International Journal of Data Informatics and Intelligent Computing (IJDIIC)
Vol. 3, No. 4, December 2024, pp. 62~76
DOI: 10.59461/ijdiic.v3i4.146

Contextual Analysis of Immoral Social Media Posts Using Self-attention-based Transformer Model

Bibi Saqia[1], Khairullah Khan[1], Atta Ur Rahman[2], Wahab Khan[1]
[1]Department of Computer Science, University of Science and Technology, Bannu, 28100, Pakistan
[2]IRC for Finance and Digital Economy, KFUPM Business School, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia

Article Info

Article history:

Received October 09, 2024
Revised November 27, 2024
Accepted December 16, 2024

Keywords:

Immoral content
Contextual Analysis
Social media
NLP
Transformer model

ABSTRACT

Immoral posts detection on social media is a serious issue in this digital era. This matter wants advanced natural language processing (NLP) methods to address user-generated text's difficult semantics and context. Incorporating advanced deep learning (DL) techniques improves the model's aptitude to handle challenges such as slang, sarcasm, and vague expressions. This work suggests a deep contextual analysis framework using a self-attention-based transformer model to detect immoral contents on soil networks efficiently. The model captures complex contextual associations and semantic nuances by harnessing the strength of self-attention mechanisms. The proposed technique enables proper differentiation between moral and immoral content. The framework is assessed on two benchmark datasets, SARC and HatEval. The experiment shows the highest F1-score, 98.10%, on the SARC dataset. While on HatEval, the model achieved 97.34%, representing greater performance than state-of-the-art approaches. The results highlight the efficiency of self-attention-based DL models in delivering efficient, scalable, and ethical answers for observing and modifying harmful content on social media networks.

Corresponding Author: Bibi Saqia(e-mail: saqiaktk@ustb.edu.pk)

1. INTRODUCTION

The quick spread of social media platforms has changed how people connect and share sentiments. Sharing opinions and information worldwide using social media is as simple as tapping the purpose of just a single-click button on a digital device [1]. Numerous industries have used web applications to improve personality traits, interact with clients, staff engagement, and market products [2]. Consequently, this change has also enabled the spread of immoral content [3]. These contents involve hate speech, sarcasm, and misinformation, which pose considerable challenges to digital ethics and societal harmony [4]. Detecting such hurtful content is a demanding concern [5]. This undesirable content requires robust, scalable, and advanced DL capable of understanding the sophisticated nuances of web data [6]. Social media posts are often casually riddled with slang [7], sarcasm [8], abbreviations, and fluctuating contextual semantics. Detecting inappropriate queries automatically can be difficult because slang, sarcasm, or ambiguous phrases are common on social media [9]. Table 1 represents immoral instances, such as the "*shut up!"* influence, which sounds impolite but is often used playfully among friends. Similarly, *"I'm dead"* might seem shocking but typically means something is very funny. A sample like *"she's so bad"* could appear negative but often denotes someone being good-looking or confident. In another instance, *"go kill it"* might refer to violence but means to do something very thriving. Finally, *"this is trash"* might appear aggressive but is often used informally to express dislike.

The complex nature of social media posts makes traditional content analysis attitudes insufficient [10]. Current improvements in machine learning (ML) have provided influential tools to investigate and process textual data [11]. Earlier studies concentrated on detecting offensive terms and grammatical expressions.

Supervised ML classification approaches, such as naive Bayes, support vector machines (SVM), and random forests (RF) have been employed for social media content analysis. These approaches depend on manual feature engineering, which is both prone to human bias and time-consuming, making them less flexible to the evolving web content [12][13][14][15]. Yet, such methods are inappropriate for capturing deep contextual and semantic features. Contextual study is essential for precisely recognizing depraved content, particularly when dealing with subtle linguistic cues and intricate social contexts. Unethical content or vulgar remarks as a societal tricky is an old area of research. The increase in anti-social attitudes in online spaces has attracted the attention of academics [16]. Therefore, it is necessary to update scholars regularly on the most recent advances. The methods from traditional ML, DL, and ensemble approaches in identifying hate speech in social networking sites[17].

Table 1. Inappropriate/immoral queries and their misinterpretations

Actual Query	Reason for Inappropriateness/Immorality	Alternative Interpretation
Shut up!	Sounds rude or disrespectful.	Commonly used in a playful or joking manner.
I'm dead	This could be misinterpreted as alarming	Slang for finding something extremely funny.
She's so bad	It might sound negative or insulting.	Often means someone is attractive or confident.
Go kill it	It may sound violent or aggressive.	Encouragement to perform exceptionally well.
This is trash	It could appear offensive or harsh.	Casual expression to indicate dislike or criticism.

The proposed work addresses these limitations by presenting a self-attention-based transformer model. The suggested model is capable of capturing complex contextual relations and semantic nuances. Moreover, the proposed framework assimilates progressive NLP preprocessing methods, which assists in reporting the challenges posed through informal, vague, and culturally definite language on social networks. Unlike prior models, this method does not depend on manual feature engineering and can automatically adjust to the dynamic nature of social media posts. Furthermore, the suggested model is assessed on benchmark datasets, representing higher performance than current approaches. With these improvements, this work deals with a different technique that not only advances the technical competencies in perceiving immoral content. But also reports the ethical worries that follow such tools.

1.1. Contributions

The proposed study employed a novel deep contextual analysis structure leveraging a self-attention-based transformer model to detect unwanted web contents efficiently. The proposed model incorporated advanced NLP preprocessing methods and contextual embeddings to address issues such as ambiguous expressions, sarcasm, and the informal nature of social media language. Benchmark datasets SARC and HatEval are utilized to assess the model, presenting its superior outcomes over current techniques.

- This work suggests a self-attention-based transformer model tailored for perceiving immoral social media posts, concentrating on deep contextual understanding and semantic evaluation.
- The framework incorporates refined preprocessing and embedding methods to improve the model's aptitude to process ambiguous, informal, and sarcastic linguistics.
- Widespread trials on benchmark datasets certify the framework's efficiency, representing enhancements over state-of-the-art approaches in identifying immoral content.
- The suggested model delivers a scalable and ethical method for observing and alleviating destructive content, addressing the developing challenges of digital control on social podiums.

1.2. Paper Organization

The remainder of this paper is structured as follows: Section 2 discusses the literature review of the proposed work. Section 3 defines the proposed methodology. Section 4 indicates the experiments conducted to evaluate the proposed study. Section 5 describes the results of the proposed model and compares it with other start-of-the-art approaches. Section 6 presents the conclusion of the proposed work and its future direction.

2. LITERATURE REVIEW

This section reviews relevant studies, categorizing them into traditional approaches, ML-based algorithms, and Transformer-based approaches to deliver a detailed understanding of the domain and explore the research gaps. The immoral content detection on social media has attained considerable attention current era. Different techniques have been explored in the literature employing improvements in NLP [18] and ML [19] to address the challenges of evaluating web content [20].

2.1. Traditional Techniques for Content Analysis

Early research focused on detecting immoral posts relied heavily on rule-based and lexicon-based methods. The work published in [21] concentrated on the automatic annotation of vulgar and offensive language on web content. Their proposed usefulness of computational methods for detecting taboo content online. The outcome of their work is a corpus of 31,749 Facebook remarks, which have been automatically annotated using a lexicon-based technique to identify offensive expressions. The study employed in [22] the repute check, which identifies and debunks rumours related to global celebrities and politicians on Twitter. By using a lexicon for derogatory terms and extracting features from texts and user accounts. The model reached 83.4% accuracy with an SVM. These consequences highlight the vulnerability of public data to signify that rumour detection can assist in curbing the spread of harmful online texts. The work performed in [23] presents a fuzzy rule-based system for sentiment assessment of social media posts combining different lexicons to sentiment into positive, negative, or neutral. Their outcomes outperform current techniques on nine Twitter datasets, providing flexibility and intuition into the best lexicon selection. These approaches applied manually crafted rules to flag harmful content. They employed a rule-based approach to detect irony, insults, and offensive terms in text [24]. However, these techniques were interpretable and straightforward yet suffered from considerable limitations. The main challenges include poor scalability and an inability to control complicated linguistic phenomena such as slang, sarcasm, or context-dependent meanings.

2.2. Machine Learning-Based Methods

The researchers began to use supervised learning approaches for unethical content identification with the advent of ML [25]. The most noticeable ML classifiers, such as SVM, Naïve Bayes, and Decision Trees, were widely employed in NLP tasks [26]. These techniques relied on feature engineering, requiring domain expertise to craft features like sentiment scores, syntactic patterns, and word frequency [27][28]. The work accomplished by [29] employed different methods for sarcasm detection, highlighting preprocessing methods and assessing diverse ML classifiers. The research performed in [30] reported the issues of sarcasm identification in tweets, like disregarding context and dealing with scarce data, via suggesting a Multi-feature fusion structure. This outline utilized a two-stage organization method, merging lexical features with contextual data to increase estimate accuracy. Experimental effects revealed the context's efficiency, reaching a precision of 0.947 with an RF classifier and outperforming baseline approaches.

2.3. Deep learnings

The work published by [31] employed an advanced DL model to perceive irony and sarcasm by capturing subtle signs, contextual dependences, and sentiment changes. They influence transfer learning from large language models and assimilate multimodal data containing images and emojis. Real-world social media content demonstrates the models' effectiveness using their proposed technique, which is evaluated on benchmark datasets. Their experimental results show an important development in understanding social media dynamics and opinion mining. The study accomplished by [6] represents the optimal outcomes of DL models. They accurately identified cyberbullying and offensive content across different tasks using Bidirectional Long Short-Term Memory (BiLSTM). Their model's effectiveness in detecting online hate speech has been evaluated by graphical representation and confusion matrices. Their proposed consequences focus on essential advanced neural networks that notice the difficulties of cyberbullying in online communities. The work performed in [32] focused on identifying cyberbullying to facilitate timely involvement. To improve cyberbullying identification, they recommend a fine-tuned pre-trained sentence transformer language model. Different experimental work has been accomplished on three datasets with better-quality presentation than state-of-the-art outcomes. Their techniques deal with latent for clarifying unethical messages, recognizing affected individuals, and assisting involvement of strategies to battle cyberbullying. Different DL models like recurrent neural networks (RNNs), convolutional neural networks (CNNs), and LSTM were applied to various kinds of immoral content on the web. These models represent considerable outcomes in the identification of hate speech and offensive comments on social media platforms [33] [34]. Still, these models often efforts with contextual nuances and long-range dependencies, confining their usefulness for complex social network texts.

3. METHODOLOGY

In this section, we describe the overall methodology of the proposed work. The suggested work concentrates on developing a self-attention-based transformer model for the contextual examination of immoral social media content. Ethical matters such as slang, scorn, sarcasm, and confusing linguistics are addressed by assimilating self-attention mechanisms and multi-head attention layers. This work assists the model in perceiving serious contextual dealings and semantic nuances. Figure 1 shows the graphical representation of the proposed self-attention-based transformer model for the contextual analysis of web content.

3.1. Dataset Description

In the proposed study, we employed two benchmark datasets, Self-Annotated Reddit Corpus [35] and HatEval [36] to evaluate inappropriate content like sarcasm and hate speech on social media platforms. The datasets are divided into training and testing classes by 80:20 to efficiently evaluate the proposed model. Table 2 defines the statistics of the dataset used in the proposed study.

3.2. Preprocessing Steps

The preprocessing of the SARC and HatEval datasets consists of different steps to certify capability with the proposed self-attention-based transformer model. The procedure initiates with text cleaning, eliminating URLs, hashtags, emojis, and special characters to confirm only the textual content is reserved (e.g., "Check this out! 😊 https://example.com #funny" became "Check this out funny"). The text is then consistent via lowercasing all characters, monitored by stopword removal to remove common but irrelevant terms, like "this" and "is." etc. Next, we employed tokenization to divide sentences into separate tokens and achieved lemmatization to decrease words to their root forms. For instance, "The posts are unethical" is transformed into "post unethical." Labels such as "ethical" and "unethical" are encrypted numerically as 0 and 1, correspondingly. Furthermore, context accumulation augmented the textual data by comprising user-interacted remarks from SARC, permitting the model to better capture the proposed meaning. We applied padding and truncation to regulate input orders to a stable span for compatibility with the transformer model. Table 3 represents the preprocessing steps of the SARC and HatEval datasets.

3.3. Self-Attention Mechanism

The self-attention mechanism detects associations between various features within the sequence and allocates every attribute a mass centred on its applicability to other attributes [37]. This appliance allows the model to emphasize important contextual and semantic features of the posts. It focuses on related parts of the input sequence, regardless of their location, allowing a nuanced understanding of the content. The features and contextual elements of web contents are signified by an order of input attributes that scramble each occurrence in the dataset. In the proposed work, we employed the input sequence, as shown in Eq. 1.

$$A = a_1, a_2, \dots, a_n \tag{1}$$

Where $a \in R^d$ denotes the input illustration of the i^{th} word in a post, n is the sequence length, and d is the embedding dimension. These representations are processed by the self-attention layers to calculate attention weights that classify the position of each token relative to others in the sequence. In this construction, the order of input tokens is considered as represented in Eq. 2.

$$\text{Attention}(\text{Q}, \text{K}, \text{V}) = \text{softmax}\left(\frac{\text{QK}^{\text{T}}}{\sqrt{\text{d}_\text{k}}}\right)\text{V} \tag{2}$$

Here, Q, K, and V are the query, key, and value matrices derived from E, and d_k is the dimensionality of the key vectors.

3.4. Multi-Head Attention

To capture different kinds of connections and improve productivity, numerous parallel self-attention layers (multi-head attention) are applied. The attention layers are handled over a feed-forward neural network to detect non-linear associations and deliver the final forecasts. This building, merging self-attention and multi-head attention mechanisms, efficiently captures links and dependencies within the input sequence. By doing so, the model emphasizes critical traits like nuanced semantics and context. Multi-head attention syndicates attention from multiple perceptions, as shown in Eq. 3.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \quad (3)$$

Table 2. Statistics of datasets

Dataset	Purpose	Instances	Additional Context
SARC	Sarcasm detection	57,000	Includes user-interacted comments
	Training set (80% of total)	45,600	Used for model learning
	Testing set (20% of total)	11,400	Used for model evaluation
HatEval	Hate speech detection	6,393	Focus on offensive and hate
	Training set (80% of total)	5,114	Used for model learning
	Testing set (20% of total)	1,279	Used for model evaluation

Table 3. Preprocessing steps

Step	Description	Example Sentence Before Preprocessing	Example Sentence After Preprocessing
Text Cleaning	Removed URLs, emojis, hashtags, mentions, and special characters to focus on the core text content.	"Check this out! 😊 https://example.com #funny"	"Check this out funny"
Lowercasing	Converted all text to lowercase to standardize input.	"THIS IS IMMORAL!"	"This is immoral!"
Stopword Removal	Eliminated common stopwords using the NLTK library to retain meaningful information.	"This is an example of a moral post."	"example moral post"
Tokenization	Split sentences into tokens using the WordPiece tokenizer.	"moral post example"	["moral", "post", "example"]
Lemmatization	Reduced words to their base forms for semantic uniformity.	"The posts were immoral."	"post immoral"
Label Encoding	Converted categorical labels (moral/immoral, sarcastic/non-sarcastic) into numerical values.	"Moral" / "Immoral"	0 / 1
Context Aggregation	Incorporated user-interacted comments from SARC to enrich contextual understanding.	Original comment: "That's funny." Context: "No sarcasm."	"That's funny. No sarcasm."
Padding and Truncation	Adjusted text length to a fixed size suitable for transformer input (e.g., 512 tokens).	"Short sentence."	"Short sentence <PAD> <PAD>"

Whereas the input query matrix (indicates the input sequence as queries) by Q, the key matrix (denotes the input order as keys) by K. Similarly the value matrix (shows the input sequence as values) by V. The results of all attention heads $(head_1, \ldots, head_h)$ are concatenated into a single matrix $Concat$. Similarly, the individual attention head calculates attention independently, where the learnable weight matrices are denoted by described in Eq. 4.

$$\text{head}_i = \text{Attention}\left(QW_i^Q,\ KW_i^K,\ VW_i^V\right), \text{and } W_i^Q,\ W_i^K, W_i^V \text{ and } W^O \quad (4)$$

Whereas $W_i^Q, W_i^K, W_i^V and\ W^O$ are learnable matrices, permitting the model to adjust and extract useful patterns during training. These are learnable weight matrices that project Q, K, V into lower-dimensional spaces for individual heads. The multi-head attention mechanism enables the model to emphasize various parts of the input sequence concurrently, refining its aptitude to capture complex relations.

3.5. Positional Encoding

Positional encoding is a method utilized in transformer models to integrate the sequential command of input data. It allocates distinct embeddings to every position in the input arrangement, permitting the model to capture the positional relations between attributes. Figure 2 represents the working procedure of positional

encoding. These embeddings are further to the input word embeddings, aiding the model in distinguishing between words based on their positions. Usually, sinusoidal functions, like sine and cosine, are employed to make these encodings, certifying smooth positional changes. This mechanism improves the model's aptitude to practice sequential data efficiently, such as text in NLP tasks, as described in Eq. 5.

$$\mathrm{PE}_{(\mathrm{pos},2\mathrm{i})} = \mathrm{sig}\left(\frac{\mathrm{pos}}{10000^{2\mathrm{i}/\mathrm{d}}}\right) \tag{5}$$

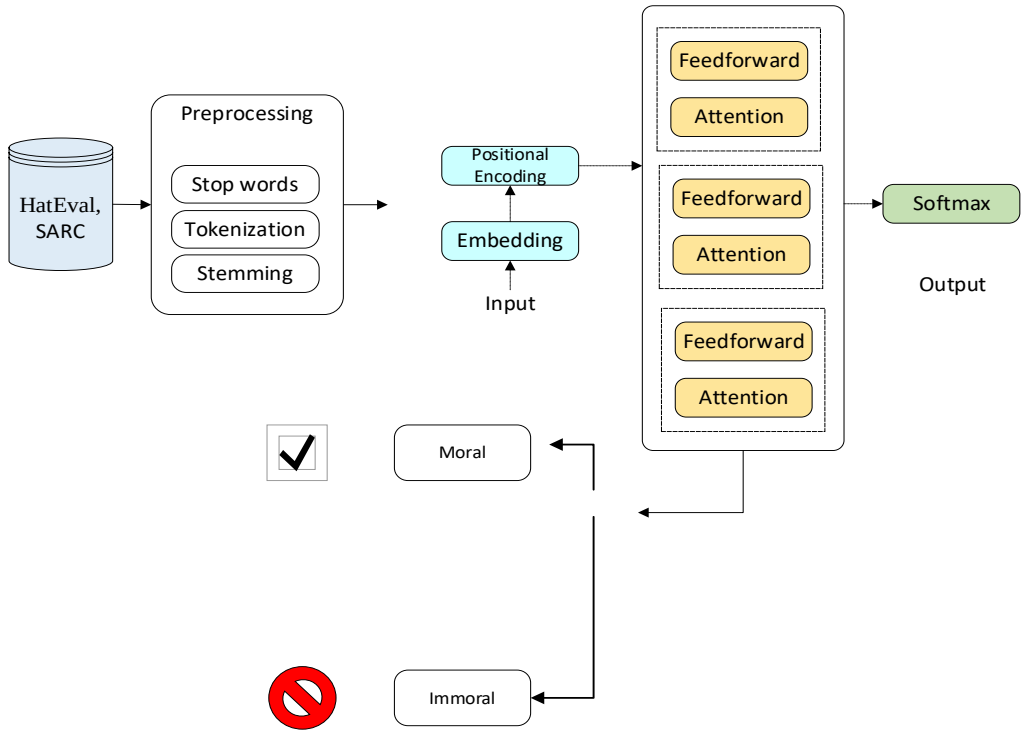

Figure 1. Proposed methodology steps

In the above equation, the sine element of the positional encoding is employed in transformer models. Positional encoding delivers a method to integrate the instruction of tokens into the model, as transformers lack an inherent logic of sequence. In this formulation, pos denotes the position of a token within the input order, i denotes the index of a specific dimension in the positional encoding vector, and d signifies the entire dimensionality of the encoding. The sine function is applied, with a scaling factor of $10000^{2i/d}$ that fine-tunes the occurrence of the sine wave across dimensions for even dimensions. This scaling certifies that each dimension captures positional data at fluctuating granularities. The sine values make periodic patterns, permitting the model to differentiate tokens based on their position in the sequence. The transformer can encode both local and global positional associations efficiently through integration with cosine encodings for odd dimensions. This mechanism permits the model to process sequential data without trusting repetition or convolution, increasing its capability to understand and operate the order of tokens.

The cosine element of the positional encoding in transformer models. Positional encoding is employed to deliver information regarding the order of tokens in a sequence, as transformers procedure input tokens instantaneously without an inherent sense of sequence in Eq. 6.

$$\mathrm{PE}_{(\mathrm{pos},2\mathrm{i}+1)} = \cos\left(\frac{\mathrm{pos}}{10000^{2\mathrm{i}/\mathrm{d}}}\right) \tag{6}$$

The above equation pos denotes the position of a token in the sequence, i is the index of the dimension being intended, and d is the entire dimensionality of the encoding. The formula relates the cosine function for odd dimensions with a scaling factor $of\ 10000^{2i/d}$. This scaling regulates the incidence of the cosine wave for

each dimension, certifying that positional encodings vary easily and distinctively across various dimensions. By merging sine and cosine functions for even and odd dimensions correspondingly, the transformer makes distinctive positional encodings for individual tokens in order. These encodings permit the model to capture both local and global positional associations efficiently, allowing it to take the sequence directive together with the token embeddings. This method helps the model handle sequential data without relying on traditional sequence-based mechanisms such as reappearance.

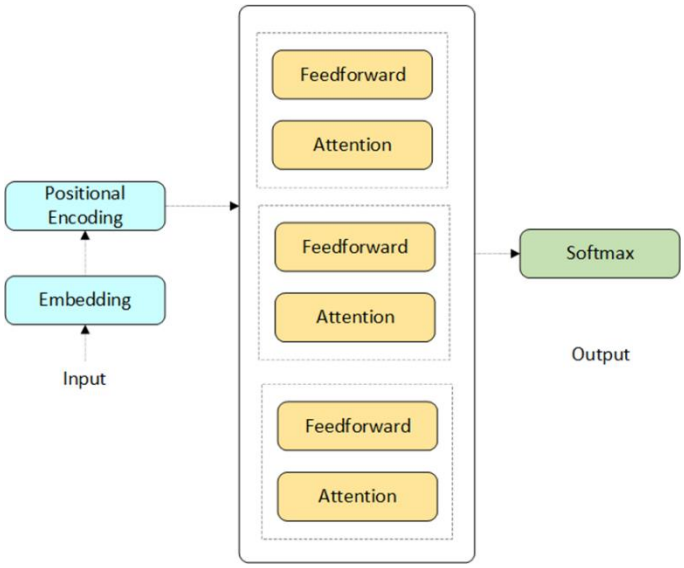

Figure 2. The working procedure of positional encoding

3.6. Transformer Encoder

The Transformer encoder is a key element of the transformer architecture designed for sequence-to-sequence activities. It processes input sequences by several layers, each containing two main sub-layers: a feed-forward neural network and a multi-head self-attention mechanism. The self-attention mechanism helps the encoder to perceive dependencies between all the features of the input sequence without considering their distance. The input embeddings retain order information using positional encoding. Each sub-layer is followed by layer normalization and residual connections to enhance learning stability. The encoder relates self-attention and feed-forward layers for the encoder output, as shown in Eq. 7.

$$\mathrm{Z} = \mathrm{LayerNorm}\big(\mathrm{E} + \mathrm{MultiHead}(\mathrm{Q}, \mathrm{K}, \mathrm{V})\big) \tag{7}$$

The above equation defines the main step in transformers. The multi-head attention mechanism is represented by $MultiHead(Q, K, V)$ by capturing associations between tokens by directing on different shares of the input sequence. The outcome is further related to the input embeddings E over a residual joining, stabilizing and aiding gradient flow and factual information during training. Lastly, layer normalization standardizes the joined production, certifying reliable scaling and refining model convergence and stability. This procedure improves the model's capability to learn operative sequence demonstrations proficiently, as described in Eq. 8.

$$\mathrm{O} = \mathrm{LayerNorm}\big(\mathrm{Z} + \mathrm{FFN}(\mathrm{Z})\big) \tag{8}$$

The above equation signifies another vital step in transformer models. Here, Z represents the output from the multi-head attention and layer normalization phase. It is delivered via a feed-forward network denoted by FFN, which uses two linear transformations with a non-linear activation in between, improving the model's ability to learn complex patterns. The output FFN is added back to Z using a residual connection, preservative of the unique data, and refining gradient flow. Lastly, layer standardization is useful to the joint output, confirming stability and well-organized training by normalizing the values across dimensions. This process improves the sequence depictions.

3.7. Classification Layer

The classification layer in the self-attention-based transformer model is the last phase in the structure and is responsible for creating predictions. It processes the refined feature structures achieved from the preceding layers, containing self-attention and feed-forward networks. A dense layer is used to transform these attributes into a smaller dimensional space corresponding to the target classes ("moral" and "immoral"). The softmax activation function is then utilized to allocate probabilities to every class. This layer successfully captures the nuanced contextual and semantic data, enabling the model to make concise and reliable predictions for detecting unethical social media content. The last encoder output is handed to a classification layer, as shown in Eq. 9.

$$y = \text{softmax}(W_c O + b_c) \tag{9}$$

Where the learnable weight and bias parameters are represented by W_c and b_c, respectively, to signify the final classification stage in the transformer model, the production from the earlier layer is denoted by O. The linear transformation $W_c O + b_c$ maps the model's output to the preferred number of classes. The softmax function then changes these standards into chances, conveying a likelihood for each group. This stage permits the model to make forecasts by choosing the class with the maximum probability.

3.8. Loss Function

In the proposed work, the cross-entropy loss function is used to optimize the model during training. Cross-entropy loss is appropriate for classification activities as it evaluates the divergence between the true class labels and the predicted probability distribution. the model enhances its ability to categorize posts accurately by minimizing this loss. The cross-entropy loss for a binary classification work is defined in Eq. 10.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{ij}\log(\hat{y}_{ij}) \tag{10}$$

Where the amount of samples is denoted by N, the quantity of classes is represented by C, the true label is presented by y_{ij} (if the class is correct then 1, otherwise 0), and the forecast probability is denoted by $\hat{y}_{ij}$ for class j. The formula computes the average negative log-likelihood of the forecast probabilities for the correct labels, penalizing incorrect calculations and inspiring the model to increase its classification accuracy.

4. EXPERIMENTS

In this section, the entire experiments with the training process, experimental setup, and evaluation criteria are discussed.

4.1. Training Process

The training process for the proposed Self-Attention-Based Transformer Model begins with data collection from benchmark datasets, specifically SARC and HatEval, containing labelled social media posts categorized as moral or immoral. Initially, the datasets (SARC and HatEval) were divided into training and testing sets using an 80:20 split, with the training set comprising 45,600 samples for SARC and 5,114 samples for HatEval. Preprocessing is applied to clean and prepare the data by removing noise, handling missing values, and normalizing text to ensure consistency. The cleaned dataset is tokenized, and positional encodings are added to represent the sequence order of words within the posts. These inputs are passed through the self-attention mechanism, where relationships between contextual and semantic elements are identified, and weights are assigned based on their relevance. Multi-head attention layers further enhance the model's ability to capture diverse interactions, while a feed-forward neural network processes the attention outputs to detect non-linear relationships. Finally, the model undergoes iterative training and validation, with optimization techniques employed to minimize loss and improve performance. This process results in a robust model capable of accurately identifying immoral social media posts.

4.2. Experimental Setup

The experimental procedure was methodically calculated to assess the performance of the planned self-attention-based transformer model in noticing immoral social media content. This procedure involved choosing the best model parameters, conducting various trials, and examining results using both datasets

(SARC and HatEval). The experiments have been performed in an organized setting by the hyperparameters and a proper setup. The trials leveraged a plain parameter selection process to balance computational competence and performance. The learning rate is set between 0.001 and 0.01, permitting stable convergence during training. The Adam optimizer is applied for its adaptive learning rate competencies, certifying rapid growth and minimalizing the loss function. The batch size is different from 32 to 128, which optimizes GPU operation and prohibits memory overflow. To certify suitable training, the model is trained for 120 epochs, with early stopping employed to avoid overfitting. The embedding layer applied 300-dimensional vectors produced by pre-trained embeddings such as Word2Vec and GloVe, apprehending rich semantic data. To alleviate overfitting, a dropout rate of 0.3 is combined into the model construction. The transformer model included four encoder layers, with each layer requiring eight attention heads. The ReLU activation function is used to present non-linearities in the system. Model presentation is assessed using key metrics, counting accuracy, precision, recall, and F1-score, confirming a broad valuation of the consequences. Overall, experimental work has been performed on a system equipped with 16 GB RAM, an Intel Core i7 processor, and an NVIDIA GeForce GTX GPU, which enabled well-organized calculation during training and implication. This thorough experimental procedure and parameter configuration consider the methodical method approved to accomplish precise and consistent classification of unwanted and inappropriate content on social networks. Table 4 indicates the model parameters applied in the suggested model of immoral contents.

Table 4. Model parameters

Parameter	Value	Description
Learning Rate	0.001–0.01	Computes the step size for parameter updates during training.
Optimizer	Adam	Certifies adaptive learning rates for faster and more stable convergence.
Batch Size	32–128	Number of samples processed before updating the model weights.
Epochs	16–64	Number of complete passes through the training dataset.
Embedding Dimensions	300	Size of word vectors representing input text.
Dropout Rate	0.3	The fraction of units dropped to prevent overfitting.
Activation Function	ReLU	Applied in hidden layers for non-linear transformations.
Number of Heads	8	Number of attention heads in the multi-head attention mechanism.
Number of Layers	4	Transformer encoder layers are stacked in the model.
Evaluation Metrics	Accuracy, Precision, Recall, F1, Kappa	Measures model performance and agreement between true and predicted labels.
Hardware Configuration	Intel Core i7, 16 GB RAM, NVIDIA GeForce GTX GPU	Computational environment used for training.

4.3. Evaluation Criteria

The evaluation of the suggested study is carried out by an inclusive set of performance metrics to certify a detailed assessment of its capability to categorize unethical social media content correctly. The following metrics and standards are applied:

Accuracy is the proportion of properly forecast cases to the entire number of occurrences:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{11}$$

Here, TP, TN, FP, and FN signify true positives, true negatives, false positives, and false negatives.

Precision counts the model's exactitude in expecting inappropriate contents and is computed as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{12}$$

It specifies the amount of properly known immoral content out of all posts expected as depraved. An advanced precision value means rarer unrelated posts are misclassified as unethical.

Recall measures the model's aptitude to recognize immoral contents:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{13}$$

It redirects the number of decadent posts properly recognized out of all actual depraved posts. A great recall worth displays the model is efficiently classifying depraved content.

The harmonic mean of precision and recall equilibriums both standards:

$$\text{F1} - \text{score} = 2 . \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

It delivers a single presentation measure, mainly valuable when the dataset is imbalanced, certifying that both precision and recall are measured equally.

5. RESULTS AND DISCUSSION

This section specifies the results of the proposed model and a detailed discussion of different baseline studies on the suggested model's significance.

5.1. Proposed Study Results

The proposed model achieved an outstanding training accuracy of 98.62% and a testing accuracy of 97.05% on the SARC dataset. Figure 3 shows the accuracy of training and testing of the SACR dataset.

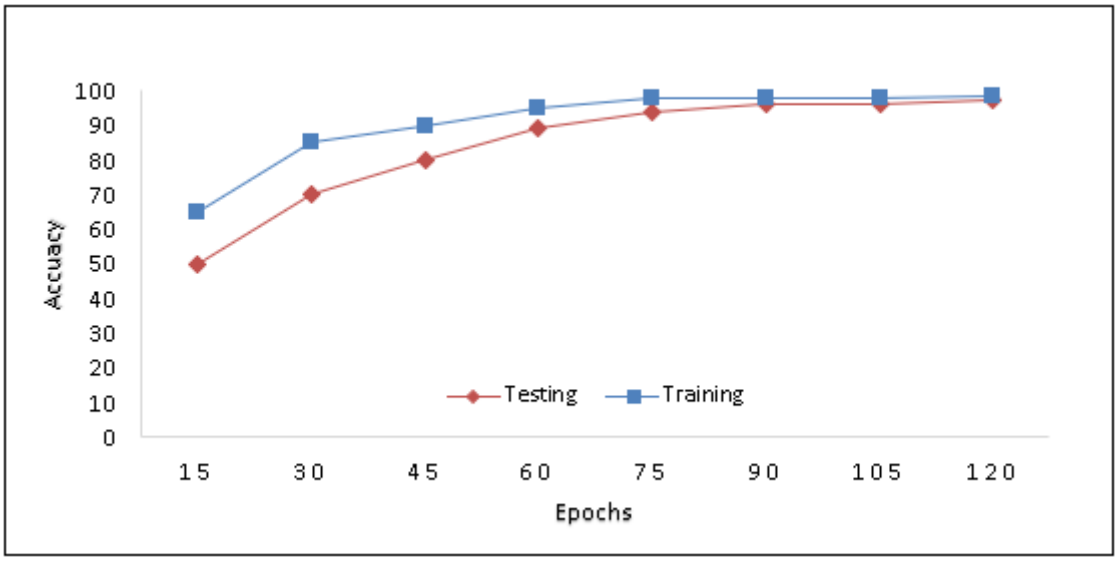

Figure 3. Training and testing the accuracy of the SACR dataset

Figure 4 indicates the training and testing loss of the model using the SARC dataset. The model is trained and tested for 120 epochs. Consequently, SARC data obtained the best result as compared to the HatEval dataset. The results of the self-attention-based transformer model are calculated using the SARC and HatEval datasets, representing its value in identifying sarcasm and offensive posts. The model accomplished a precision of 97.95%, a recall of 98.25%, and an F1-score of 98.10%, replicating its aptitude to correctly detect

sarcastic occurrences on the SARC dataset. Similarly, on the HatEval dataset, the model attained a precision of 96.86%, a recall of 97.83%, and an F1-score of 97.34%, signifying its strong performance in noticing offensive posts. These consequences highlight the model's reliable and consistent performance compared to HatEval datasets. Table 5 represents the results of the proposed model on the SARC and HatEval datasets in terms of precision, recall, and F1 score.

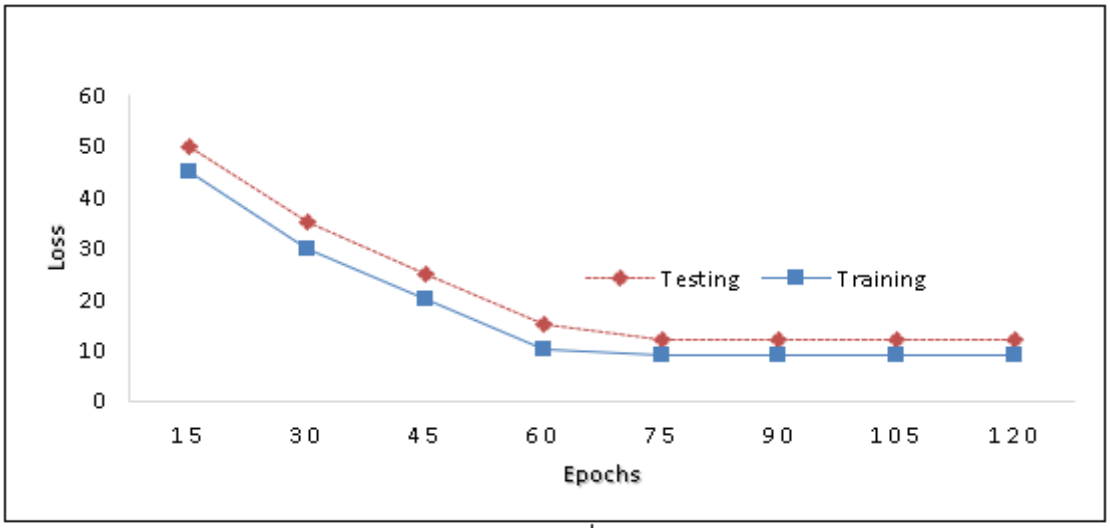

Figure 4. Training and testing loss of the model using the SARC dataset

5.2. Comparison with baseline approaches

Table 6 represents a comparative analysis of the results of the proposed model with baseline studies.

Table 5. Proposed models result

Models	Dataset	Precision (%)	Recall (%)	F1-Score (%)
Self-attention-based transformer model	SARC	97.95	98.25	98.10
	HatEval	96.86	97.83	97.34

Table 6. Comparison with baseline studies

Study	Method	Dataset	Precision (%)	Recall (%)	F1-Score (%)
[38]	MHA-BiLSTM	SARC	60.26	53.71	56.79
[39]	BERT	Hate speech	62.22	22.52	33.07
[40]	RSGNN	HatEval	74.29	74.14	74.04
[41]	BanglaBERT	Cyberbullying	85.80	90.0	87.85
[42]	LSTM-SSA model	HatEval	0.920	0.955	0.937
[43]	Hybrid Auto-Encoder-Based Model	SARC	0.83	0.85	0.84
[44]	IMLB-SDC technique	SARC	0.947	0.952	0.949
[45]	SD-GOADL	Reddit-2018 database	91.00	90.94	90.93
Proposed	Self-attention-based transformer Model	SARC	97.95	98.25	98.10
		HatEval	96.86	97.83	97.34

Figure 5 shows the graphical representation of the proposed model over different baseline tasks in immoral post-detection on social networks. The graph indicates that the suggested model achieved the highest results using the SARC dataset compared to state-of-the-art techniques. This reasonable evaluation highlights the performance of numerous approaches for sarcasm and offensive terms identification across different datasets. The study conducted by [38] employed a multi-head attention-based bidirectional LSTM technique that accomplished unassertive outcomes on the SARC dataset, with a precision of 60.26%, recall of 53.71%, and an F1-score of 56.79%. The research published in [39] applied the BERT model for hate speech exposure achieved less efficiently, with a precision of 62.22%, recall of 22.52%, and an F1-score of 33.07%. The study

accomplished in [40] RSGNN model indicated enriched presentation on the HatEval dataset with an F1-score of 74.04%. The work performed by [41] used the BanglaBERT model for cyberbullying identification and achieved greater metrics, with an F1-score of 87.85%.

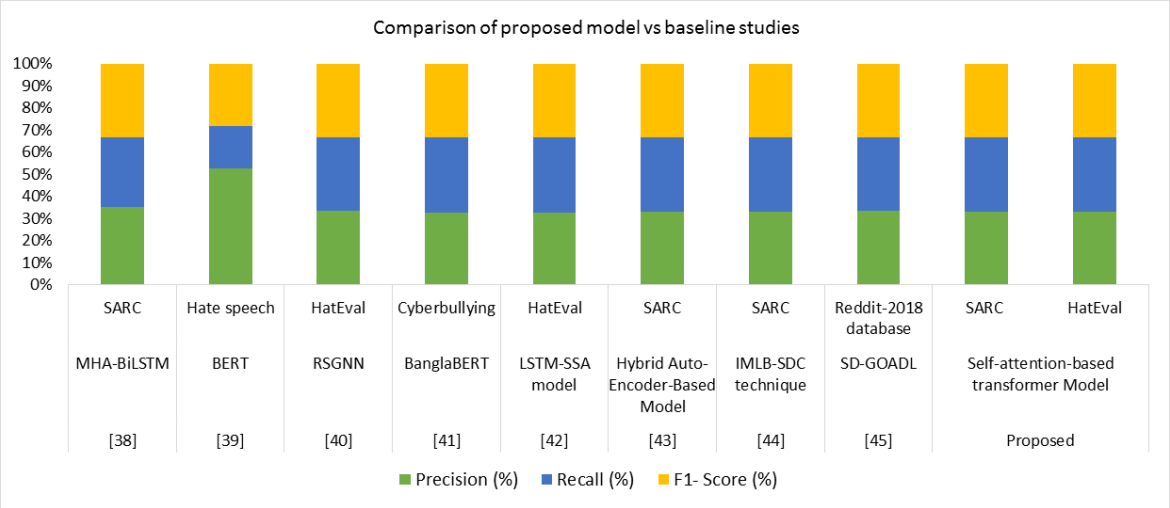

Figure 5. Proposed vs. Baseline results

Table 7. Complexity of the proposed model

Component	Operation	Complexity	Explanation
Self-Attention	Matrix multiplications and softmax	$O(n^2 \cdot d)$	The attention mechanism involves computing attention scores for all pairs of tokens, resulting in quadratic scaling with n.
Feed-Forward Network	Linear transformations	$O(n \cdot d^2)$	Each token passes through a dense layer, leading to quadratic complexity with respect to d.
Embedding Layer	Token embedding using contextual encodings	$O(n \cdot d)$	Maps tokens to vectors with a linear pass over the sequence using trainable embeddings.
Overall Model	Combination of the above components	$O(n^2 \cdot d + n \cdot d^2)$	Dominated by self-attention for long sequences (n^2) and feed-forward layers (d^2).

The work published in [42] applied the LSTM-SSA model to beat several other techniques using the HatEval dataset, with an F1-score of 93.7%. The authors of the studies[43] applied the hybrid auto-encoder-based model and attained an F1-score of 84% using the SARC dataset for irony identification. The study performed in [44] employed an intelligent ML-based sarcasm detection and classification IMLB-SDC method marginally upgraded in this case with an F1-score of 94.9%. The work conducted in [45] applied the grasshopper optimization algorithm SD-GOADL method, which exhibited robust consequences using the Reddit-2018 dataset, reaching an F1-score of 90.93%. The suggested model outperformed all the approaches in this study, reaching the uppermost precision, recall, and F1 scores on both the SARC and HatEval datasets. Specifically, it reached F1 scores of 98.10% and 97.34%, correspondingly representing its superior capability to control sarcasm and offensive language detection effectively.

The computational complexity of the proposed model can be examined based on its essential operations, containing the attention mechanism, feed-forward layers, and embedding processes. The complexity mainly depends on the input order length (n) and the embedding size (d), which find the computational cost for different operations. The self-attention mechanism, which calculates attention scores for all sets of tokens in an order, has a complexity of O (n2·d), where d is the embedding dimension and n signifies the sequence length. This quadratic scaling with n arises from the basic to capture contextual relations between entire tokens. After this, the feed-forward network improves the features by employing linear transformations to each token, leading to a complexity of O (n·d2), which depends on both the embedding dimension and the sequence length. Furthermore, token embeddings are prepared by a trainable embedding

layer, which maps input tokens to vector symbols with a complexity of O (n·d). Integrating these elements, the overall computational complexity of the model is O (n2·d+n·d2), with the self-attention mechanism usually controlling for longer orders due to its quadratic dependency on n. This complexity investigation notifies the balance between the model's capacities to handle nuanced contextual data and its computational requirements. Table 7 represents the complexity of the proposed model.

6. CONCLUSIONS, LIMITATIONS, & FUTURE AVENUE

In this work, we proposed a self-attention-based transformer model for the contextual examination of dissolute content on the web. We are addressing the rising want for automated tools to screen and control immoral content in digital spaces. The model influences advanced NLP methods, allowing it to understand not only the surface-level values or meaning of a text but also consider the context and determination behind user-generated content. The proposed model is commendably modified to detect unethical posts with great precision and consistency using the SARC and HatEval datasets. The self-attention mechanisms performed an essential role in its success, complying with the scheme with emphasis on the main parts of the input while seeing the broader context. This certified a nuanced understanding of intricate dialectal patterns, which is important for handling the vague and context-dependent nature of unwanted content. The capability to capture semantic dependencies and contextual relationships within the text provides an important benefit over traditional approaches. The experimental results show noteworthy enhancements in evaluation metrics, highlighting the model's strength and flexibility. The proposed model achieved outstanding results on both datasets. The model achieved the best results, particularly when using the SARC dataset, having a training accuracy of 98.62 and a testing accuracy of 97.05, respectively. This study contributes to the ongoing efforts to improve content moderation schemes, making the suggested method a favourable applicant for real-world uses.

The future work will concentrate on discovering different avenues for the improvement of the proposed technique. First, integrating more datasets and multilingual competencies can enlarge the model's strength across different web contexts. Second, assimilating explainable artificial intelligence methods could improve transparency by providing interpretable insights into the model's decision-making process. Third, deploying real-time recognition systems with optimized computational efficacy. Finally, addressing problems like bias mitigation and adversarial attacks will support the model's reliability. These directions have the goal of progressing ethical implications and the practical implementation of automated inappropriate content detection.

DATA AVAILABILITY STATEMENT

The original data presented in the study are openly available in Kaggle at https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset
https://www.kaggle.com/datasets/danofer/sarcasm

CONFLICTS OF INTEREST

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

REFERENCES

[1] S. Saumya, A. Kumar, and J. P. Singh, "Filtering offensive language from multilingual social media contents: A deep learning approach," *Eng. Appl. Artif. Intell.*, vol. 133, p. 108159, Jul. 2024, doi: 10.1016/j.engappai.2024.108159.

[2] A. U. Rahman, F. Al-Obeidat, A. Tubaishat, B. Shah, S. Anwar, and Z. Halim, "Discovering the Correlation Between Phishing Susceptibility Causing Data Biases and Big Five Personality Traits Using C-GAN," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 4, pp. 4800–4808, Aug. 2024, doi: 10.1109/TCSS.2022.3201153.

[3] P. Hajibabaee *et al.*, "Offensive Language Detection on Social Media Based on Text Classification," in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, IEEE, Jan. 2022, pp. 0092–0098. doi: 10.1109/CCWC54503.2022.9720804.

[4] B. Mathew, R. Dutt, P. Goyal, and A. Mukherjee, "Spread of Hate Speech in Online Social Media," in *Proceedings of the 10th ACM Conference on Web Science*, New York, NY, USA: ACM, Jun. 2019, pp. 173–182. doi: 10.1145/3292522.3326034.

[5] A. Arora *et al.*, "Detecting Harmful Content on Online Platforms: What Platforms Need vs. Where Research Efforts Go," *ACM Comput. Surv.*, vol. 56, no. 3, pp. 1–17, Mar. 2024, doi: 10.1145/3603399.

[6] R. Abdrakhmanov *et al.*, "Offensive Language Detection on Social Media using Machine Learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 5, 2024, doi: 10.14569/IJACSA.2024.0150557.

[7] K. Matsumoto, F. Ren, M. Matsuoka, M. Yoshida, and K. Kita, "Slang feature extraction by analyzing topic change on social media," *CAAI Trans. Intell. Technol.*, vol. 4, no. 1, pp. 64–71, Mar. 2019, doi: 10.1049/trit.2018.1060.

[8] R. Schifanella, P. de Juan, J. Tetreault, and L. Cao, "Detecting Sarcasm in Multimodal Social Platforms," in *Proceedings of the 24th ACM international conference on Multimedia*, New York, NY, USA: ACM, Oct. 2016, pp. 1136–1145. doi: 10.1145/2964284.2964321.

[9] H. Yenala, A. Jhanwar, M. K. Chinnakotla, and J. Goyal, "Deep learning for detecting inappropriate content in text," *Int. J. Data Sci. Anal.*, vol. 6, no. 4, pp. 273–286, Dec. 2018, doi: 10.1007/s41060-017-0088-4.

[10] M. Abulaish, A. Kamal, and M. J. Zaki, "A Survey of Figurative Language and Its Computational Detection in Online Social Networks," *ACM Trans. Web*, vol. 14, no. 1, pp. 1–52, Feb. 2020, doi: 10.1145/3375547.

[11] M. Anand, K. B. Sahay, M. A. Ahmed, D. Sultan, R. R. Chandan, and B. Singh, "Deep learning and natural language processing in computation for offensive language detection in online social networks by feature selection and ensemble classification techniques," *Theor. Comput. Sci.*, vol. 943, pp. 203–218, Jan. 2023, doi: 10.1016/j.tcs.2022.06.020.

[12] A. H. Razavi, D. Inkpen, S. Uritsky, and S. Matwin, "Offensive Language Detection Using Multi-level Classification," 2010, pp. 16–27. doi: 10.1007/978-3-642-13059-5_5.

[13] Bibi Saqia, Khairullah Khan, Atta Ur Rahman, and Wahab Khan, "Deep Learning-Based Identification of Immoral Posts on Social Media Using Fine-tuned Bert Model," *Int. J. Data Informatics Intell. Comput.*, vol. 3, no. 4, pp. 26–39, Nov. 2024, doi: 10.59461/ijdiic.v3i4.143.

[14] G. Xiang, B. Fan, L. Wang, J. Hong, and C. Rose, "Detecting offensive tweets via topical feature discovery over a large scale twitter corpus," in *Proceedings of the 21st ACM international conference on information and knowledge management*, New York, NY, USA: ACM, Oct. 2012, pp. 1980–1984. doi: 10.1145/2396761.2398556.

[15] I. Abdellaoui, A. Ibrahimi, M. A. El Bouni, A. Mourhir, S. Driouech, and M. Aghzal, "Investigating Offensive Language Detection in a Low-Resource Setting with a Robustness Perspective," *Big Data Cogn. Comput.*, vol. 8, no. 12, p. 170, Nov. 2024, doi: 10.3390/bdcc8120170.

[16] S. T. Aroyehun, and A. Gelbukh, "Aggression detection in social media: Using deep neural networks, data augmentation, and pseudo labeling," *Proc. first work. trolling, Aggress. cyberbullying*, pp. 90–97, 2018.

[17] N. S. Mullah and W. M. N. W. Zainon, "Advances in Machine Learning Algorithms for Hate Speech Detection in Social Media: A Review," *IEEE Access*, vol. 9, pp. 88364–88376, 2021, doi: 10.1109/ACCESS.2021.3089515.

[18] S. S. Ilhan, S. Sivakumar, N. J, S. Ramesh, N. Sreeram, and R. Rajalakshmi, "Hate Speech Detection and Classification Using NLP," in *2024 Second International Conference on Advances in Information Technology (ICAIT)*, IEEE, Jul. 2024, pp. 1–7. doi: 10.1109/ICAIT61638.2024.10690655.

[19] A. Toktarova *et al.*, "Hate Speech Detection in Social Networks using Machine Learning and Deep Learning Methods," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 5, 2023, doi: 10.14569/IJACSA.2023.0140542.

[20] Anjum and R. Katarya, "Hate speech, toxicity detection in online social media: a recent survey of state of the art and opportunities," *Int. J. Inf. Secur.*, vol. 23, no. 1, pp. 577–608, Feb. 2024, doi: 10.1007/s10207-023-00755-2.

[21] A. Maisto, S. Pelosi, S. Vietri, and P. Vitale, "Mining offensive language on social media," *Proc. Fourth Ital. Conf. Comput. Linguist. CLiC-it*, pp. 1–354, 2017.

[22] M. P. S. Bhatia and S. R. Sangwan, "Debunking Online Reputation Rumours Using Hybrid of Lexicon-Based and Machine Learning Techniques," 2020, pp. 317–327. doi: 10.1007/978-981-15-3369-3_25.

[23] S. Vashishtha and S. Susan, "Fuzzy rule based unsupervised sentiment analysis from social media posts," *Expert Syst. Appl.*, vol. 138, p. 112834, Dec. 2019, doi: 10.1016/j.eswa.2019.112834.

[24] K.A. Gemes, Á. Kovács, M. Reichel, and G. Recski, "Offensive text detection on English Twitter with deep learning models and rule-based systems," *CEUR Workshop Proc.*, pp. 1–14, 2021.

[25] M. Corazza, S. Menini, P. Arslan, R. Sprugnoli, E. Cabrio, S. Tonelli, and S. Villata, "Comparing different supervised approaches to hate speech detection," *EVALITA*, pp. 1–6, 2018.

[26] A. Muneer and S. M. Fati, "A Comparative Analysis of Machine Learning Techniques for Cyberbullying Detection on Twitter," *Futur. Internet*, vol. 12, no. 11, p. 187, Oct. 2020, doi: 10.3390/fi12110187.

[27] K. Sundararajan and A. Palanisamy, "Multi-Rule Based Ensemble Feature Selection Model for Sarcasm Type Detection in Twitter," *Comput. Intell. Neurosci.*, vol. 2020, pp. 1–17, Jan. 2020, doi: 10.1155/2020/2860479.

[28] K. Sentamilselvan, P. Suresh, G. K. Kamalam, S. Mahendran, and D. Aneri, "Detection on sarcasm using machine learning classifiers and rule based approach," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1055, no. 1, p. 012105, Feb. 2021, doi: 10.1088/1757-899X/1055/1/012105.

[29] C. Eke, A. A. Norman, L. Shuib, F. Faith B., and Z. A. Long, "RANDOM FOREST-BASED CLASSIFIER FOR AUTOMATIC SARCASM CLASSIFICATION ON TWITTER DATA USING MULTIPLE FEATURES," *J. Inf. Syst. Digit. Technol.*, vol. 4, no. 2, Dec. 2022, doi: 10.31436/jisdt.v4i2.345.

[30] C. I. Eke, A. A. Norman, and L. Shuib, "Multi-feature fusion framework for sarcasm identification on twitter data: A machine learning based approach," *PLoS One*, vol. 16, no. 6, p. e0252918, Jun. 2021, doi: 10.1371/journal.pone.0252918.

[31] D. Olaniyan, R. O. Ogundokun, O. P. Bernard, J. Olaniyan, R. Maskeliūnas, and H. B. Akande, "Utilizing an Attention-Based LSTM Model for Detecting Sarcasm and Irony in Social Media," *Computers*, vol. 12, no. 11, p. 231, Nov. 2023, doi: 10.3390/computers12110231.

[32] K. Gutiérrez-Batista, J. Gómez-Sánchez, and C. Fernandez-Basso, "Improving automatic cyberbullying detection in social network environments by fine-tuning a pre-trained sentence transformer language model," *Soc. Netw. Anal. Min.*, vol. 14, no. 1, p. 136, Jul. 2024, doi: 10.1007/s13278-024-01291-0.

[33] A. Kumar, "A Study: Hate Speech and Offensive Language Detection in Textual Data by Using RNN, CNN, LSTM and BERT Model," in *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, May 2022, pp. 1–6. doi: 10.1109/ICICCS53718.2022.9788347.

[34] A. Bisht, A. Singh, H. S. Bhadauria, J. Virmani, and Kriti, "Detection of Hate Speech and Offensive Language in Twitter Data Using LSTM Model," 2020, pp. 243–264. doi: 10.1007/978-981-15-2740-1_17.

[35] M. Khodak, N. Saunshi, and K. Vodrahalli, "A large self-annotated corpus for sarcasm," *Proc. Elev. Int. Conf. Lang. Resour. Eval. (LREC 2018)*, 2018.

[36] V. Basile *et al.*, "SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 54–63. doi: 10.18653/v1/S19-2007.

[37] A. U. Rahman, Y. Alsenani, A. Zafar, K. Ullah, K. Rabie, and T. Shongwe, "Enhancing heart disease prediction using a self-attention-based transformer model," *Sci. Rep.*, vol. 14, no. 1, p. 514, Jan. 2024, doi: 10.1038/s41598-024-51184-7.

[38] A. Kumar, V. T. Narapareddy, V. Aditya Srikanth, A. Malapati, and L. B. M. Neti, "Sarcasm Detection Using Multi-Head Attention Based Bidirectional LSTM," *IEEE Access*, vol. 8, pp. 6388–6397, 2020, doi: 10.1109/ACCESS.2019.2963630.

[39] V. Mercan, A. Jamil, A. A. Hameed, I. A. Magsi, S. Bazai, and S. A. Shah, "Hate Speech and Offensive Language Detection from Social Media," in *2021 International Conference on Computing, Electronic and Electrical Engineering (ICE Cube)*, IEEE, Oct. 2021, pp. 1–5. doi: 10.1109/ICECube53880.2021.9628255.

[40] R. Song, F. Giunchiglia, Q. Shen, N. Li, and H. Xu, "Improving Abusive Language Detection with online interaction network," *Inf. Process. Manag.*, vol. 59, no. 5, p. 103009, Sep. 2022, doi: 10.1016/j.ipm.2022.103009.

[41] K. Saifullah, M. I. Khan, S. Jamal, and I. H. Sarker, "Cyberbullying Text Identification based on Deep Learning and Transformer-based Language Models," *EAI Endorsed Trans. Ind. Networks Intell. Syst.*, vol. 11, no. 1, Feb. 2024, doi: 10.4108/eetinis.v11i1.4703.

[42] A. Kumar and S. Kumar, "Optimized Deep Neural Networks Using Sparrow Search Algorithms for Hate Speech Detection," *Int. J. Comput. Digit. Syst.*, vol. 15, no. 1, pp. 617–626, Feb. 2024, doi: 10.12785/ijcds/150145.

[43] D. K. Sharma, B. Singh, S. Agarwal, H. Kim, and R. Sharma, "Sarcasm Detection over Social Media Platforms Using Hybrid Auto-Encoder-Based Model," *Electronics*, vol. 11, no. 18, p. 2844, Sep. 2022, doi: 10.3390/electronics11182844.

[44] D. Vinoth and P. Prabhavathy, "An intelligent machine learning-based sarcasm detection and classification model on social networks," *J. Supercomput.*, vol. 78, no. 8, pp. 10575–10594, May 2022, doi: 10.1007/s11227-022-04312-x.

[45] N. Srinu, K. Sivaraman, and M. Sriram, "Enhancing sarcasm detection through grasshopper optimization with deep learning based sentiment analysis on social media," *Int. J. Inf. Technol.*, Aug. 2024, doi: 10.1007/s41870-024-02057-9.

BIOGRAPHIES OF AUTHORS

Bibi Saqia received an MS degree in Computer Science from the University of Science and Technology Bannu in 2018. Currently, her PhD is in progress at the University of Science and Technology Bannu. She has more than 10 publications in various reputed journals and conferences, including IEEE Transactions. Her research interests include data mining, machine learning, and artificial intelligence. She can be contacted at email: saqiaktk@ustb.edu.pk

Khairullah Khan received a PhD degree in information technology from Universiti Teknologi PETRONAS, Malaysia, in 2012, where he worked on machine learning for the automatic detection of opinion targets from the text. He is currently a Professor at the Department of Computer Science, University of Science and Technology, Bannu, Pakistan. He has more than 50 publications in various reputed journals and conferences, including IEEE Transactions. His research interests include Data Mining, Web Mining, Opinion Mining, Machine Learning and Artificial Intelligence. He can be contacted at email: khairullah@ustb.edu.pk

Atta Ur Rahman received the BS (Hons) degree in Telecommunication from USTB, in 2014 and the MS degree in Computer Science from the same university. He obtained his PhD degree in Computer Science in 2022 from Ghulam Ishaq Khan (GIK) Institute of Engineering Sciences and Technology, Pakistan. Currently, he is working as a postdoctoral Researcher at King Fahd University of Petroleum and Minerals. He has more than 25 publications in various reputed journals and conferences, including IEEE Transactions. His research interests include human-computer interaction, artificial intelligence in healthcare, and Federated learning for privacy preservation. He can be contacted at email: attaur.rahman@kfupm.edu.sa

Wahab Khan received an MS degree in computer science from the University of Science and Technology, Bannu, Pakistan, in 2009. He is currently pursuing a PhD degree in computer science at the International Islamic University Islamabad, Pakistan. He has more than 25 publications in various reputed journals and conferences, including IEEE Transactions. His research interests include natural language processing, machine learning, and deep learning and data mining. He can be contacted at email: wahabshri@gmail.com