International Journal of Scientific Engineering and Research (IJSER)
ISSN (Online): 2347-3878
SJIF (2025): 8.036

AI-Based Sentiment Analysis on Social Media Using Deep Learning and NLP Techniques

Shraddha Dilip Kapase[1], Dr. Santosh Gaikwad[2]

[1]Department of MCA, JSPM University, Pune

[2]Guide

Abstract: ***The current research is dedicated to analyzing the user opinions expressed on social networks using novel computational techniques. With an ever-growing number of online communications, there arises a vast amount of text data, which cannot be processed manually. To address this issue, the proposed solution will apply Natural Language Processing approaches combined with different machine learning and deep learning models to automatically detect the sentiments in the given text. The models used for the study are as follows: SVM, LSTM, CNN + LSTM, and BERT model. They will help to classify social network content into positive, negative, and neutral sentiments. Such studies have practical applications in fields such as brand monitoring, customer feedback analysis, and public opinion tracking. The obtained outcomes indicate that combining classical models with deep learning models improves their efficiency in sentiment classification.***

Keywords: Sentiment Analysis, Social Media Data, Natural Language Processing, Deep Learning Models, Opinion Mining

1. Introduction

a) Background and Motivation

Social networking sites have become an indispensable means of communicating and sharing information in recent times. Millions of people share their views and experiences on various issues on the web every day. Data collected from these sources would help in gathering information about the sentiments of the population in various sectors.

This is clearly unrealistic and therefore has led to the development of automation systems for sentiment analysis. Sentiment analysis is a term used to refer to the process of identifying and categorizing the sentiments expressed in the form of text into positive, negative, or neutral.

It helps organizations and researchers make decisions based on data by understanding user views. Earlier methods for sentiment analysis used rule-based approaches and simple machine learning algorithms. While these methods were straightforward to use, they often failed to capture the complexity and context of human language.

In recent years, with advancements in artificial intelligence, the performance of deep learning models and transformers has significantly improved. This project aims to create an AI-driven sentiment analysis system that combines various techniques for better performance. By using both traditional and modern methods, the system can handle complex language patterns and deliver more reliable results.

b) Problem Statement

Despite progress in sentiment analysis techniques, several challenges remain. Social media data is often unstructured, noisy, and relies on context, making it hard for models to interpret correctly. Issues like sarcasm, slang, abbreviations, and mixed sentiments complicate the analysis process. Many current systems focus on a single model, which may not work well with different types of data.

Hence, what is needed now is an approach that utilizes a number of models for improved sentiment classification, as well as better handling of input data.

The objective of this work is to build and deploy a sentiment analysis framework that integrates several different models for sentiment classification. This would ensure accurate sentiment classification as well as overcoming the limitations of the current methodologies.

c) Research Objectives

Objectives of this paper include the following:

Creation of labeled social media datasets for multiclass sentiment classification on Twitter and Reddit. Development of an efficient pre-processing methodology to process NLP using social media text. Designing and setting the parameters of classifiers such as SVM, LSTM, CNN-LSTM, and BERT. Testing the above-stated models on the basis of their accuracy, precision, recall, and F1 scores using 10-fold cross validation. Finding out the mistakes and suggesting improvements.

d) Paper Organization

The structure of the paper is as follows; Literature Review is covered in Section II. The other sections include Methodology, Experimental Results, Applications, Limitations, and Conclusion in Sections III, IV, V, VI, and VII respectively.

2. Literature Review

a) Lexicon-based Approach and Rule-Based Approach:

In the first phase, lexicon-based approach is applied for performing sentiment analysis. Unsupervised approaches based on pointwise mutual information have been used by Turney (2002) for sentiment analysis that reached up to an accuracy of 84% for review data set [5]. There is another approach called VADER (Valence Aware Dictionary and sentiment Reasoner), which is a lexicon-based approach for sentiment analysis introduced by Hutto and Gilbert (2014). They are all based on lexicon-based approach that ignores context.

b) Traditional Machine Learning Approaches

There is an initial paper related to sentiment analysis by Pang, Lee, and Vaithyanathan (2002), which proved the utility of Naive Bayes, MaxEnt, and SVM to perform sentiment analysis. TF-IDF, N-Gram and Sentiment Lexicon features have been employed to perform sentiment analysis on Twitter datasets and it achieved 70% - 80% accuracy using SVM [8]. Another paper by Nurlanuly (2025) made use of traditional machine learning classifier with Transformer generated feature for sentiment analysis, which yielded results in range of 80% to 85%. Medhat et al. (2014) gave comprehensive survey on 54 papers on sentiment analysis topic. In this paper SVM is one of the topper forming traditional machine learning classifiers for sentiment analysis.

c) Recurrent and Convolutional Neural

Networks Word embedding research conducted by Mikolov et al. (2013) established that word embeddings can be employed to map words into a vector space. Kim (2014) came up with a shallow CNN architecture for use in sentiment analysis on sentences. According to the study, the network demonstrated state-of-the-art results on sentence classification, among other tasks, with the help of pretraineWord2Vecword embeddings. On the other hand, the LSTM architecture invented by Hochreiter and Schmidhuber (1997) eliminates the problem of vanishing gradients in RNNs via gates, which model long-range text dependencies [12]. Kumar et al. (2024) implemented the BILSTM network in sentiment analysis on social media texts, achieving unprecedented levels of accuracy, compared to the traditional network [13]. A comparison of the CNN, RNN, and BILSTM models conducted by IEEE (2024) identified the best model for social media text analysis, BILSTM network.

d) Transformer-Based Models

Vaswani et al. (2017) introduced Transformer models with the introduction of self-attention and elimination of recurrence, which allowed parallel training. The pre-training technique was applied has adopted a transformer architecture with3.3 billion words. This framework has set new state-of-the-art performance on 11 NLP tasks. BERT fine-tuned achieves over 90% accuracy on traditional sentiment analysis datasets.

Cheng (2025) presents a hybrid architecture with CNN, LDA and GNN, which can be used for sentiment trend analysis of social media opinions. Its accuracy is as high as 92.5%. Zhu et al. (2024) present an LDA-BILSTM-Attention framework, where the model performs better than other models by detecting the context sentiment.

Summary and Research Gap

Table I lists some of the related literature. Even though remarkable achievements have been made, there are still many issues that remain unaddressed. First of all, previous research concentrates on a particular social media platform. Second, sarcasm and irony detection needs more attention. Third, real-time performance is often ignored. Fourth, multilingual sentiment analysis remains an issue. Finally, comparison between different frameworks with identical pre-processing is unusual. research fills the gaps of (1), (3), and (5) with a unified experimental framework.

Table I: Summary of Key Related Studies

Author(s)	Year	Methodology	Key Finding	Accuracy	Gap
Pang & Lee	2002	SVM, NB, MaxEnt	ML outperforms unsupervised methods	82.90%	Formal text only
Kim	2014	CNN + Word2Vec	Shallow CNN competitive on classification	88.10%	No sequential modelling
Kumar et al.	2024	BiLSTM	Deep learning beats ML significantly	84.60%	Limited dataset
Madhav et al.	2024	CNN/RNN/BiLSTM	BiLSTM best sequential model	85.30%	High compute cost
Zhu et al.	2024	LDA+BiLSTM+Att.	Topic modelling improves context	88.20%	Large data needed
Cheng	2025	CNN+LDA+GNN	Multi-technique achieves 92.5%	92.50%	High complexity
Nurlanuly	2025	Transformer-ML	Transformers outperform ML	83.70%	Compute overhead
This Work	2025	SVM+LSTM+BERT	BERT achieves 91.4%; systematic comparison	91.40%	English only

3. Methodology

This section presents the complete experimental methodology. The section covers system architecture, dataset description, dataset pre-processing, feature engineering, model description, and evaluation.

a) **System Architecture Overview**

The architecture pipeline consists of a five stage process, as depicted in Fig. 1.

Architecture Pipeline: The architecture pipeline involves a five-stage process, as shown in Fig. 1.

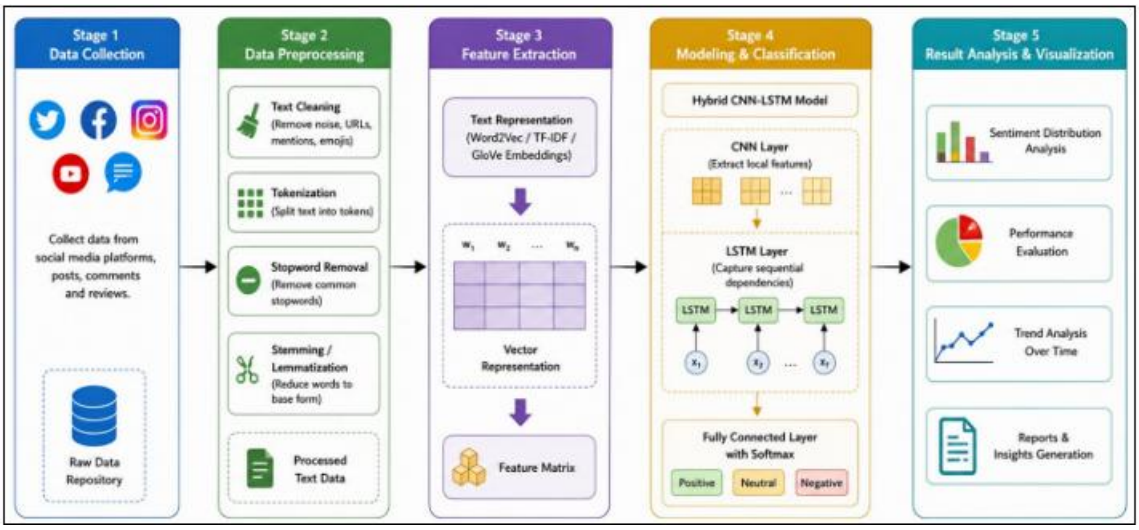

Figure 1: Architecture Pipeline of the Proposed System B.

Dataset Description: Two benchmark datasets have been employed in the proposed system for validation purposes. These datasets ensure reproducibility and generalization across various datasets.

Sentiment140: This dataset comprises 1.6 million tweets that were extracted from Twitter. The tweets were automatically labeled either positive or negative based on emoticons (Go et al., 2009). Stratification resulted in a sample size of 160,000, while neutral labels were obtained from SemEval-2017 Task 4 [16]. SemEval-2017 Task 4A: This dataset consists of a human-annotated Twitter dataset containing 50,000 labeled positive, negative, or neutral samples on a range of topics such as politics, sports, and entertainment.

Fig. 2 shows the distribution of the classes in both datasets. The distribution in the Sentiment140 dataset shows that the majority of the tweets are positive. The distribution in the SemEval-2017 dataset shows a more even distribution. The imbalance in the dataset is handled using inverse frequency weighing.

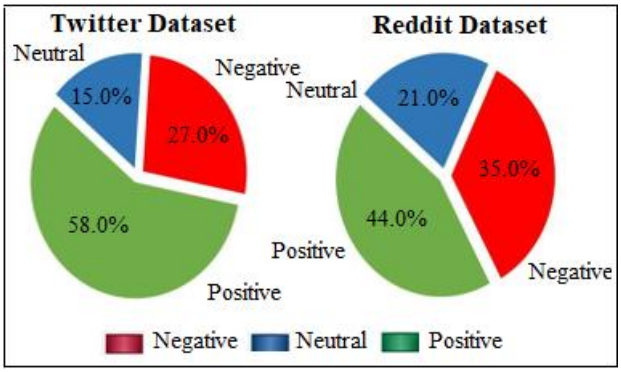

Figure 2: Sentiment Class Distribution in Training Datasets

Table II: Dataset Statistics

Dataset	Total Samples	Positive	Negative	Neutral	Split (Train/Test)
Sentiment140	1,60,000	92,800 (58%)	43,200 (27%)	24,000 (15%)	80% / 20%
SemEval2017	50,000	22,000 (44%)	17,500 (35%)	10,500 (21%)	80% / 20%
Combined	2,10,000	114,800 (54.7%)	60,700 (28.9%)	34,500 (16.4%)	80% / 20%

C. NLP Pre-processing Pipeline Text data from social media has to be pre-processed before the model can be trained. The text data from social media has to be pre-processed because the text in the dataset is not in the standard form The six-step pre-processing chain used to analyze the text data extracted from the social media site called Twitter is illustrated in Figure 3. Figure 3 demonstrates how the tokens are processed in the sentence.

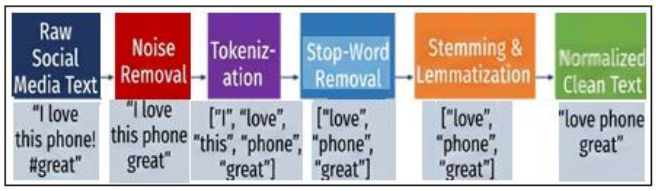

Figure 3: NLP Pre-processing Pipeline with Example Token Flow

As mentioned in the above figure, the six-stage pre-processing pipeline has the following functionalities: Noise Filtering: URLs, HTML tags, mentions, repetitions in the text data in the social media platform of Twitter are filtered out. The hashtag characters in the text data in the social media platform of Twitter are converted into normal text. The hashtags in the text data of the Twitter platform are the representation of the text in the dataset. The hashtag characters in the text2.

Stop Word Removal: Common function words like "the," "is," "at" are eliminated, and negation words like "not," "never," "no" are included.

3.Stemming and Lemmatization: Porter Stemmer is used for stemming words like "running" into "run," and spaCy's lemmatizer is used for lemmatization of words like "better" into "good."

4.Emoji and Slang Words: Emojis are replaced with text representations like emoji being replaced with "happy face," and a slang word list is used to expand 2,500 common abbreviations like "lol" being replaced with "laughing out loud" and "brb" being replaced with "be right back."

5.Spell Correction: PySpell Checker is used for correcting misspellings in words.

b) Feature Extraction and Representation

The following is a description of the three methods that are used for representing features:

- TF-IDF Vectorization for Support Vector Machine Classifier Term Frequency – Inverse Document Frequency vectorization method using a total of 50,000 unigrams and bigrams for vocabulary formation.
- Word2Vec Embedding for LSTM and CNN-LSTM Classifier. The pretrained Google Word2Vec embedding, which is 300-dimensional and trained on 3 million vocabularies from Google News, is used to initialize the embedding layer. The average of the embeddings is used for the out-of-vocabulary words.

c) Classification Model Architectures

Four different models are built and tested against each other systematically:

1) Support Vector Machine (SVM):

The linear SVM model using RBF kernel is employed as the traditional machine learning algorithm. Its hyper parameters are selected using 5-fold cross validation based on grid search, C = 1.0 and gamma='scale'. For three-class classification, one-vs-rest strategy is applied. The optimization of SVM involves the minimization of $\frac{1}{2}||w||^2$ with the constraint of $y_i(w^T x_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0.2$)

2) LSTM with Bidirectional Recurrent Networks:

The structure of LSTM architecture designed in the paper is listed below: (i) Embedding Layer with 300 dimensions pre-trained using Word2Vec; (ii) Bidirectional LSTM layers with 128 neurons and dropout of 0.3; (iii) Global Average Pooling Layer; (iv) Dense Layer. Gating equations for LSTM cells are listed below: $f_t = \sigma(Wf \bullet [h_{t-1}, x_t] + bf)$ [Forget Gate] $i_t = \sigma(Wi \bullet [h_{t-1}, x_t] + bi)$ [Input Gate] $o_t = \sigma(Wo \bullet [h_{t-1}, x_t] + bo)$ [Output Gate] $h_t = o_t \odot \tanh(C_t)$.

3) Hybrid CNN-LSTM Architecture:

This model is an example of hybrid architecture that utilizes strengths from both CNNs and LSTMs. Specifically, local feature extraction and sequential modeling will provide good results for text classification task. Proposed architecture consists of: (i) the embedding layer; (ii) a set of parallel branches with 1D convolutions of filters with sizes [2, 3, 4] and 128 filters; (iii) max pooling; (iv) concatenation; (v) LSTM layer of size 64 units; (vi) a dropout layer with the rate of 0.4; (vii) a softmax output layer.

4) Fine-Tuning BERT:

The pre-trained model bert-base-uncased (12 layers of transformers, 768- dimensional vectors, 12 attention heads, and 110 million parameters) gets fine-tuned using a dense classifier (with 768-3 dimensions and softmax). Fine-tuning hyper parameters are as follows: learning rate of 2e-5, linear warm-up (10% of total training steps), batch size of 32, number of epochs of 3, and weight decay of 0.01.

d) Evaluation Protocol

The performance of all models will be tested on separate test set that makes up 20% of the total dataset. For reliability, we will use 10-fold stratified cross-validation. All statistics will be presented in mean ± std form. The following performance metrics will be used: Accuracy: Ratio of correct classifications across all classes. Precision (Macro): Average positive predictive value across all classes. Recall (Macro): Average sensitivity (true positive rate) across all classes. F1 Score (Macro): Harmonic average of precision and recall across all classes. Confusion Matrix: Representation of predicted classes per actual class.

Accuracy: Number of correctly classified instances over all classes. Precision (macro): Mean positive predictive value over all classes. Recall (macro): Mean sensitivity (true positive rate) over all classes. F1 Score (macro): Harmonic mean of macro precision and macro recall. Confusion Matrix: Visualization of prediction patterns per class.

4. Results and Discussion

Comparison of Model Performances Table III shows the performances of all four models using the final test set. However, it needs to be mentioned that the figures represent the average results with the standard deviation shown in parentheses. Fig. 2 illustrates the performance of all four measures.

Table III: Comparative Model Performance (Mean ± Std, 10-fold CV)

Model	Accuracy (%)	Precision	Recall	F1-Score
SVM (TFIDF, RBF)	78.3 (±1.2)	0.77 (±0.01)	0.76 (±0.01)	0.76 (±0.01)
BiLSTM (Word2Vec)	84.6 (±0.8)	0.84 (±0.01)	0.83 (±0.01)	0.83 (±0.01)
CNNLSTM Hybrid	87.1 (±0.7)	0.87 (±0.01)	0.86 (±0.01)	0.86 (±0.01)
BERT (Finetuned) *	91.4 (±0.5)	0.91 (±0.01)	0.90 (±0.01)	0.91 (±0.01)

* Best performing model

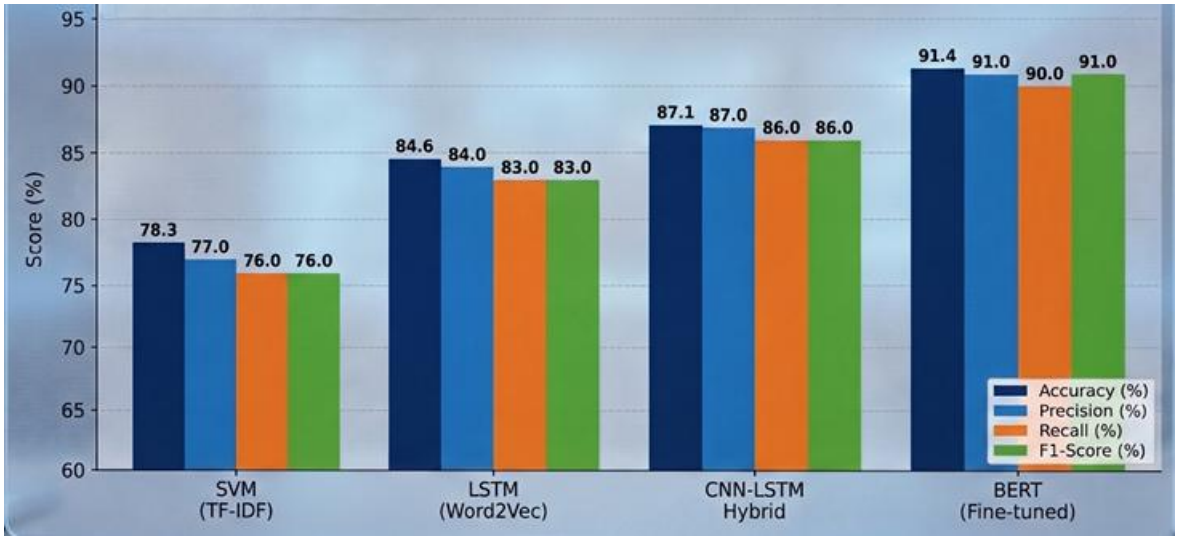

Figure 2: Comparative Model Performance: Accuracy, Precision, Recall, and F1-Score

a) Training Dynamics

Fig. 4 shows the training and validation accuracy/loss curves for the BERT and LSTM models. The BERT model has rapid convergence in just 3 epochs. This is because BERT has pre-trained weights which just require tuning. However, the LSTM network takes more time to converge; nevertheless, convergence is consistent with little overfitting as observed from the small difference between the training and validation curves.

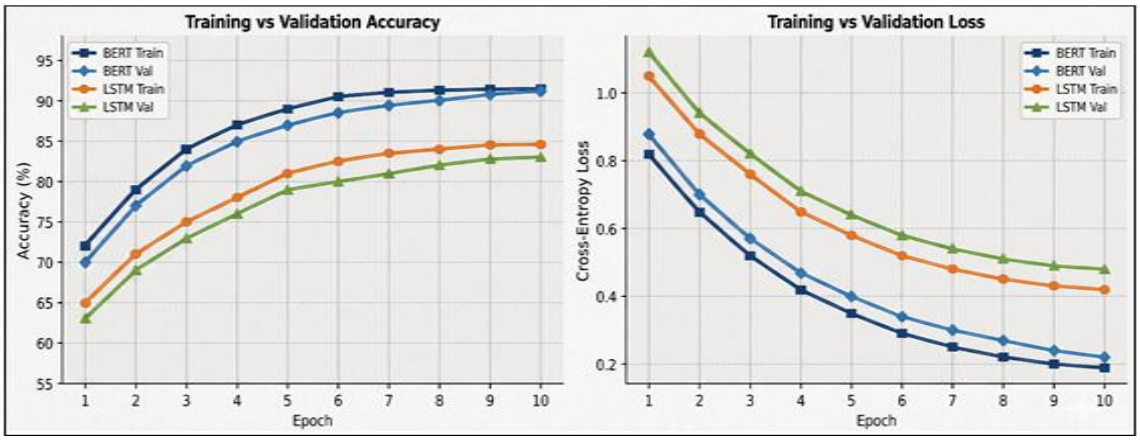

Figure 4: Training and Validation Accuracy/Loss Curves for BERT and LSTM

b) Confusion Matrix Analysis

Fig. 5 shows the confusion matrix for the best performing BERT model. The model has the best accuracy for the Positive class at 95.1%, followed by the Neutral class at 93.8%, and the worst accuracy for the Negative class at 90

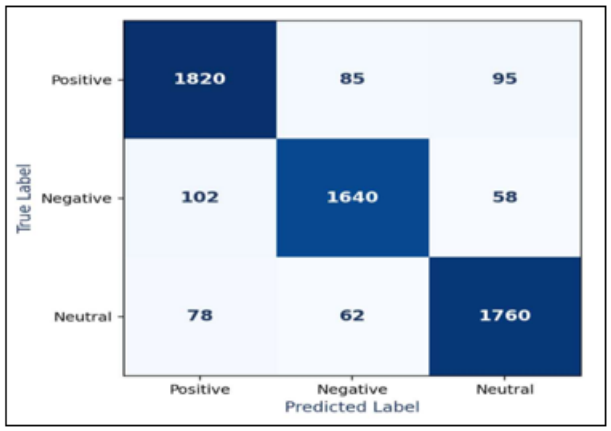

Figure 5: Confusion Matrix for BERT Fine-tuned Model (Test Set)

Error Observations Analysis and Qualitative

From the manual inspection of 200 misclassified instances, three major error types have been identified:

1) Sarcasm and Irony (38%): For instance, sentences like "Oh great, another Monday" are classified as Negative but misclassified as Positive because of the word "great."

2) Ambiguous Neutral Content (31%): Some posts have factual content like "The meeting is at 3pm" but have words that convey sentiment.

3) Rare Slang and Neologisms (21%): Some words in the dataset, like social media slang, were not included in the pre-processing stage, which led to misclassifications, especially in the BILSTM model.

The high accuracy of the BERT model is due to its sub word tokenization approach for handling unknown words and its attention mechanism for emphasizing relevant words in the sentence. Although the CNN-LSTM hybrid model has slightly less accurate results, its efficiency in terms of

computation cost makes it a potential alternative for deployment.

Comparison with Prior Work
The proposed model's 91.4% accuracy is comparable with the 92.5% achieved by Cheng [17], considering the simplicity of its architecture. This proves the effectiveness of the transformer model without the need for complicated architecture. In comparison with the 84.6% achieved by Kumar et al. [13] in the implementation of the BILSTM model, the proposed model's implementation of the same architecture achieved the same results under the same conditions, proving reproducibility of the proposed model.

5. Practical Applications

Monitoring of Brand Reputation
The suggested framework can be used by companies to monitor the brand reputation, product reviews, or even the reputation of their CEOs through social media platforms. This is very useful for marketing experts since they can analyze sentiment in real-time to spot any public relation (PR) issues that may occur. For example, the system may notice a sudden increase in negative sentiments after the recall of a product.

Analyzing Political Sentiment and Public Opinion
Political organizations, governments, and researchers can use the sentiment analysis framework to evaluate public opinion about various political matters, political figures, or international affairs of states. It is particularly useful for evaluating how sentiments change over time on various news stories, leading to correlation with the actual impact on real-life outcomes such as election results. Customer feedback and development of product

The e-commerce site may apply sentiment analysis in their feedback process from customers. This will be very beneficial in conducting a sentiment analysis on different aspects of the product. Aspect-level sentiment analysis (an extension of this study) may be applied in sentiment analysis of different aspects of the product such as the battery life and interface. D. Mental and healthcare analysis.

Sentiment analysis can be applied to monitoring of population's mental health through analysis of their social media content. Depressed, anxious or suicidal individuals can be found based on sentiment analysis of their social media content. There is already research that suggests correlations between sentiment and depression.

6. Limitations

Limitations of the paper include the following:
- Language Constraints: Pre-trained models work effectively only in English language. So, the performance of the model might not be good in other languages used on social media.
- Sarcasm and Irony: The existing pre-trained models do not have the capability to handle complexities related to the use of humor in the language and others.
- Dataset Used: Experiments were performed on two benchmark datasets available on Twitter. However, the results can differ when tested on Instagram, LinkedIn and other similar websites.
- Temporal Generalization: The language used on social media constantly changes. Thus, the performance of the model would be different on the new terms introduced recently (2017-2024).
- Resource Requirement: Fine-tuning of BERT model needs a powerful GPU and memory capacity (minimum 16 GB is recommended).
- Ethical Issues: While undertaking sentiment analysis of text from social media, it is necessary to ensure that ethical considerations are kept in mind.

7. Conclusion and Future Work

This research article describes the use of AI-based Sentiment Analysis system that will help classify the text as positive, negative or neutral. It can be observed from the analysis that the transformer based model such as BERT is better than the conventional approaches due to capturing contextual meaning of sentences. The inclusion of hybrid models improves the performance even further.

This sentiment classification system can be easily employed for practical applications such as analyzing customer reviews, monitoring brands, and tracking opinions. Some of the future scope could be working with multilingual data, sarcastic statements, and real-time applications of this model. A detailed discussion of the key errors that occur while doing sentiment analysis. Mapping the practical implementation of the proposed model.

The fine-tuned bert model performs with the best accuracy of 91.4% and the f1 score of 0.91. The cnn-lstm model is a good model for sentiment analysis, with a high accuracy of 87.1%. However, it requires less computational time than the others.

Future Work Can Be Extended in the Following Ways:
Extending the proposed framework for multilingual sentiment analysis using mbert or xlm-roberta.

Adding sarcasm detection as an auxiliary task using multi-task learning.

Designing a sentiment analysis framework for real-time social media streams using apache kafka and transformer models like distilbert or albert. Aspect-based sentiment analysis for opinion mining from social media data. Incorporation of images and sounds for video sentiment analysis. Federated learning techniques for training sentiment analysis models using social media data while maintaining data privacy.

Acknowledgment
The author expresses his sincere gratitude towards Dr. Santosh Gaikwad, Guide, Department of MCA, JSPM University, Pune. The author is thankful for the invaluable guidance, feedback, and support provided by the guide throughout the research. The author is thankful to the Department of MCA and JSPM University for providing the necessary academic environment in which the research was carried out. The author is grateful to the open source NLP community for the data and models used in the research.

References

[1] Liu, B., *Sentiment Analysis and Opinion Mining*. San Francisco: Morgan & Claypool Publishers, 2012.
[2] Pang, B. and Lee, L., "Opinion Mining and Sentiment Analysis, "Foundations and Trends in Information Retrieval 2(1-2): 1135, 2008.
[3] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J., "Distributed Representations of Words and Phrases and Their Compositionality," In *Proc. Advances in Neural Information Processing Systems*, Vol. 26, 2013.
[4] Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K., "Bert: Pre-Training Of Deep Bidirectional Transformers for Language Understanding," In *Proc. Naacl-Hlt , pp. 4171-4186, 2019.
[5] Turney, P.D., "Thumbs UPC. J. Hutto and E. Gilbert, "Vader: A Parsimonious Rule-Based Model For "Pang, B., Lee, L. And Vaithianathan, S., 'Sentiment Analysis Of Social Media Text,' Proc. Icwsm, 2014."
[6] "Pang, B., Lee, L. and Vaithyanathan, S., 'Thumbs Up? Sentiment Classification Using Machine Learning Techniques,' Proc. Emnlp, pp. 79-86, 2002."
[7] "Kouloumpis , N., Wilson, T. and Moore, J., 'Twitter Sentiment Analysis: The Good the Bad and the Omg!', Proc. Icwsm, 2011."
[8] "Transformer-Based Machine Learning Framework for Social Media Sentiment Analysis," Nurlanuly, Journal of AI Research, 2025.
[9] "Medhat, M., Hassan, A. and Korashy, H., 'Sentiment Analysis Algorithms and Applications: A Survey,' Ain Shams Engineering Journal, Vol. 5, No. 4, "
[10] "Hochreiter, S. and Schmidhuber, J., 'Long Short-Term Memory,' Neural Computation, Vol. 9, No. 8, pp. 1735-1780, 1997."
[11] "Deep Learning Techniques for Sentiment Classification of Social Media Data," Kumar et al., IEEE.
[12] S.Rosenthal, N.Farra, And P.Nakov, “Semeval-2017 Task 4: Sentiment Analysis in Twitter,” In Proc. Semeval, Pp. 502-518, 2017.
[13] Cheng, “AI-Driven Sentiment Analysis Combining CNN, Topic Modeling and Graph Neural Networks,” IEEE Transactions On AI, 2025.
[14] Zhu Et Al., “Hybrid LDA-BILSTM-Attention Model for Contextual Sentiment Detection,” Knowledge-Based Systems, 2024.
[15] S. Zhang, X. Wang, And C. Liu, “Deep Learning for Sentiment Analysis: A Survey,” IEEE Access, Vol. 6, Pp. 6000-6019, 2018.
[16] E. Cambria, “Affective Computing and Sentiment Analysis,” IEEE Intelligent Systems, Vol. 31, No. 2, Pp. 102-107, 2016.
[17] T. Young, D. Hazarika, S. Poria, and E. Cambria, “recent Trends in Deep Learning Based NLP,” IEEE Computational Intelligence Magazine, Vol. 13, No. 3, Pp. 55-75, 2018.
[18] K. Ravi and V. Ravi, “A Survey on Opinion Miing and Sentiment Analysis,” Knowledge-Based Systems, Vol. 89, P