100 |
| dropout (Dropout) | (None, 300, 300) | 0 |
| lstm (LSTM) | (None, 100) | 160400 |
| dense (Dense) | (None, 1) | 101 |

Total params: 21249601 (81.06 MB)
Trainable params: 160501 (626.96 KB)
Non-trainable params: 21089100 (80.45 MB)

Figure 7. The layers in the LSTM model

### 3.5.2. Model Training

With the model architecture defined, we proceed to train the LSTM model on the training dataset. During training, the model learns to map input sequences of words to their corresponding sentiment labels (positive or negative) through the process of gradient descent optimization [12]. The training process involves iteratively adjusting the model's parameters to minimize the difference between predicted and actual sentiment labels.

### 3.5.3. Model Evaluation

Once training is complete, we evaluate the performance of the LSTM model using a separate validation dataset. Evaluation metrics such as accuracy, precision, recall, and F1-score are computed to assess how well the model generalizes to unseen data. Additionally, a confusion matrix shown in Figure 8 is generated to visualize the model's predictions and identify any misclassifications between positive and negative sentiments.

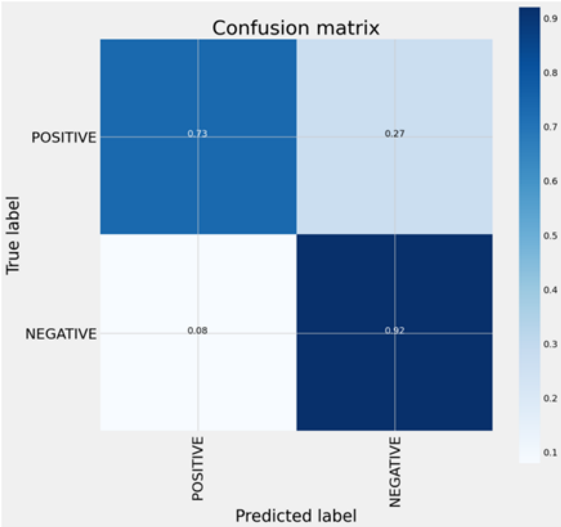



Figure 8. The confusion matrix obtained for the proposed LSTM model

### 3.5.4. Prediction using the model

After evaluation, the trained LSTM model is deployed to make predictions on new, unseen text data. Given a piece of text input, the model predicts the sentiment associated with it, classifying it as either positive or negative [13]. Predictions can be made for individual text samples or batches of text samples, depending on the application requirements.

### 3.5.5. Accuracy Score

After evaluation, the trained LSTM model is deployed to make predictions on new, unseen text data. Given a piece of text input, the model predicts the sentiment associated with it, classifying it as either positive or negative. Predictions can be made for individual text samples or batches of text samples, depending on the application requirements.

### 3.6. Extract the Topic using TextRazor

TextRazor leverages a combination of cutting-edge NLP techniques and a vast knowledge base to provide industry-leading entity recognition, disambiguation, and linking capabilities. At the core of TextRazor's technology lies its sophisticated algorithms, which are meticulously designed to process and analyze textual data with precision and efficiency.

Named Entity Recognition (NER) serves as a foundational aspect of TextRazor's technology, allowing it to identify entities such as People, Places, and Companies within text. This process involves parsing through the linguistic context surrounding each entity mentioned, utilizing techniques like part-of-speech tagging and dependency parsing to accurately label and categorize entities.

To address the inherent ambiguity of language, TextRazor employs advanced disambiguation strategies. By combining signals from various sources—including shallow methods, graph-based analyses, and deep linguistic insights—TextRazor assigns confidence scores to each entity, enabling it to resolve ambiguity and provide contextually accurate interpretations.

Furthermore, TextRazor excels in linking recognized entities to canonical IDs in linked datasets such as Wikipedia, DBPedia, and Wikidata. This linkage not only enriches the extracted information but also facilitates seamless integration with existing applications reliant on linked data, thus enhancing the utility and interoperability of TextRazor's outputs.

Overall, TextRazor's technology represents a fusion of state-of-the-art NLP methodologies and a rich knowledge base, enabling it to deliver unparalleled accuracy and comprehensiveness in extracting meaningful insights from textual data across diverse languages and domains.

To utilize the TextRazor API for topic extraction, the first step is to set the TextRazor API key, which authenticates access to the API's capabilities. Next, create a TextRazor client object to facilitate interaction with the API and enable text analysis tasks. Prepare a list of sentences or text snippets to be analyzed for topic extraction. Iterate through each sentence in this list, and for each sentence, send the text data to the TextRazor API for analysis. This process leverages TextRazor's natural language processing capabilities to extract relevant topics. Finally, print the extracted topics from each analyzed sentence to gain insights into the underlying themes or subjects discussed in the text.

## 4. RESULTS AND DISCUSSION

The results demonstrate the effectiveness of the developed system in performing sentiment analysis and topic extraction on social media text data. The utilization of the Long Short-Term Memory (LSTM) neural network for sentiment classification yielded accurate categorization of text into positive, negative, or neutral sentiments.

Figure 9 displays the results of a sentiment analysis model applied to three different sentences. The first sentence, "Just discovered an amazing new artist on Spotify, and I'm already hooked. Can't wait to see what else they have in store," is classified as positive with a high confidence score of 0.987, reflecting the enthusiastic and excited tone conveyed by words like "amazing" and "hooked." The second sentence, "I have nothing to lose," is labelled neutral with a confidence score of 0.641, indicating a lack of strong positive or negative emotion, consistent with the phrase's balanced tone. The third sentence, "worst exam ever," is predicted as negative with a confidence score of 0.749, capturing the clear negative sentiment conveyed by the word "worst." Each Prediction was made in approximately 0.15 to 0.17 seconds, demonstrating the model's efficiency in analyzing sentiments.

```
predict("Just discovered an amazing new artist on Spotify and I'm already␣
    ↪hooked. Can't wait to see what else they have in store")

1/1 [==============================] - 0s 77ms/step

{'label': 'POSITIVE',
 'score': 0.9873714447021484,
 'elapsed_time': 0.14923906326293945}

predict("I have nothing to lose")

1/1 [==============================] - 0s 95ms/step

{'label': 'NEUTRAL',
 'score': 0.6411340832710266,
 'elapsed_time': 0.1727430820465088}

predict("worst exam ever")

1/1 [==============================] - 0s 91ms/step

{'label': 'NEGATIVE',
 'score': 0.7497435212135315,
 'elapsed_time': 0.1691293716430664}
```

Figure 9. Testing the model to predict the sentiments (positive, neutral and negative)

The sentiment analysis model demonstrates an overall accuracy of 86% from Table 1, indicating that it correctly predicts the sentiment for the vast majority of instances. Accuracy, defined as the ratio of correctly predicted instances (both true positives and true negatives) to the total number of instances, measures the overall correctness of the model's predictions.

For the positive class, the precision is 88%, meaning that out of all predicted positive instances, 88% are actual positives. Precision is calculated as the ratio of true positives to the sum of true positives and false positives, indicating the accuracy of positive predictions. This high precision is complemented by a recall of 92%, which is calculated as the ratio of true positives to the sum of true positives and false negatives, showing that the model effectively identifies 92% of the true positive instances. Consequently, the F1 score for the positive class is 0.90, which is the harmonic mean of precision and recall, reflecting a strong balance between these two metrics.

For the negative class, the precision is 82%, indicating a slightly lower accuracy in predicting negative instances, while the recall is 73%, meaning the model correctly identifies 73% of the true negative instances.

The F1 score for the negative class is 0.77, signifying a relatively lower performance in detecting negative sentiments. This is computed similarly to the positive class, balancing precision and recall.

The macro average precision of 0.85 and the macro average F1-score of 0.84 illustrate that, on average, the model performs well across both classes when treating them equally. The macro average is the mean of precision (or other metrics) for each class, treating all classes equally regardless of their size. However, the weighted averages for precision, recall, and F1-score are all 0.86, reflecting the model's consistent performance across the dataset while considering the class distribution, which is more heavily weighted towards the positive class. The weighted average takes into account the proportion of each class in the dataset, providing a more balanced measure for datasets with class imbalance.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

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

$$Negative\ Precision = \frac{TN}{TN+FN} \tag{3}$$

$$Positive\ Recall = \frac{TP}{TP+FN} \tag{4}$$

$$Negative\ Recall = \frac{TN}{TN+FP} \tag{5}$$

$$f1\ score = 2 * \frac{PRECISION*RECALL}{PRECISION+RECALL} \tag{6}$$

$$Macro\ Average = \frac{Precision\ of\ class\ 0 + precision\ of\ class\ 1}{2} \tag{7}$$

$$Weighted\ Average = \frac{(TP\ of\ class\ 0 + TP\ of\ class\ 1)}{(total\ number\ of\ class\ 0 + total\ number\ of\ class\ 1)} \tag{8}$$

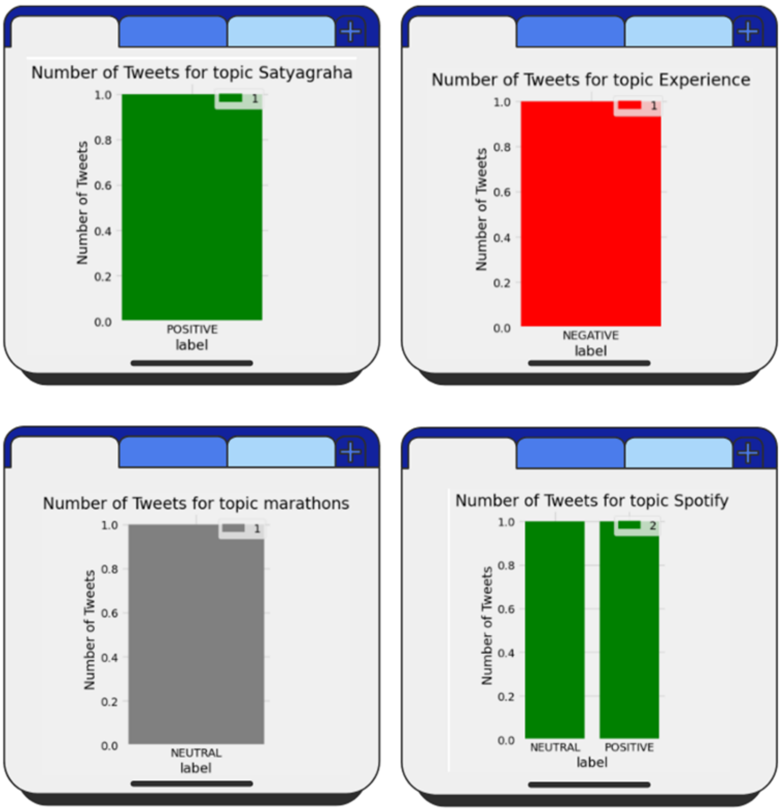



Figure 10. Comparing simulation results in a bar graph

Table 1. The results obtained for the proposed LSTM model

| | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Negative | 0.82 | 0.73 | 0.77 | 7026 |
| Positive | 0.88 | 0.92 | 0.90 | 14526 |
| | | | | |
| Accuracy | | | **0.86** | 21552 |
| Macro average | 0.85 | | 0.84 | 21552 |
| Weighted average | 0.86 | 0.86 | 0.86 | 21552 |

Figure 10 presents a series of bar graphs that offer insights into the sentiment trends and topic distributions across various subjects on Twitter. Each bar graph is colour-coded to represent different sentiment labels, making it easy to understand the overall sentiment for each topic at a glance.

The first bar graph, labelled "Number of Tweets for topic Satyagraha," displays a single green bar indicating that the sentiment associated with tweets on this topic is overwhelmingly positive. The second graph, titled "Number of Tweets for topic Experience," features a red bar, signifying that the sentiment is predominantly negative for this topic. In the third graph, "Number of Tweets for topic marathons," a grey bar is shown, representing a neutral sentiment. This suggests that tweets about marathons are generally neutral in tone. Lastly, the fourth graph, "Number of Tweets for topic Spotify," shows two bars: one green and one grey. The green bar indicates positive sentiment, while the grey bar indicates neutral sentiment, with the positive sentiment slightly outweighing the neutral.

These visualizations, through their clear and intuitive colour-coded sentiment labels, not only highlight the dominant sentiments but also facilitate mindful interaction and enrich user well-being by providing an easily interpretable overview of public opinion on various topics. Overall, the results of this paper underscore the potential of advanced natural language processing techniques in addressing the challenges of misinformation and negativity prevalent on social media platforms. By offering a comprehensive solution for sentiment analysis and topic extraction, coupled with intuitive visualizations, the developed system contributes to the enhancement of user well-being and the cultivation of a healthier social media experience.

## 5. CONCLUSION

In conclusion, this paper highlights the intricate dynamics within social media and emphasizes the necessity of fostering a positive and constructive digital environment. Leveraging the Long Short-Term Memory (LSTM) neural network, the system achieves precise sentiment classification, enabling the accurate categorization of text into positive, negative, or neutral sentiments. Furthermore, employing TextRazor for topic extraction enhances the system's capability to identify key themes and subjects within the text data. This amalgamation of techniques enables a comprehensive understanding of textual content, empowering businesses and organizations to derive actionable insights from vast volumes of unstructured data. The robust performance of the proposed LSTM model, with an impressive accuracy of 86% in sentiment classification, underscores the efficacy of our approach, reinforcing the potential impact of our system in fostering a healthier digital discourse.

By addressing the pervasive challenges of misinformation and negativity on social media platforms, the development of this innovative system, capable of both sentiment analysis and topic extraction, offers a promising solution. Through the provision of insights into sentiment trends and topic distributions, complemented by intuitive visualizations such as colour-coded sentiment labels, which highlight positive sentiments in green and negative sentiments in red, the system promotes mindful interaction and enriches user well-being. Looking ahead, this research sets the stage for further advancements in natural language processing, contributing to ongoing endeavours aimed at cultivating a healthier and more engaging social media experience for all users.

## DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

## CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest in this work.

## REFERENCES

[1] O. Almonajed and S. Jukić, "Sentiment Analysis on Twitter Data using Big Data," *J. Eng. Nat. Sci.*, vol. 3, no. 1, 2021, doi: 10.14706/JONSAE2021311.

[2] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment Analysis Based on Deep Learning: A Comparative Study," *Electronics*, vol. 9, no. 3, p. 483, Mar. 2020, doi: 10.3390/electronics9030483.

[3] M. Furini and M. Montangero, "Sentiment analysis and Twitter: a game proposal," *Pers. Ubiquitous Comput.*, vol. 22, no. 4, pp. 771–785, Aug. 2018, doi: 10.1007/s00779-018-1142-5.

[4] C. Weisser *et al.*, "Pseudo-document simulation for comparing LDA, GSDMM and GPM topic models on short and sparse text using Twitter data," *Comput. Stat.*, vol. 38, no. 2, pp. 647–674, Jun. 2023, doi: 10.1007/s00180-022-01246-z.

[5] Y. Chai, D. Kakkar, J. Palacios, and S. Zheng, "Twitter Sentiment Geographical Index Dataset," *Sci. Data*, vol. 10, no. 1, p. 684, Oct. 2023, doi: 10.1038/s41597-023-02572-7.

[6] Y. Qi and Z. Shabrina, "Sentiment analysis using Twitter data: a comparative application of lexicon- and machine-learning-based approach," *Soc. Netw. Anal. Min.*, vol. 13, no. 1, p. 31, Feb. 2023, doi: 10.1007/s13278-023-01030-x.

[7] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: A survey," *WIREs Data Min. Knowl. Discov.*, vol. 8, no. 4, Jul. 2018, doi: 10.1002/widm.1253.

[8] U. Naseem, I. Razzak, M. Khushi, P. W. Eklund, and J. Kim, "COVIDSenti: A Large-Scale Benchmark Twitter Data Set for COVID-19 Sentiment Analysis," *IEEE Trans. Comput. Soc. Syst.*, vol. 8, no. 4, pp. 1003–1015, Aug. 2021, doi: 10.1109/TCSS.2021.3051189.

[9] P. Mishra, S. A. Patil, U. Shehroj, P. Aniyeri, and T. A. Khan, "Twitter Sentiment Analysis using Naive Bayes Algorithm," in *2022 3rd International Informatics and Software Engineering Conference (IISEC)*, IEEE, Dec. 2022, pp. 1–5. doi: 10.1109/IISEC56263.2022.9998252.

[10] A. S. M. Alharbi and E. de Doncker, "Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information," *Cogn. Syst. Res.*, vol. 54, pp. 50–61, May 2019, doi: 10.1016/j.cogsys.2018.10.001.

[11] A. Aljebreen, W. Meng, and E. Dragut, "Segmentation of Tweets with URLs and its Applications to Sentiment Analysis," *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 14, pp. 12480–12488, May 2021, doi: 10.1609/aaai.v35i14.17480.

[12] Y. Yu, X. Si, C. Hu, and J. Zhang, "A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019, doi: 10.1162/neco_a_01199.

[13] A. Farzad, H. Mashayekhi, and H. Hassanpour, "A comparative performance analysis of different activation functions in LSTM networks for classification," *Neural Comput. Appl.*, vol. 31, no. 7, pp. 2507–2521, Jul. 2019, doi: 10.1007/s00521-017-3210-6.


## BIOGRAPHIES OF AUTHORS

**Jency Jose** is an Assistant Professor at Mount Carmel College in Bengaluru. With a profound passion for computer science, Jency has dedicated herself to both teaching and research, aiming to impart knowledge and make impactful contributions to the field. With 14 years of teaching experience, she brings a wealth of knowledge and expertise to her role. Over the course of her career, she has authored and co-authored numerous papers, each contributing to the advancement of computer science. She can be contacted at email: j.jency@gmail.com

**Simritha R**. received her bachelor's degree in Computer Application from Ethiraj College for Women in Chennai in 2021. She is currently pursuing her Master's in Computer Science at Mount Carmel College in Bengaluru. Simritha has already made strides in academic publishing with a paper in the Indian Journal of Natural Sciences. Her primary research interests lie in Natural Language Processing (NLP) and Deep Learning. She can be contacted at email: simritha14@gmail.com.