**Evaluating Sentiment Analysis Models for Social Media: A Literature Review**

Kai-Kian Toh2, a), Lee-Ying Chong1, 2, b), Siew-Chin Chong1, 2, c) and Pey-Yun Goh1, 2, d)

1Centre for Advanced Analytics, COE of Artificial Intelligence, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia

2Faculty of Information Science and Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia

b) Corresponding author: lychong@mmu.edu.my

a) kaikian1105@gmail.com

c)chong.siew.chin@mmu.edu.my

d)pygoh@mmu.edu.my

**Abstract.** Sentiment analysis helps people understand the emotions, attitudes and opinions share on social media. This paper reviews 13 studies published between 2020 and 2024, covering 17 different models. Some of these models use classic machine learning methods like Support Vector Machines and Naïve Bayes, while others use deep learning techniques such as Convolutional Neural Networks and Transformer-based architectures. Each model’s setup, training process and performance are described, along with their main strengths and limitations. The review showed that deep learning models often achieve higher accuracy and can handle messy, unstructured text more easily. At the same time, traditional methods remain popular because they are simple to use and do not demand as much computing power. Overall, this review paints a clear picture of today’s sentiment analysis tools and suggests how future systems might balance accuracy with efficiency.

# **INTRODUCTION**

Sentiment analysis uses natural language processing to determine whether social media text is positive, negative or neutral based on the intensity and polarity of words. It is widely used to measure feelings in comments, tweets and reviews on platforms such as Twitter, Facebook and online forums. Researchers select methods such as logistic regression, convolutional neural networks and hybrid techniques that combine multiple approaches depending on data characteristics and research objectives.

This paper reviews 13 studies on social media sentiment analysis published between 2020 and 2024. Studies were chosen because they focus on sentiment tasks applied to platform‑native content (for example, tweets, comments and user reviews) and appear in peer‑reviewed journals or reputable conferences. Work that deals only with formal text (such as news articles), out‑of‑scope languages or datasets smaller than 5 000 posts has been excluded. In total, these studies cover 17 different models ranging from classic machine‑learning classifiers through modern transformer architectures to hybrid deep‑learning systems. Model designs, training methods and performance scores are compared, and key strengths and weaknesses appear in a side‑by‑side table. The objective is to highlight current trends and provide practical guidance for choosing the most appropriate sentiment‑analysis approach.

# **LITERATURE REVIEW**

Younas et al. [1] combined two transformer-based architectures which are Multilingual BERT (mBERT) and XLM-RoBERTa (XLM-R). mBERT is an extension of BERT trained on over 100 languages using a shared vocabulary, making it suitable for multilingual tasks [2]. XLM-R, on the other hand, was introduced by Conneau et al. [3] and improves cross-lingual understanding through more extensive training data and dynamic masking strategies. Both models begin with cub-word tokenization of code-mixed Roman Urdu-English texts that overlays BERT. The special [CLS] token produces a pooled representation, which is then passed through a softmax layer to classify the text into one of three sentiment categories: positive, neutral, or negative. By contrast, XLM-R is a variant of BERT-large in structure which uses dynamic Language Masking to highlight the gaps between code-mixed languages. It is trained on 2.5TB of filtered CommonCrawl data. The researchers tune the key hyperparameters such as batch size, learning rate, and number of epochs to find the best performance. After finetuning on the MultiSenti dataset, mBERT has an accuracy of 0.69, while XLM-R comes to an even higher 0.71 on this. This shows that in practice, XLM-R outperforms other methods for mixing of language styles. Therefore, it can conclude that XLM-R has better performance than mBERT when operating on code-mixed social media text, at least as measured by classification accuracy.

Kavitha et al. [4] utilized the Random Forest, Decision Tree, and Logistic Regression (LR) for sentiment analysis on Twitter data. Random Forest builds an ensemble of decision trees by randomly sampling data points and features at each split. Each tree makes its own prediction, and the final sentiment is decided by a majority vote from all trees. In contrast, Decision Tree builds a single tree by repeatedly choosing the most informative feature and using measures like information gain to split the data until a stopping condition is met. This process directly assigns a sentiment label based on the final node reached. Meanwhile, LR calculates the probability of each emotion category by applying the sigmoid function to the weighted sum of TF-IDF features and adjusts the weights to minimize the prediction error via gradient descent. The comparison results show that the accuracy of Random Forest is 95%, accuracy of Decision Tree is 83%, and the accuracy of Logistic Regression is 78%.

Chen [5] presented a deep learning framework for sentiment analysis of medical texts that integrated a pre-trained BERT model with three distinct deep learning modules: a Fully Connected Network (FCN), a Convolutional Neural Network (CNN), and a Graph Convolutional Network (GCN). To generate contextualized word embeddings, the framework starts by feeding pre-processed medical text into a BERT layer (or its variant, COVID-TWITTER-BERT, when fine-tuned for social media environments). The FCN module, implemented as a two-layer network with ReLU activation, directly maps these embeddings to sentiment scores, capturing general sentiment trends. The CNN module applies one-dimensional convolutional layers to extract local, high-level features from the BERT output, enabling the identification of important sentiment cues in text segments. Notably, CNNs have been effectively utilized in sentence classification tasks, as demonstrated by Kim [6], highlighting their capability in capturing local features in text. The GCN module [7], builds a graph representation of the text where words are nodes, and their semantic dependencies are edges, using two GCN layers to capture complex inter-word interactions. Experimental results on the METS-CoV dataset show that the CNN-based approach achieves the best performance with the highest accuracy and F1 score, particularly when combined with COVID-TWITTER-BERT, as it excels at extracting fine-grained local features.

Two transformer-based models, BERT and RoBERTa, as described in [8], are employed to perform sentiment analysis on Twitter data by leveraging the inherent strengths of transformer architectures and the benefits of transfer learning. Initially BERT [2] was introducing to bidirectional training for deep language understanding, and enhanced variant of Bert, RoBERTa [9] become an optimized version with more extensive pretraining and data. In this approach, the BERT model undergoes a meticulous preprocessing stage where the text is tokenized into sub-words and special tokens such as [CLS] and [SEP] are added to frame the context, which is then fine-tuned on a sentiment labelled dataset over ten epochs. This process steadily improved the model’s accuracy from 95.10% to 99.16% while the loss gradually decreased. Meanwhile, since the RoBERTa model is enhanced version that was pre-trained longer on a larger corpus. Its process is similar but slightly modification in its tagging scheme and hyperparameter. The result showed that RoBERTa achieved better accuracy improvement which increasing from 99.53% to 99.70 %, and its loss value is lower. In conclusion, both models successfully capture the nuances of informal social media language, but extended pre-training of RoBERTa provides marginal benefits, but at the cost of higher computational requirements.

Random Forest, KNN, and Naive Bayes are applied in [10] to analyse social media sentiment. The Random Forest model builts a set of decision trees to capture the complex relationships between features. However, its average accuracy of 86.06% shows that it may miss some nuances of social media language. In contrast, the KNN classifier evaluates the proximity of instances in feature space. The result is a mediocre average accuracy of 89.28%, which means it is quite sensitive to parameters and distance metrics in ways that other machine-learning algorithms are not. Noticeably, the naïve Bayes classifier utilizes likelihood estimation on the condition of feature independence. The result was an average accuracy of 92.01%. It also showed robust precision, recall and F1 score.

Bidirectional Long Short-Term Memory (BiLSTM) and Convolutional Neural Network (CNN) are implemented in [11] to analyse sentiment in Bangla text. The BiLSTM model processes sequences in both forward and backward directions. This process helps to capture the long-range dependencies and subtle context needed for understanding sentiment. In contrast, the CNN model applies convolutional filters to directly extract local features and are the basis of computing n-gram patterns that signify a change in sentiment. Both models augmented with GloVe word embeddings to encode semantics and optimized by Adam optimizer. Which can be seen from the results, while BiLSTM model obtained an accuracy of 97.36% the CNN model won out with 99.43%.

Mishra et al. [12] applied a multinomial Naive Bayes algorithm to analyse Twitter sentiment. The method predicts whether a tweet is positive or negative by simply calculating the word frequency probability of each sentiment category. Although this method seems simple, it works very well. Results show that it achieves an overall accuracy of about 94%. The advantages of this model are its speed and scalability, making it a practical choice for real-time analysis of large amounts of Twitter data. However, because it assumes that each word acts independently, it sometimes ignores deeper relationships between words.

Pardeshi et al. [13] conducted a comparative analysis of Naive Bayes, Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) using various n-gram features. Naive Bayes calculates class probabilities based on word frequencies and performs modestly. However, its simple approach limits its ability to capture complex word interdependencies. Random Forest builds an ensemble of decision trees and aggregates their predictions, which helps reduce overfitting and handle non-linear patterns. However, its overall accuracy is only moderate. SVM stands out as the most effective model especially with word-character n-grams. By finding an optimal hyperplane in high-dimensional space to separate sentiments, it is resulting in the highest accuracy among the methods. KNN, being a straightforward similarity-based algorithm, lags because it struggles with high-dimensional data and misses subtle nuances in text.

Mehedy et al. [14] compared Naive Bayes, Support Vector Machines (SVM), Random Forest, and a Multi-Layer Perceptron (MLP) for sentiment analysis on Bengali Twitter data. Although Naive Bayes is simple and fast speed, but it only reached about 51% accuracy. It may be that its underlying assumption that all features are independent may oversimplify complex sentiment signals, especially for neutral posts. For SVM model which tries to maximize the margin between classes in high-dimensional also achieved around 51% accuracy, similar with Naïve Bayes. This showed that it is difficult to capture the nuances needed for classifying three sentiment categories. For the Random Forest model, it combines multiple decision trees to enhance the robustness. The result showed that it achieved accuracy of 62% since it still struggled with very subtle sentiment differences. Most notably, the MLP is a feedforward neural network that can learn complex non-linear relationships. It achieved an accuracy of 87.31%, performing best in capturing the complex emotional patterns required for this task.

RoBERTa-LSTM, as presented in [15], is a hybrid model that combines the strengths of Transformer-based architectures and Recurrent Neural Networks to tackle sentiment analysis. In this approach, RoBERTa tokenizes and encodes raw text into contextualized word embeddings using a powerful byte-level Byte-Pair Encoding strategy. These embeddings are then passed to an LSTM layer, which excels at capturing long-range dependencies and sequential patterns in text. The output of the LSTM is flattened and further refined through dense layers, ultimately forming a SoftMax classification layer that predicts the probability of sentiment. Notably, the model combines data augmentation with GloVe word embedding to address the dataset imbalance problem and improve generalization ability. With fine-tuning of the hyperparameters (256 LSTM units, Adam optimizer, learning rate of 1e-5), the RoBERTa-LSTM model achieved 93%, 91%, and 90% of F1 scores on the IMDb, Twitter US Airline Sentiment, and Sentiment140 datasets. The overall accuracy is also significantly boosting compared to baseline models.

CNN and LSTM models are applied in [16] to analyze sentiment in social media text. The CNN model uses various kernel sizes and performs one-dimensional convolutions to extract local features and key emotional cues. It then employs a pooling layer for dimensionality reduction. In contrast, the LSTM model processes the text step by step, effectively capturing long-term dependencies and maintaining contextual information across sentences. Both models are finely adjusted by modifying parameters such as learning rates, batch sizes and the number of hidden layers. The upshot is that both models yield high accuracies and have both precisions followed by recall. For the CNN, it can fast find the local sentiment trend. However, long range relationships might escape notice. The LSTM, it can understand the context over a longer sentence, even though training process is slow, and it may suffer from problems like vanishing gradients.

Long Short-Term Memory (LSTM) and Naive Bayes are compared in [17] for sentiment analysis on Amazon product reviews. The Multinomial Naive Bayes classifier uses Bayes' theorem to estimate the probability that a review is positive, negative or neutral. Although it has potential for improving performance, however, it can miss those emotionally loaded hints conveyed by context as no word in Multinomial Naive Bayes has influence from previous ones. On the other hand, the LSTM model is constructed to ameliorate problems such as vanishing gradients, so it is proficient at maintaining long-term dependencies. This allows LSTM to better comprehend the context and flow of language used in the reviews. Thus, LSTM reaching an accuracy of 93% while Naive Bayes only gets 87% accuracy.

Safa Fahim et al [18] applied Logistic Regression, Decision Tree, K-Nearest Neighbors (KNN), and Random Forest for sentiment analysis on Twitter data. The Logistic Regression model uses a linear function to estimate the probability of each sentiment class and achieves the highest accuracy among the four models. The Decision Tree model constructs a hierarchical tree by splitting features based on entropy and information gain. This is resulting in moderate performance but with risks of overfitting. The KNN classifier assigns sentiment labels based on the majority vote of the k closest neighbors in the feature space. However, it shows the lowest performance, likely due to its sensitivity to noise and outliers. Meanwhile, the Random Forest model combines multiple decision trees using ensemble learning, which improves accuracy by reducing individual tree errors, though it requires higher computational resources. Overall, Logistic Regression leads in performance, followed by Random Forest, while the Decision Tree and KNN models are less effective in this sentiment analysis task.

# **Critical ANALYSIS**

Tables 1 and 2 show the key details, strengths and weaknesses of each study at a glance. Two clear trends stand out. First, models with greater capacity (such as ensembles or deep networks) generally outperform simpler ones on tasks that involve subtle or complex sentiment cues, suggesting that richer learning methods are better at picking up nuanced patterns. Second, all code‑mixed studies struggle with spelling variations and mixed‑language tokens, which standard word embeddings and tokenizers do not handle well. These points suggest that future work should explore higher‑capacity models alongside subword or character‑level techniques to cover both complex relationships and unusual word forms.

**TABLE 1.** Summary of 13 studies from 2020 to 2024

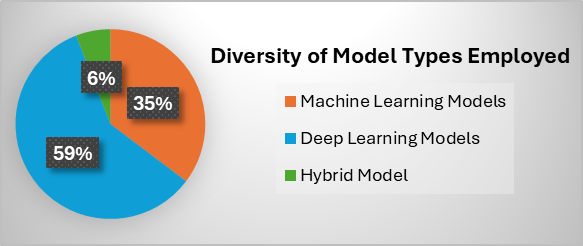
|  |  |  |  |
| --- | --- | --- | --- |
| **Author(s),**  **Year** | **Dataset Used** | **Dataset Language** | **Models Used & Accuracy** |
| Younas et al. [1] | MultiSenti dataset  (20,735 Pakistan’s 2018 election tweets) | * English * Roman Urdu | * mBERT: 69& * XLM-R: 71% |
| Kavitha et al. [4] | Twitter dataset  (31962 president debate tweets) | * English | * Random Forest: 95% * Decision Tree: 83% * Logistic Regression: 78% |
| Chen [5] | METS-CoV dataset (10,000 COVID-19 pandemic tweets) | * English | * FCN: 65.99%-66.79% * CNN: 66.29% -72.97% * GCN: 63.77%-70.73% |
| Prasanthi et al. [8] | COVID-19 NLP Text Classification dataset (over 50,000 COVID-19social media posts) | * English | * BERT: 99.16% * RoBERTa: 99.70% |
| Mridula et al. [10] | Kaggle dataset | * English | * Random Forest: 93.33% * KNN: 96.66% * Naive Bayes: 97.77% |
| Mahmud et al. [11] | Self-collected dataset (2272 social media data) | * Bangla | * BiLSTM: 97.36% * CNN: 99.43% |
| Mishra et al. [12] | Twitter collection dataset (31962 data) | * English | * Naïve Bayes: 94% |
| Pardeshi et al. [13] | Sentiment 140 dataset (1.6 million tweets) | * English | * Naïve Bayes: 55.89% * Random Forest: 62.54% |
| Mehedy. et al. [14] | Self-collected dataset and Kaggle open dataset (SentNoB) (14161 data) | * Bangla | * Naïve Bayes: 51% * Random Forest: 62% * SVM: 51% * MLP: 87.31% |
| Tan et al. [15] | IMDb dataset (50000 movie review), Twitter US Airline Sentiment dataset (14640 tweet) and Sentiment140 dataset (1.6 million tweets) | * English | * RoBERTa-LSTM: 89.70% -92.96% |
| Cao [16] | Multiple sources dataset (10,000 Twitter, Facebook, Instagram text posts) | * English | * CNN * LSTM |
| Meghana et al. [17] | Kaggle dataset (3.26 million reviews) | * English | * LSTM: 88% |
| Fahim et al. [18] | Kaggle dataset | * English | * Random Forest: ~77 % * Decision Tree: ~70% * Logistic Regression: ~78% * KNeighbors: ~51% |

# **DISTRIBUTION OF SENTIMENT ANALYSIS MODELS**

Figures 1 and 2 offer two clear views of how sentiment analysis models are used in social media studies. Together, they help readers grasp the main trends and the range of methods applied across the 13 papers. While Figure 1 highlights the variety of model types, Figure 2 shows which specific models appear most often, revealing researcher preferences over time.

**TABLE 2.** Summary of strengths and limitations of models

|  |  |  |  |
| --- | --- | --- | --- |
| **Models** | | **Strengths** | **Limitations** |
| mBERT [1] | Computationally efficient and flexible for multilingual tasks. | | Lower performance on code-mixed informal texts compared to XLM-R. |
| XLM-R [1] | Strong performance due to extensive pretraining and improved capture of code-mixed language nuances. | | Higher computational complexity and greater resource requirements during fine-tuning. |
| Logistic Regression [4][18] | Simple, efficient, and effective for both binary and multiclass classification tasks. | | Struggles with class imbalance and distinguishing similar sentiment categories; limited for non-linear patterns. |
| Random Forest [4][10][13][14][18] | Robust ensemble approach that reduces overfitting and effectively models non-linear interactions. | | Moderate F-score indicating uneven precision/recall; requires careful tuning and can be computationally intensive. |
| Decision Tree [4][18] | Simple and easy-to-interpret structure. | | Prone to overfitting. |
| FCN [5] | Simple structure with stable performance in capturing overall sentiment. | | Limited in capturing fine-grained or sequential features. |
| CNN [5][11][16] | Excels at extracting local features and contextual cues | | Requires careful tuning; may miss long-range dependencies |
| GCN [5] | Effective in modelling inter-word relationships. | | Less robust in leveraging deep features from BERT compared to CNN. |
| BERT [8] | Effectively captures contextual nuances in informal language through detailed tokenization and fine-tuning. | | Its performance, while robust, is slightly outperformed by RoBERTa, suggesting limitations in its pre-training scope. |
| RoBERTa [8] | Achieves higher accuracy due to extensive pre-training on large corpora. | | Demands more computational resources and longer fine-tuning time |
| KNN [10][13][18] | Leverages instance similarity effectively; intuitive and straightforward. | | Highly sensitive to parameter tuning and distance metric; struggles with high dimensionality and noise/outliers. |
| Naive Bayes [10][12][13][14][17] | Fast, scalable, and works well even with minimal training data. | | The assumption of feature independence oversimplifies natural language, potentially missing subtle sentiment cues. |
| BiLSTM [11] | Excels in capturing long-term contextual dependencies through bidirectional processing. | | Slightly less effective at detecting localized sentiment features compared to CNN. |
| SVM [13][14] | Performs well in high-dimensional feature spaces, with effective margin maximization leading to high accuracy. | | Computationally demanding and sensitive to kernel parameters; may struggle with nuanced ternary sentiment classification. |
| Multi-Layer Perceptron (MLP)[14] | Excellent at modelling non-linear patterns and achieving strong accuracy. | | Requires significant computational resources and careful parameter tuning. |
| RoBERTa-LSTM [15] | Combines RoBERTa’s rich, contextualized embeddings with LSTM’s sequence modelling, enhancing performance; data augmentation further boosts generalization. | | Computationally intensive with longer training times; demands extensive hyperparameter tuning and complex architecture makes interpretation and debugging more challenging. |
| LSTM [16][17] | Effectively maintains context over long sequences, capturing long-term dependencies for improved sentiment accuracy. | | Training is slower and can suffer from vanishing gradients; requires careful tuning and more computational resources. |

****

**FIGURE 1**.Diversity of model types employed

Figure 1 is a pie chart that groups the 17 models into three categories. Traditional machine learning methods make up 35% of the models, deep learning methods make up 59%, and hybrid models account for 6%. This split shows a growing interest in deep learning, since it can handle complex meanings and patterns in social media text. At the same time, traditional models still play a key role when computing power is limited or clear explanations are needed.

Figure 2 is a bar chart that counts how often each model appears in the reviewed studies. Random Forest and Naïve Bayes lead with five mentions each. CNN and KNN follow with three mentions each. Logistic Regression, Decision Tree, SVM, and LSTM each appear twice. Random Forest is praised for handling nonlinear patterns and reducing overfitting. Naïve Bayes stands out for its simplicity and speed on large, noisy datasets. CNNs excel at spotting local word groups that signal sentiment. KNN works well on smaller, well-structured data. Logistic Regression and SVM are chosen for their clear, easy-to-understand results. LSTM models handle longer text by keeping track of word order.

**FIGURE 2.** Most frequently used models

Overall, the combined analysis of Figures 1 and 2 underscores the considerations researchers face when choosing sentiment analysis models. Factors such as dataset size, computational resources, model interpretability, and the complexity of the text influence model selection. The reviewed studies reflect a pragmatic approach, balancing the sophistication of deep learning with the efficiency and accessibility of traditional techniques. This suggests that while deep learning continues to rise in popularity, traditional methods remain indispensable in various applied settings. These visual insights not only help to map current trends but also serve as a guide for future researchers in selecting appropriate models for their sentiment analysis tasks.

# **CONCLUSION**

This literature review shows that Naive Bayes and Random Forest are the most frequently used models in social media sentiment analysis. Deep learning methods appear in greater variety, but traditional models still retain a large share of applications. Random Forest often outperforms Naive Bayes because it can learn how words combine to express sentiment. CNN models excel at finding key phrases and local patterns and LSTM models capture context over longer spans of text. Transformer models add value by focusing on the most important tokens in each sentence. Code‑mixed data brings extra challenges, since spelling differences and mixed‑language tokens lead to many out‑of‑vocabulary errors. Subword or character‑level methods help reduce these errors by breaking words into smaller units. With the growing use of sentiment analysis, its applications extend to areas such as decision-making [19], elections [20], and beyond. This review is important to ensure the accuracy and reliability of these impactful applications. By choosing the right model, researchers capable to construct system that produce trustworthy and actionable insights.

# **REFERENCES**

1. Younas, Aqsa, et al. “Sentiment Analysis of Code-Mixed Roman Urdu-English Social Media Text Using Deep Learning Approaches.” *2020 IEEE 23rd International Conference on Computational Science and Engineering (CSE)*, IEEE, 2020, pp. 66–71.
2. Devlin, Jacob, et al. “BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.” *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, Association for Computational Linguistics, 2019, pp. 4171–4186.
3. Conneau, Alexis, et al. “Unsupervised Cross-Lingual Representation Learning at Scale.” *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2020, pp. 8440–8451.
4. Kavitha, M., et al. “Sentiment Analysis Using NLP and Machine Learning Techniques on Social Media Data.” *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, IEEE, 2022, pp. 112–115.
5. Chen, Yinan. “Sentiment Analysis of Medical Text Based on Deep Learning.” *2024 9th International Symposium on Computer and Information Processing Technology (ISCIPT)*, IEEE, 2024, pp. 478–482.
6. Kim, Yoon. “Convolutional Neural Networks for Sentence Classification.” arXiv:1408.5882, 2014.
7. Kipf, Thomas N., and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks.” arXiv:1609.02907, 2016.
8. Prasanthi, Kundeti Naga, et al. “A Novel Approach for Sentiment Analysis on Social Media Using BERT & ROBERTA Transformer-Based Models.” *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, IEEE, 2023, pp. 1–6.
9. Liu, Yinhan, et al. “RoBERTa: A Robustly Optimized BERT Pretraining Approach.” arXiv:1907.11692, 2019.
10. Mridula, B., et al. “Deciphering Social Media Sentiment for Enhanced Analytical Accuracy: Leveraging Random Forest, KNN, and Naive Bayes.” *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 2024, pp. 1410–1415.
11. Mahmud, Md. Shihab, et al. “Deep Learning Based Sentiment Analysis from Bangla Text Using Glove Word Embedding along with Convolutional Neural Network.” *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, 2022, pp. 1–6.
12. Mishra, Prashantkumar, et al. “Twitter Sentiment Analysis Using Naive Bayes Algorithm.” *2022 3rd International Informatics and Software Engineering Conference (IISEC)*, IEEE, 2022, pp. 1–5.
13. Pardeshi, Shyelendra Madansing, et al. “A Qualified Study of Machine Learning Algorithms for Sentiment Analysis in Social Media.” *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, IEEE, 2023, pp. 577–581.
14. Mehedy, Md., et al. “Ranking Mental Illness among Social Media Users.” *2021 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, IEEE, 2021, pp. 1–4.
15. Tan, Kian Long, et al. “RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis With Transformer and Recurrent Neural Network.” *IEEE Access*, **10**, 2022, pp. 21517–21525.
16. Cao, Ludan. “Sentiment Analysis of Social Media Text Based on Deep Learning.” *2023 3rd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, IEEE, 2023, pp. 1–5.
17. Meghana, Marella Sai, et al. “Sentiment Analysis on Amazon Product Reviews Using LSTM and Naive Bayes.” *2023 7th International Conference on Computing Methodologies and Communication (ICCMC)*, IEEE, 2023, pp. 626–631.
18. Fahim, Safa, et al. “Twitter Sentiment Analysis Based Public Emotion Detection Using Machine Learning Algorithms.” *2022 17th International Conference on Emerging Technologies (ICET)*, IEEE, 2022, pp. 107–112.
19. A.O. Aremu, and I. Muhammad, “Sentiment Analysis in Social Media: A Case Study of Hike in University School Fees in Selected Nigerian Universities,” *Journal of Informatics and Web Engineering* 3(2), 98–104 (2024).
20. A.W. W, H.H. Andana, J. Zeniarja, and A. Febriyanto, “Sentiment Analysis of the 2024 General Election Through Twitter using Long-Short-Term Memory Algorithm,” *Journal of Informatics and Web Engineering* 4(2), 387–400 (2025).