International Journal of Electrical and Computer Engineering (IJECE)
Vol. 15, No. 3, June 2025, pp. 3139~3148
ISSN: 2088-8708, DOI: 10.11591/ijece.v15i3.pp3139-3148

Enhancing cyberbullying detection with advanced text preprocessing and machine learning

Rakesh Bapu Dhumale[1], Ajay Kumar Dass[2], Amit Umbrajkaar[3], Pradeep Mane[1]
[1]Department of Electronics and Telecommunication Engineering, AISSMS Institute of Information Technology, Pune, India
[2]Department of Electronics and Telecommunication Engineering, Sinhgad College of Engineering, Pune, India
[3]School of Mechanical Engineering, D. Y. Patil International University, Pune, India

Article Info

Article history:

Received Aug 24, 2024
Revised Feb 7, 2025
Accepted Mar 4, 2025

Keywords:

Cyberbullying
Detection
Online threats
Social media
Social media misuse
Support vector machines
Text classification

ABSTRACT

The use of social media and the internet has been increasing dramatically in recent years. Cyber-bullying is the term used to describe the misuse of social media by some people who make threatening comments. This has a devastating influence on people's lives, especially those of children and teenagers, and can lead to feelings of depression and suicidal thoughts. The methodology proposed in this paper includes four steps for identifying cyberbullying: preprocessing, feature extraction, classification, and evaluation. The first step is to create a labeled, varied dataset. Word2Vec and term frequency-inverse document frequency are used in feature extraction to transform text into high-dimensional vectors. Word2Vec creates word embeddings using the skip-gram and continuous bag-of-words models, while term frequency-inverse document frequency assesses the text's term relevancy. Support vector machine classifiers are used in the model, and their effectiveness is compared to that of other techniques like logistic regression and naïve Bayes. The classifiers support vector machine, naïve Bayes, and logistic regression were assessed. The maximum accuracy was 95% for the support vector classifier with skip-gram and 93% for continuous bag-of-words. For sentiment categories, F1-scores, recall, and precision were computed. The average precision and recall were 0.77 and 0.79, respectively.

Corresponding Author:

Rakesh Bapu Dhumale
Department of Electronics and Telecommunication Engineering, AISSMS Institute of Information Technology
Pune, India
Email: rbd.scoe@gmail.com

1. INTRODUCTION

Social media addiction has increased globally in direct proportion to the phenomenal growth in data service availability. Similar to other nations, cyberbullying has sharply increased [1]. With everyone living on digital and online platforms in this Web 4.0 era, it is extremely challenging to safeguard society against the startling increase in cybercrime [2]. It has been discovered that teenagers are the main victims of cyberbullying [3]. Cyberbullying is a dangerous and damaging behavior that can make victims try suicide and suffer lasting consequences [4]. Cyberbullying detection can be seen as a classification problem in which posts made online are classified as either bullying or non-bullying ones [5]. A system that enhances the performance of detecting cyberbullying will be created via a variety of computer vision (CV), machine learning (ML) and natural language processing (NLP) techniques [6].

Social media's rise has led to an increase in cyberbullying, which calls for effective detection techniques. In [7], a hybrid random forest-based CNN model for text classification is presented. It achieves good accuracy and faster results for timely intervention than previous ML and deep learning (DL) methods. The work [8], improved a system that uses term frequency-inverse document frequency (TF-IDF) (bigram) feature extraction and multi-classification algorithms to detect six different types of cyberbullying tweets with high accuracy. Traditional classifiers were outperformed by ensemble approaches integrating decision trees, random forest, and XGBoost; the stacking classifier achieved an accuracy of 90.71%.

Effective detection techniques are required because of the surge in cyberbullying that has resulted from the increased usage of social media platforms. In order to categorize aggressive tweets, Muneer *et al.* [9] proposes an automated method that uses a stacking ensemble of five ML algorithms. It outperforms conventional models and prior studies, with an accuracy rate of 94.00%. The study [10] addresses the challenge of identifying cyberbullying comments with subtle expressions by proposing a novel approach using the bidirectional encoder representations from transformers (BERT) pre-training model and the BiSRU++ model with attention mechanisms, improving classification accuracy and generalization through a multi-task learning framework.

In order to address the problem of cyberbullying on social media, Gs *et al.* [11] suggests a ML-based detection method that makes use of a variety of datasets and sophisticated NLP techniques. It achieves high accuracy by using support vector machine (SVM) as the most efficient algorithm and emphasizes the significance of ethical considerations for improving online safety. The study improves the identification of cyberbullying in China by putting forth a hybrid model that combines deep bidirectional long short-term memory (Bi-LSTM) with XLNet [12]. This model beats both standard and deep learning baselines, improving accuracy by 1.13% over BERT, 4.29% over SVM, and 1.49% over gated recurrent unit.

The study repurposes the Sentiment140 dataset to identify cyberbullying on Twitter using artificial intelligence. High precision is demonstrated by naive Bayes (NB) and Bi-LSTM models; yet, the study emphasizes the need for improved models and a wider range of data [13]. The problem of cyberbullying is getting worse as social media changes. The work presents combinational network for bullying detection (CNBD), a deep learning model that improves accuracy, precision, and recall in identifying cyberbullying in photographs. It combines binary encoder image transformer (BEiT) and multi-layer perceptron (MLP) and is augmented by OCR and image captioning [14].

The study creates BullyFilterNeT, a cyberbullying detection system for Bengali social media writings using transformer-based and non-contextual embedding. The BanglaBERT-based BullyFilterNeT successfully handles out-of-vocabulary issues with an accuracy of 88.04% [15]. Tapsoba *et al.* [16] suggest a ML strategy that combines long short term memory (LSTM), random forest (RF) and SVM to improve cyber threat identification in Burkina Faso. With RF, the approach achieves better accuracy in cyberbullying, phishing and online frauds. The paper [17] uses latent semantic analysis for sentiment analysis to classify cyberbullying tweets, achieving higher accuracy than existing methods, thus aiding in reducing online abuse and supporting safe social media usage. The study proposes a ML approach using a multinomial NB classifier to detect cyberbullying on social networks, achieving 88.76% accuracy in classifying shaming, sexual harassment, and racism, with fuzzy rule sets for bullying strength [18].

In order to detect cyber bullying in Arabic, the study investigates ML methods. Because of their high accuracy, Al Harbi *et al.* [19] suggest ridge and logistic regression (LR). The goal is to use these methods to identify cyber bullying in Arabic. With ML, cyberbullying can be automatically detected by analyzing linguistic patterns. Hani *et al.* [20] suggests a classifier-based supervised strategy. Neural network outperforms other models with an accuracy of 92.8%, according to evaluation. Cyberbullying via Wikipedia and Twitter is detected by NLP and ML with over 90% and 80% accuracy, respectively [21]. The goal of [22] is to combine ML and NLP to provide an efficient method for identifying abusive and harassing messages sent online. The study uses TF-IDF and bag-of-words (BoW) variables to examine the accuracy of four ML algorithms. The goal of [23] is to combine ML and NLP to provide an efficient method for identifying abusive and harassing messages sent online. The study uses TF-IDF and BoW variables to examine the accuracy of four ML algorithms. Developing reliable numerical representations of text messages is a major concern in this study. In order to improve representation learning, the work presents the semantic-enhanced marginalized denoising auto-encoder (SMSDA), an enhanced deep learning model that augments stacked denoising auto-encoders with semantic dropout noise and sparsity constraints [24].

This research provides a novel method that achieves 95% accuracy in cyber threat detection by integrating Word2Vec with SVM. Preprocessing, feature extraction, and classification using Word2Vec and TF-IDF are all part of the methodology; SVM performs better than other methods like logistic regression and naïve Bayes. The employment of both skip-gram and continuous bag-of-words models, which improve the feature extraction process, is a significant advance in Word2Vec. The suggested approach performs better than previous studies and has an average precision and recall of 0.77 and 0.79, respectively. The study

emphasizes the significance of tackling the rapidly expanding issue of cyberbullying, particularly its detrimental effects on teenagers, and the usefulness of this research for proactive measures like real-time surveillance. This work is not only effective but also scalable, paving the way for future research to improve search accuracy and compatibility across different platforms and languages.

2. MATERIALS AND METHODS

There are four stages in the suggested methodology for detecting cyberbullying, preprocessing, feature extraction, classification, and evaluation. This section goes over each step in depth. A labeled dataset is compiled with cases of both non-cyberbullying and cyberbullying. The target population should be well-represented in this dataset, which should be diverse. One of the most important steps in the detection of cyberbullying is preprocessing. It entails removing spam content and cleaning texts such as removing punctuation and stop words. Preprocessing has been utilized in the proposed model to remove and sort-out unwanted noise in text detection like repetitive words, special characters and stop words are removed. After that, stemming was used to the remaining words to restore them to their original forms. A clean twitter dataset is produced by this preprocessing, which can be used to run the recommended model and produce predictions.

A crucial stage in text classification for cyberbullying detection is feature extraction. Word2Vec and TF-IDF approaches have been employed for feature extraction in the proposed model which is based on word statistics. This model solely takes into account word expressions that are consistent across all texts. As a result, one of the most often utilized feature extraction methods for text detection is TF-IDF. Word2Vec is a neural network with two layers that processes text by "vectorizing" words. Corpuses of text serves as its input and a set of vectors attribute vectors that reflect the words in that structure serve as its output. The Word2Vec technique creates a high-dimensional vector for every word by utilizing two hidden layers of shallow neural networks: the skip-gram model (SGM) and the continuous bag-of-words (CBOW).

The continuous bag-of-words model (BWM) uses the surrounding context words to forecast the middle word. The words that come before and after the current (middle) word make up the context. Because it doesn't matter what words are used in what sequence in the context, this architecture is known as a bag-of-words model. The continuous SKM forecasts words that fall within a specific range both before and after the current word in a phrase. The example is given in Figure 1. A SGM predicts the context (or neighbors) of a word given the word itself, whereas a BWM predicts a word based on the surrounding context. SKM, or n-SKM that permit tokens to be skipped as given in Figure 1, are used to train the model. A set of skip-gram pairings of (*target_word*, *context_word*) where *context_word* appears in the nearby context of *target_word* can be used to indicate the context of a word. A window size defines the context words for each of the words in this phrase. The span of words that can be regarded as context words on each side of a target word depends on the window size. Think of a group of words Ш. If the sliding window size ($\acute{C}$) is 2, and $ω_i$ is the input (center word), then the context words are $ω_{i-2}$, $ω_{i-1}$, $ω_{i+1}$, and $ω_{i+2}$. The SGM using the neural network's backpropagation equations, which is implemented in this work is shown in Figure 1.

Optimizing the likelihood of predicting context words based on the target word is the SGM's training goal. The aim for a word sequence consisting of $ω_{i-2}$, $ω_{i-1}$, $ω_{i+1}$, and $ω_{i+2}$ can be expressed as (1) for the average log probability.

$$\log(p(W)) = \frac{1}{N}\sum_{t=1}^{N}\sum_{-\acute{C}\le k\le \acute{C}}\log\left(p(ω_{t+k}|ω_t)\right) \quad (1)$$

The basic SKM defines this probability using the soft-max function as given in (2).

$$P(\text{context\_word}/\text{target\_word}) = \frac{e^{\left(V'_{\text{context\_word}}V_{\text{target\_word}}\right)}}{\sum_{i=1}^{V} e^{\left(V'_i V_{\text{target\_word}}\right)}} \quad (2)$$

Where, the vocabulary size is V. $V_{\text{target\_word}}$ is the target word's vector representation. $V'_{\text{context\_word}}$ is the context word's vector representation. The exponentiated dot product between the target word vector and the context word vector is represented by the numerator. The total of these exponentiated dot products for each word in the lexicon is the denominator.

In order to calculate the denominator of this formulation, a thorough softmax over the entire vocabulary must be performed, which frequently consists of a significant number of phrases (10^5 - 10^7). An effective approximation for a complete softmax is the noise contrastive estimation (NCE) loss function. It is possible to simplify the NCE loss by using negative sampling with the goal of learning word embeddings rather than modeling the word distribution. The goal of simplified negative sampling for a target word is to isolate the context word from negative samples that are selected from a word noise distribution ($P_n(V)$).

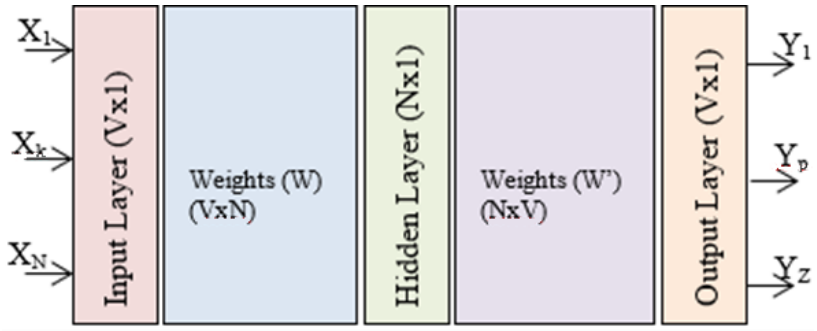

Figure 1. Skim gram architecture using back propagation

More exactly, considering the loss for a *target_word* as a classification problem between the *context_word* and negative samples is an efficient approximation of full softmax over the vocabulary for a skip-gram pair. A *target_word* and *context_word* pair is considered negative if and only if the *context_word* does not occur in the *window_size* neighborhood of the *target_word*. (better, back) is pair of negative samples of example "You better watch your back." SKG using back propagation neural network (BPNN) is shown in Figure 1. The variables are defined as below.
V: Count of distinct terms in our text corpus (vocabulary)
X: Input layer (single hot encoding of our input word)
N: The quantity of neurons within the neural network's hidden layer
W: Weights separating the input layer from the hidden layer
W': Weights separating the output layer from the hidden layer
Y: A soft-max output layer with probabilities for each word in our lexicon

TF-IDF is the procedure for figuring out how relevant a word is to a text in a corpus or series. A word's meaning increases in direct proportion to how often it appears in the text; but, the corpus's (dataset's) word frequency balances this out. The TF-IDF weight of a term in a document is represented mathematically as (3).

$$W_{(d,t)} = TF(d,t) * \log\left(\frac{N}{d\ f(t)}\right) \quad (3)$$

The number N in this instance indicates the total number of documents, and the number of documents in the corpus that include the term T is denoted by D f(T). The first component in (3) improves memory, and the second term improves word embedding accuracy.

In this work, SVM classifiers have been used to determine if a tweet qualifies as cyberbullying or not. SVM classifier performance is contrasted with that of LightGBM, SGD, RF, AdaBoost, and NB. SVM optimizes the distance (margin) between two categories by transforming the original feature space into a user-defined higher-dimensional space based on a kernel. SVM first makes an approximation of the hyperplane dividing the two categories.

3. IMPLEMENTATION

The following sentence is an example taken into consideration. "You better watch your back." Tokenizing the sentence, or dividing it into discrete words or tokens, is the first step in the process. Tokenization facilitates the conversion of the sentence into a format that is easily processed for word embeddings and other NLP tasks. The detail steps are below:

a. First step is to create a vocabulary to save mappings from tokens to integer indices.

$$\{0 :< you >, 1 :< better >, 2 :< watch >, 3 :< your >, 4 :< back >\}$$

b. In order to store mappings between integer indices and tokens, create an inverted vocabulary.

$$\{< you >: 0, < better >: 1, < watch >: 2, < your >: 3, < back >: 4\}$$

c. After that, the sentence must be vectorized. [0, 1, 2, 3, 4].

d. Generate skip-gram pairs from the sentence. Few examples of positive (*target_word*, *context_word*) pair are shown in Figure 2.
e. Negative sampling for one skip-gram: The skip-grams function slides across a specified window span to return all affirmative skip-gram pairs. Words from the vocabulary must be randomly sampled in order to generate more skip-gram pairings that will be used as training negative examples. The process involves applying the approach to the target word of a single skip-gram and passing the context word as the true class to prevent it from being sampled.
f. Construct one training example: Negative sampled context words for a particular positive skip-gram that do not occur in the window size neighborhood of the *target_word*. Combine the one context word that is positive and the one that is negative. For every target word, this yields a set of positive skip-grams with label as 1 and negative samples with label as 0. The training samples are shown in Table 1.
g. The process for figuring out a word's relevance to a text in a corpus or series is called TF-IDF. Mathematically, a term's TF-IDF weight in a document is represented by (2). Let us consider following sentences.
   - Sentence 1: You better watch your back
   - Sentence 2: Watch your back
   - Sentence 3: Be careful with your back.

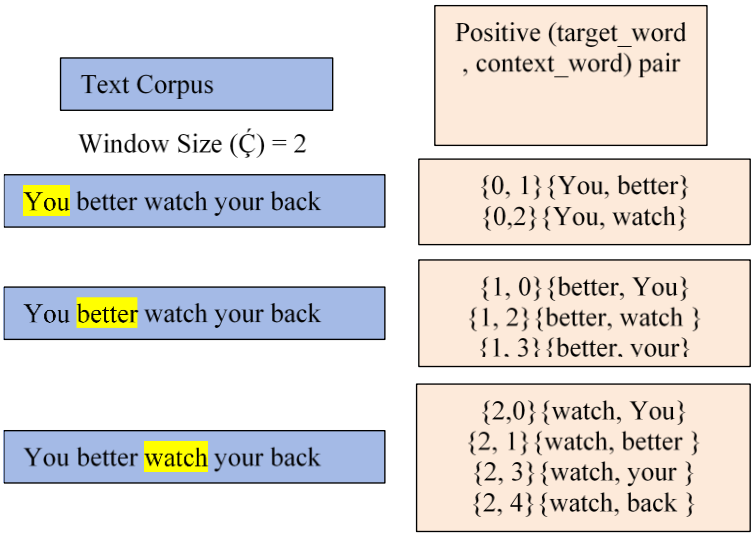

Figure 2. Skip-gram model

Table 1. The training samples

Word	Context Word	Labels
0	{1,2,3,4}	{1,1,0,0}
1	{0,2,3,4}	{1,1,1,0}
2	{0,1,3,4}	{1,1,1,1}
3	{0,1,2,4}	{0,1,1,1}
4	{0,1,2,3}	{0,0,1,1}
↓	↓	↓
A few words are absent from the sentence provided. Just like several other indices that were utilized in the training samples, they are part of the vocabulary.		
V	{34, 5, 2, 91,7}	{0,1,1,1}

The word's relevance to a text in corpus is calculated using (5) and (6). The calculations are given in Tables 2, 3 and 4.

$$TF = \frac{Number\ of\ repetation\ of\ words\ in\ a\ sentence}{Number\ of\ words\ in\ a\ sentence} \quad (5)$$

$$DF = \log\left(\frac{No\ of\ sentences}{No\ of\ sentences\ containing\ word}\right) \quad (6)$$

Table 2. Term frequency

Words	Sent 1	Sent 2	Sent 3
you	1/5	0/3	0/5
better	1/5	0/3	0/5
watch	1/5	1/3	0/5
your	1/5	1/3	1/5
back	1/5	1/3	1/5
be	0/5	0/3	1/5
careful	0/5	0/3	1/5
with	0/5	0/3	1/5

Table 3. Inverse document frequency

Words	IDF
you	log(3/1)=0.47
better	log(3/1)=0.47
watch	log(3/2)=0.17
your	log(3/3)=0.00
back	log(3/3)=0.00
be	log(3/1)= 0.47
careful	log(3/1)=0.47
with	log(3/1)=0.47

Table 4. Term frequency inverse document frequency (TF X IDF)

Words	Sent 1	Sent 2	Sent 3
you	0.094	0.000	0.000
better	0.094	0.000	0.000
watch	0.034	0.566	0.000
your	0.000	0.000	0.000
back	0.000	0.000	0.000
be	0.000	0.000	0.094
careful	0.000	0.000	0.094
with	0.000	0.000	0.094

Finding the optimal hyperplane to divide the negative and positive classes is the idea behind SVM classification. By applying the kernel method, SVM may also operate on high-dimensional datasets. The SVM kernel performs a variety of tasks, including additive multi-quadratic inverse, sigmoid, linear, polynomial and Gaussian RBF functions. SVM Polynomial is the kernel function used in this study. When a hyperplane can separate the data, a linear SVM is used; when only curved lines can do so, a non-linear SVM is utilized.

An SVM kernel that uses a polynomial function to translate data into a higher-dimensional space is called a polynomial kernel. To do this, it takes the dot product of the polynomial function in the new space and the data points in the original space. When the data cannot be separated linearly, this kernel is frequently employed in SVM classification tasks. Occasionally, a hyperplane that divides the classes can be found via the polynomial kernel by transferring the data into a higher-dimensional space. A number of parameters, such as the polynomial's degree and coefficient, can be adjusted to enhance the performance of the polynomial kernel. The polynomial kernel for polynomials of degree d is defined as (7).

$$K\left(\overrightarrow{X_i}, \overrightarrow{X_j}\right) = \left(X_i^T X_j + 1\right)^d \quad (7)$$

where $K\left(\overrightarrow{X_i}, \overrightarrow{X_j}\right)$ is kernel function, $\overrightarrow{X_i}, \overrightarrow{X_j}$ are inputs and *d* is an order. By expressing it as a QP problem and utilizing a commonly accessible numerical analysis library to solve it, the hyperplane in the ideal SVM is achieved. The sequential technique is an additional, very straightforward option.

4. RESULTS AND DISCUSSION

The experiment's settings are shown in this section along with an explanation of their significance. Python 3.8 and Google Colab GPUs are used. The suggested cyberbullying model and other baseline models were implemented using the TensorFlow library, which is used for applications like CV and NLP. We used a publicly available dataset, as cited in reference [25] of our manuscript. To help the cyberbullying model grasp the context and analyze the important information in the text, an input matrix of 35,873 tweets was

created and tokenized using the TensorFlow library's word sequence analysis. Prior to tokenization, a preprocessing phase was used to eliminate missing and duplicate documents, erroneous text formats, and text content loss. In order to reduce noise and the size of the feature set, words that provided no significance to the phrase were eliminated from the text. They also wouldn't affect text processing for the intended purpose. SVM classifiers were built and evaluated with NB and LR for the purpose of classifying cyberbullying texts. Each classifier was initialized and then trained using the scaled oversampled training data using the fit method. After the test dataset was categorized, the accuracy score technique was used to evaluate the test's effectiveness. Figure 3 demonstrates the best performing classifiers, which were based on SVM, NB and LR. The accuracy percentages of three distinct classifiers, namely NB, LR, and support vector classifier, are given in the Figure 3 after they have been trained with two distinct training models, CBOW and skip-gram. The highest accuracy in this category was 93% for the support vector classifier with CBOW, followed by 89% for NB and 91% for LR. NB increased to 92%, LR to 93%, and support vector classifier once more had the highest accuracy at 95% for the skip-gram training model. For both training models, the support vector classifier showed the best overall accuracy of all the classifiers, scoring 93% for CBOW and 95% for skip-gram. Despite being less accurate than the other two classifiers, NB managed to achieve impressive results, scoring 89% for CBOW and 92% for skip-gram. In comparison to the CBOW model, the skip-gram training approach typically yielded greater accuracy percentages across all classifiers. According to this analysis, the combination that works best among those examined is the support vector classifier combined with skip-gram, which yields the maximum accuracy.

The precision, recall, and F1-score for the three sentiment categories like positive, neutral, and negative are given in the Table 5. The classifier produced an F1-score of 0.86, recall of 0.81, and precision of 0.92 for negative sentiment. The levels for neutral sentiment were 0.57, 0.69, and 0.63, in that order. An F1-score of 0.69, a recall of 0.75, and a precision of 0.64 were obtained from positive sentiment categorization. The overall average precision, recall, and F1-score were 0.79, 0.77, and 0.78, respectively. TensorFlow baseline models and a cyberbullying model were implemented in the experiment using Google Colab GPUs and Python 3.8. To aid the model in comprehending context and analyzing important data, an input matrix consisting of 35,873 tweets was generated and tokenized. Preprocessing methods reduced noise and feature set size by removing duplicate and missing documents, incorrect text formats, and unnecessary words. To categorize texts including cyberbullying, NB and LR were evaluated with SVM classifiers. Accuracy scores were used to evaluate each classifier's performance after it had been trained using scaled oversampled training data.

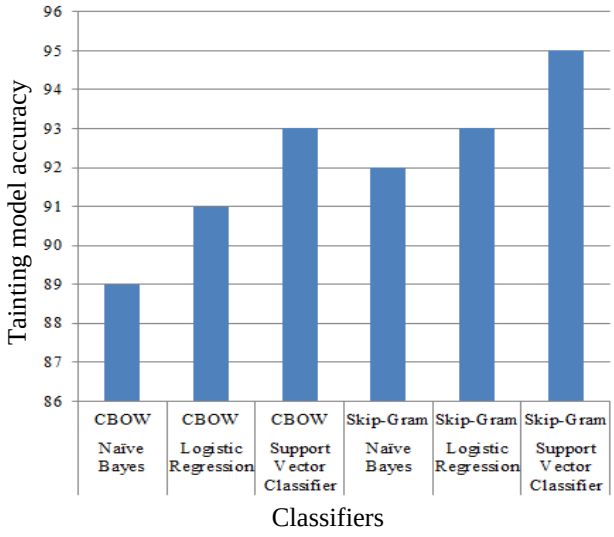

Figure 3. Accuracy percentage by classifier and training model accuracy (%)

Table 5. Classification report of support vector classifier

	Precision	Recall	F1-score
Negative	0.92	0.81	0.86
Neutral	0.57	0.69	0.63
Positive	0.64	0.75	0.69
Average/Total	0.79	0.77	0.78

5. CONCLUSION

Cyberbullying, in which people use social media to post threatening remarks, is a major issue. This can have a serious negative effect on people's lives, especially that of kids and teenagers, resulting in suicidal thoughts and depression. The study suggests a four-step procedure that involves preprocessing, feature extraction, classification, and evaluation to identify cyberbullying. Making a varied labeled dataset is the first step. Word2Vec and TF-IDF are used in feature extraction to transform text into vectors. SVM classifiers are used in the model, and their efficacy is compared to that of logistic regression and NB. The highest accuracy was 93% with CBOW and 95% with the support vector classifier employing skip-gram. For sentiment categories, the average precision and recall were 0.77 and 0.79, respectively.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Rakesh Bapu Dhumale	✓	✓	✓	✓		✓		✓	✓	✓			✓	
Ajay Kumar Dass		✓				✓			✓		✓	✓		
Amit Umbrajkaar	✓		✓			✓				✓			✓	
Pradeep Mane					✓		✓	✓		✓			✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author, RBD.

REFERENCES

[1] C. Cheng, Y. ching Lau, L. Chan, and J. W. Luk, "Prevalence of social media addiction across 32 nations: Meta-analysis with subgroup analysis of classification schemes and cultural values," *Addictive Behaviors*, vol. 117, p. 106845, 2021, doi: 10.1016/j.addbeh.2021.106845.

[2] R. S. Deora and D. M. Chudasama, "Brief study of cybercrime on an internet," *Journal of Communication Engineering & Systems*, vol. 11, no. 1, pp. 1–6, 2021, doi: 10.37591/JoCES.

[3] S. Alim, "Cyberbullying in the world of teenagers and social media: A literature review," *International Journal of Cyber Behavior, Psychology and Learning*, vol. 6, no. 2, pp. 68–95, 2016, doi: 10.4018/IJCBPL.2016040105.

[4] R. Dennehy, S. Meaney, M. Cronin, and E. Arensman, "The psychosocial impacts of cybervictimisation and barriers to seeking social support: Young people's perspectives," *Children and Youth Services Review*, vol. 111, no. February 2020, p. 104872, 2020, doi: 10.1016/j.childyouth.2020.104872.

[5] V. Sahana, K. M. A. Kumar, and A. A. Darem, "A comparative analysis of machine learning techniques for cyberbullying detection on FormSpring in textual modality," *International Journal of Computer Network and Information Security*, vol. 15, no. 4, pp. 36–47, 2023, doi: 10.5815/ijcnis.2023.04.04.

[6] C. Raj, A. Agarwal, G. Bharathy, B. Narayan, and M. Prasad, "Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques," *Electronics (Switzerland)*, vol. 10, no. 22, pp. 1–20, 2021, doi: 10.3390/electronics10222810.

[7] A. F. Alqahtani and M. Ilyas, "An ensemble-based multi-classification machine learning classifiers approach to detect multiple classes of cyberbullying," *Machine Learning and Knowledge Extraction*, vol. 6, no. 1, pp. 156–170, 2024, doi: 10.3390/make6010009.

[8] T. N. Harshitha *et al.*, "ProTect: a hybrid deep learning model for proactive detection of cyberbullying on social media," *Frontiers in Artificial Intelligence*, vol. 7, 2024, doi: 10.3389/frai.2024.1269366.

[9] A. Muneer, A. Alwadain, M. G. Ragab, and A. Alqushaibi, "Cyberbullying detection on social media using stacking ensemble learning and enhanced BERT," *Information (Switzerland)*, vol. 14, no. 8, 2023, doi: 10.3390/info14080467.
[10] G. Xingyi and H. M. Adnan, "Potential cyberbullying detection in social media platforms based on a multi-task learning framework," *International Journal of Data and Network Science*, vol. 8, no. 1, pp. 25–34, 2024, doi: 10.5267/j.ijdns.2023.10.021.
[11] N. Gs, A. Shenoyy, K. Chaturya, L. Jc, and J. Shree, "Detection of cyberbullying using NLP and machine learning in social networks for bi-language," *International Journal of Scientific Research & Engineering Trends*, vol. 10, no. 1, pp. 2395–2566, 2024.
[12] M. S. Islam, A. N. Orno, and M. Arifuzzaman, "Approach to social media cyberbullying and harassment detection using advanced machine learning," *Available at SSRN 4705261*. Mar. 12, 2024, doi: 10.21203/rs.3.rs-4031554/v1.
[13] A. Orelaja *et al.*, "Attribute-specific cyberbullying detection using artificial intelligence," *Journal of Electronic & Information Systems*, vol. 6, no. 1, pp. 10–21, 2024, doi: 10.30564/jeis.v6i1.6206.
[14] S. Pericherla and E. Ilavarasan, "Overcoming the challenge of cyberbullying detection in images: A deep learning approach with image captioning and OCR integration," *International Journal of Computing and Digital Systems*, vol. 15, no. 1, pp. 393–401, 2024, doi: 10.12785/ijcds/150130.
[15] K. Saifullah, M. I. Khan, S. Jamal, and I. H. Sarker, "Cyberbullying text identification: A deep learning and transformer-based language modeling approach," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 11, no. 1, pp. 1–12, 2024, doi: 10.4108/EETINIS.V11I1.4703.
[16] W. C. Tapsoba *et al.*, "Cyber threat's detection using machine learning algorithms," pp. 0–9, 2024.
[17] D. C. Joy Winnie Wise, S. Ambareesh, B. P. Ramesh, D. Sugumar, J. P. Bhimavarapu, and A. S. Kumar, "Latent semantic analysis based sentimental analysis of tweets in social media for the classification of cyberbullying text," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 7s, pp. 26–35, 2024.
[18] A. Akhter, K. A. Uzzal, and M. M. A. Polash, "Cyber bullying detection and classification using multinomial naïve Bayes and fuzzy logic," *International Journal of Mathematical Sciences and Computing*, vol. 5, no. 4, pp. 1–12, 2019, doi: 10.5815/ijmsc.2019.04.01.
[19] B. Y. AlHarbi, M. S. AlHarbi, N. J. AlZahrani, M. M. Alsheail, and D. M. Ibrahim, "Using machine learning algorithms for automatic cyber bullying detection in Arabic social media," *Journal of Information Technology Management*, vol. 12, no. 2, pp. 123–130, 2020, doi: 10.22059/JITM.2020.75796.
[20] J. Hani, M. Nashaat, M. Ahmed, Z. Emad, E. Amer, and A. Mohammed, "Social media cyberbullying detection using machine learning," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, 2019, doi: 10.14569/IJACSA.2019.0100587.
[21] V. Jain, V. Kumar, V. Pal, and D. K. Vishwakarma, "Detection of cyberbullying on social media using machine learning," in *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*, 2021, pp. 1091–1096, doi: 10.1109/ICCMC51019.2021.9418254.
[22] M. M. Islam, M. A. Uddin, L. Islam, A. Akter, S. Sharmin, and U. K. Acharjee, "Cyberbullying detection on social networks using machine learning approaches," in *2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering, CSDE 2020*, 2020, pp. 5–10, doi: 10.1109/CSDE50874.2020.9411601.
[23] T. H. Teng and K. D. Varathan, "Cyberbullying detection in social networks: A comparison between machine learning and transfer learning approaches," *IEEE Access*, vol. 11, pp. 55533–55560, 2023, doi: 10.1109/ACCESS.2023.3275130.
[24] K. Siddhartha, K. R. Kumar, K. J. Varma, M. Amogh, and M. Samson, "Cyber bullying detection using machine learning," *2022 2nd Asian Conference on Innovation in Technology, ASIANCON 2022*, pp. 1–4, 2022, doi: 10.1109/ASIANCON55314.2022.9909201.
[25] H. Hosseinmardi, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Prediction of cyberbullying incidents in a media-based social network," *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2016*, no. August, pp. 186–192, 2016, doi: 10.1109/ASONAM.2016.7752233.

BIOGRAPHIES OF AUTHORS

Rakesh Bapu Dhumale is an associate professor in the Department of Electronics and Telecommunication Engineering at AISSMS Institute of Information Technology, Pune, with over 16 years of academic experience. He holds a Ph.D. in electronics and telecommunication engineering and has published 97 international journal papers, 22 conference papers, 8 Copyrights and 6 patents. He specializes in artificial intelligence and machine learning. He has secured research funding exceeding ₹24.8 lakh and contributed as technical program chair for IEEE ESCI conferences. He can be contacted at email: rbd.scoe@gmail.com.

Ajay Kumar Dass has been in defense for 36 years after graduating in electrical engineering. Parallelly, he has been closely associated with academic field with about ten papers to his credit in last three years, published in international journals. He has 18 years of experience in defense production as QA expert. He is M.Tech. in power electronics and industrial drives followed by a doctorate in electronics engineering with his research work in AI field. Currently he is working as vice president in a power generation company. He can be contacted at email: ajaykdassdef@gmail.com.

Amit Umbrajkaar is associate professor in Department of Mechanical Engineering at School of Engineering, D Y Patil International University, Pune with over 27 years of academic experience. He holds Ph.D. in mechanical engineering and has published 19 papers in international journals and international conference. He has published 06 patents and 5 copy rights. His core area of research work are viz. system design, vibration analysis, implementation of ANN and machine learning in fault diagnosis domain. He can be contacted at email: ameethumbrajkaar@gmail.com.

Pradeep Mane BE, ME (E&TC), Ph.D., is the principal at AISSMS IOIT, Pune. With 86 journal papers, 66 conference publications, and 11 co-authored engineering books, his contributions are notable. He has guided 7 Ph.D. scholars as a recognized guide at SPPU and Bharati Vidyapeeth. A fellow of IEI, IETE, and a member of IEEE, ISA, and ISTE, he has received awards such as best principal (SPPU, 2021) and national awards from ISTE and CSI. He was CO-PI for the ISRO-UOP research grant and has served on the BOS for Electronics Faculty at SPPU and Bharati Vidyapeeth. He can be contacted at: pbmane6829@gmail.com.