Cyberbullying Detection using Machine Learning and Deep Learning

Aljwharah Alabdulwahab[1], Mohd Anul Haq[2*], Mohammed Alshehri[3]
Department of Information Technology, College of Computer and Information Sciences,
Majmmah University, Al-Majmaah , Saudia Arabia.[1, 3]
Department of Computer Science, College of Computer and Information Sciences,
Majmaah University, Al-Majmaah, Saudi Arabia[2]

***Abstract*—With the human passion for gaining knowledge, learning new things and knowing the news that surrounds the world, social networks were invented to serve the human need, which resulted in the rapid spread and use among people, but social networks have a dark and bright side. The dark side is that strangers or anonymous people harass some users with obscene words that the user feels wrong about, which leads to psychological harm to him, and here we try to discover how to discover electronic bullying to block this alarming phenomenon. In this context, the utility of Natural Language Processing (NLP) is employed in the present investigation to detect electronic bullying and address this alarming phenomenon. The machine learning (ML) method is moderated based on specific features or criteria for detecting cyberbullying on social media. The collected characteristics were analyzed using the K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Naive Bayes (NB), Decision Trees (DT), and Random Forest (RF) methods. Naturally, there are test results that use or operate on the proposed framework in a multi-category setting and are encouraged by kappa, classifier accuracy, and f-measure standards. These apparent outcomes show that the suggested model is a valuable method for predicting the behavior of cyberbullying, its strength, and its impact on social networks via the Internet. In the end, we evaluated the results of the proposed and basic features with machine learning techniques, which shows us the importance and effectiveness of the proposed features for detecting cyberbullying. We evaluated the models, and we got the accuracy of the KNN (0,90), SVM (0,92), and Deep learning (0,96).**

Keywords*—Cyberbullying detection; machine learning; deep learning; natural language processing (NLP), feature extraction; CNN

I. Introduction

Cyberbullying is a severe cybersecurity concern that constantly affects more people using social media and the Internet. Bullying is arguably hostile behavior displayed by a single individual or a group of people, who can only be present in certain places or at certain times of the day (such as during school hours) and can alternatively occur everywhere and at any time through electronic methods.

Cyberbullying was not taken seriously at the turn of the 20th century when social media, in general, and Internet usage were still in their infancy. At the time, the optimistic advice for dealing with cyberbullying was to "disconnect" or "turn off the screen [1, 2]. However, when the effects of online hate speech increase, these proposals lose their effectiveness. More than following the usual suggested cybersecurity standards and procedures is required to prevent cybercrime. 41% of American citizens reported experiencing online harassment personally in 2017, while 66% reported seeing discourse offensive to others. Additionally, about 50% of young individuals who use social media sites have been said to be various experience several kinds of cyberbullying. Popular social networking sites like Twitter are not immune to this menace.

The identification of cyberbullying has grown in importance as an NLP topic, cyberbullying detection's objective is like other NLP jobs. Entails preprocessing the text (like a tweet) and extracting critical information in a particular method. That enables the use of machine learning to understand and categorize each text. The classic methods for classifying text involve the use of a way to make text representation simpler, such as the bag-of-words (BoW) approach, then a machine learning classifier like the logistic regression (LR) or support vector machine (SVM) approach [3] (see Fig. 1).

Natural Language Processing (NLP), the field of artificial intelligence that focuses on the interaction between computers and human language, has made significant strides in identifying online abuse [3]. However, specific challenges persist, including limitations in accommodating long texts within the constraints of social media platforms, the imbalance between hostile and constructive comments, the inherent ambiguity of natural language, and the prevalent use of slang [4] in the past ten years, neural network-based models have outperformed conventional machine learning methods on several NLP tasks.

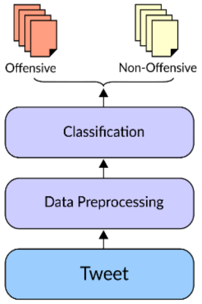

Fig. 1. The general framework of deep learning architecture.

[*]Corresponding Author

These NLP techniques rely on the successful use of word embeddings, specifically deep learning, and neural network dense vector representations. In contrast to conventional machine learning algorithm-based methods, which rely primarily on manually created characteristics that are viewed as insufficient Deep learning methods use multilevel automated feature representation, which is laborious, to distinguish the input. a model of a multilayer perceptron neural network in various NLP tasks, CNN-based models, and recurrent neural networks (RNN) have demonstrated promising outcomes and developed a novel character-based technique that combines a CNN model with an RNN architecture to categorize text. Long short-term memory (LSTM) categorizes phrases separately to learn text meaning; CNN then used word analysis to extract regional attributes.

Motivated by the notable results of numerous NLP research aiming to categorize extensive texts, deep learning systems have proved their effectiveness. This paper conducts a comprehensive examination of cyberbullying within social networks, employing advanced technologies and methodologies, including Natural Language Processing (NLP) and machine learning. In Section II, "Related Works," existing research is reviewed, offering insights into the techniques used to combat online harassment. Section III, "Materials and Methods," outlines the tools and strategies utilized, particularly NLP and machine learning. Section IV, "Computational Complexity," delves into the algorithms driving the cyberbullying detection system. Section V, "Results," presents the findings, emphasizing the method's high accuracy in predicting cyberbullying's impact on online social networks. Section VI, "Limitations and Future Scope," explores the study's boundaries and suggests future research directions. Finally, in Section VII, "Conclusions," the paper summarizes the implications of the research, shedding light on technology's role in enhancing online safety, and invites further exploration into these crucial issues.

II. Related Work

This section examines methods for detecting cyberbullying in online social networks (OSNs). Cyberbullying incidents were broken down into Race, sexual orientation, culture, and IQ are some of the categories that Dinakar et al. in [5]. They employed four distinct classifiers as a result (Naive Bayes (NB), grip with rules, J48 with trees, and SVM) to identify the comments submitted on several contentious YouTube videos as a use case. The dataset, which consists of over 50,000 words, consists of three sections: testing, validation, and training. The accuracy of Rule-based Jrip hasn't, however, exceeded 80% and this is a good percentage to reduce online bullying. A method to identify fine-grained cyberbullying techniques, such as insults and threats.

The authors referenced online bullying and content that was taken from Ask.FM website contains English and Dutch language features that are like OSNs. The authors divided the prospective participants between a cyberbullying discourse harasser, target, and onlooker—into three groups. Two groups were formed in the class: onlooker-defenders and bystander-assistants, who back the harasser while speaking up for the victim. The comments were then separated using SVMs. However, bullying idioms are harder to locate in the text on Twitter. One of the first to provide a technique to identify cyberbullying on the Twitter network was Sanchez. et al. [6]. The authors used Twitter abuse against a specific gender was identified using an NB classifier. The accuracy of their approach, though, was only 70%, and the amount of the dataset they employed was modest. To include a wide range of examples of cyberbullying, the abusive models should be applied generally rather than only to one topic. Using word2vec as a feature representation approach and both NB and RF classifiers, Saravanarj et al. [7] provided a broad framework to identify false positives and abusive tweets. The framework may extract demographic data about the perpetrators, including They, also mentioned age, name, and gender. The recommended techniques, though, cannot generate precise outcomes in comparison to complex machine learning techniques like deep learning. This study achieved results of 78%.

The authors Al-garadi et al. [8] conducted a study to detect cyberbullying on Twitter, which takes advantage of several unique features, which explicitly include activity, network, user, and tweeting on the Twitter platform. For classification purposes, these traits, and the samples they were associated with were entered into a machine-learning system. According to the authors' analysis of four machine learning algorithms, the RF method is superior to all other algorithms in the area under the receiver operating characteristic curve and f-measurement such as KNN, SVM, and NB only about 599 out of 10007 authors. The dataset included total tweets related to bullying. The Big Five models (including neuroticism, agreeableness, and extraversion) and the Dark Triad (including psychopathy) have been used and I have achieved scores of 60%. by Balakrishnan, Khan, and Arabnia, Balakrishnan et al. [9,10]. To analyze the personalities of Twitter users and gradually spot online cruelty. The proposed method aimed to look at the connection between personality factors and online bullying. The writers divided the tweets into four categories representing the user's behavior: bully, spammer, attacker, and everyday person. Then the writers classified each tweet into one of the types mentioned earlier using the random forest RF ensemble approach. Results from the suggested system using these personality factors were favorable. Although it was a modest number, the dataset included 5453 tweets that were gathered using the hashtag "Gamergate." Additionally, the tweets are more specialized than they should be about a particular community (using the hashtag "Gamergate").

A significant amount of Twitter comments were examined by Chatzako et al. [11,12] to identify features of abusive conduct. These tweets were from people who engaged in various conversations, including those on the NBA, the Gamergate scandal, and comments about television programs about female wages. Discrepancy on stations run by the British Broadcasting Corporation (BBC). The writers looked into several aspects taken from Twitter, including user attributes, network-based features, and tweets. They then experimented with several cutting-edge classification techniques to differentiate across user accounts and achieved an accuracy of 91%.

A deep learning detection technique was presented by Gamback et al. [13] To recognize Twitter cyberbullying comments. The method divided the remarks into four categories: non-offensive, racist, sexist, and all three (i.e., sexism and racism). The authors used the character four grams to represent text. Word2vec was also employed for semantic analysis authors. After that, the authors used one of these methods to condense the feature set: Features of un-layer CNN (i.e., max-pooling layer). and they used a SoftMax method to classify each tweet as a result. When tested utilizing cross-validation by ten, the suggested approach had an F-score of 78.3%. Six thousand six hundred fifty-five tweets make up the datasets that the authors used to identify cyberbullying in Twitter comments.

Pradhan et al. [14] explored two deep learning architectures as well as a neural network model. The CNN-LSTM and CNN-BiLSTM Deep learning architectures and neural networks are two examples. Deep learning both methods yielded encouraging results, with an accuracy rate of about 92%. The efficiency of models for self-attention (these models obtained cutting-edge results in a variety of machine translation tasks) by Pradhan et al. [14], The detection of cyberbullying was investigated with the cyberbullying datasets from Form Spring, Wikipedia, and Twitter The application of the transformer architecture paradigm for self-attention was examined by the authors. A multiheaded self-attention layer was employed in this architecture to replace the recurrent layers that were utilized for encoding and decoding, the results from the suggested method were satisfactory.

Agrawal et al. [15] experimentally showed through a framework that this method could get around some of the drawbacks of previous approaches, such as limiting the limitation of hate speech identification to a specific category (i.e., cyberbullying) and using custom features available through traditional machine learning methods. To overcome these restrictions, the authors investigated four deep learning architectures: BiLSTM with an attention layer, CNN, and LSTM. Additionally, the authors categorize assault, bullying, racism, and sexism are four criteria used to classify hate speech on the social internet. Furthermore, they used transfer learning to apply the information gained from a deep understanding of two datasets, one of which was remarkably similar, extensive tests were conducted on the researched architectures using the Twitter, Wikipedia, and Formspring datasets. Around 16 thousand tweets from Twitter were used by the authors Pradhan et al. and Agrawal et al. [14,15] to identify social media cyberbullying using Spanish-language content, Plaza et al. [16] have devised a method. The authors investigated a few deep-learning algorithms to detect hate speech in Spanish. They were enhancing performance. The authors' deep learning models were trained. Specifically, used Transfer learning is used to address a few sample problems. Additionally, the authors evaluated how well SVM and LR are examples of standard machine learning techniques. Compared models of deep learning that have been pre-trained include the enhanced-BERT, LSTM, CNN, and LSTM. The trials demonstrated that using BERT techniques and pre-trained models together enhanced performance in terms of accuracy in comparison to other deep learning and conventional models. Table I shows the comparison study of literature Review.

TABLE I. COMPARISON STUDY OF LITERATURE REVIEW

Authors	APP	Methods	Limitation	Accuracy
[5]	YouTube	Jrip, J48, NB, and SVM with rules	A small dataset with poor precision	80%
[6]	Twitter	Classification NB	A small dataset with poor precision	70%
[7]	Twitter	NB and random forests	Not reported	78%
[8]	Twitter	KNN, RF, NB, and SVM	The dataset is not that large	60%
[9,10]	Twitter	J48, RF, and NB	Constrained to a certain demographic with a fairly small dataset	60%
[11,12)	Twitter	A probabilistic artificial neural network and ensemble	The dataset is not that large	91%
[13]	Twitter	CNN uses SoftMax	The dataset is quite compact	78.3%
[14]	Twitter, Wikipedia, and Formspring	CNN, LSTM, and SVM	There is a minimal amount of data	92%
[15]	Twitter, Wikipedia, and Formspring	CNN, LSTM, and SVM	The dataset is quite compact	92%
[16]	Twitter	BERT techniques and trained models	The data collection is not that large	95%

Motivated by the notable results of numerous NLP research aiming to categorize extensive texts, deep learning systems have proved their effectiveness. We look into the possibility of recognizing short sentences utilizing the multichannel deep learning model.

III. MATERIALS AND METHODS

A. Dataset Description

The tweets dataset used in this study was sourced from [2] In two columns, the data includes tweets and class. The dataset has two types: cyberbullying and not cyberbullying. There were 47692 tweets utilized in total to create this dataset.

1) Exploratory data analysis: The table description of the Cyberbullying dataset. Table II displays the percentage of cyberbullying incidents, 38000 tweets however the percentage of cyberbullying incidents is 47692 tweets.

TABLE II. CYBERBULLYING DATASET

#	Column	Non-null Count	Dtype
0	Tweet_text	47692 non-null	Object
1	Cyberbullying_type	47692 non-null	Object

According to Table II, the dataset's two columns are object types. In this dataset, 36 tweets were found to be duplicates; these tweets were eliminated. Eliminate contractions, emojis, new line characters, links, and stop words.

Remove just the "#" sign to maintain the center of the sentence's # hashtags while eliminating the # hashtags at the conclusion & and $ are filtered special characters that are present in some words. The words were also reduced in complexity via stemming. Stemming is a technique used to strip away the suffixes, prefixes, and other word elements of a given word until only its root or lemma remains. This approach is useful in NLP [17](see Fig. 2).

It was observed in Fig. 3 that the tweets (including very long tweets) ethnicity has reached less than 100. While in Fig. 4, the tweets (excluding very long tweets) ethnicity has reached less than 100, we conclude that most of the words in the tweets are bullying ethnicity and we must detect and limit these bad tweets to reduce bullying.

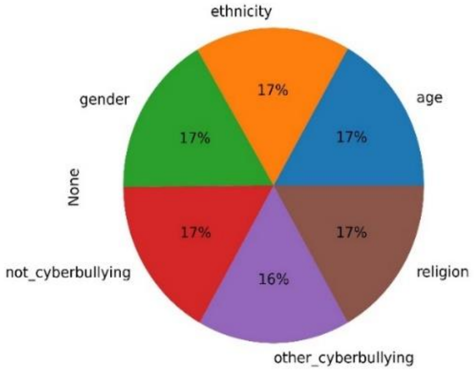

Fig. 2. The proportion of cyberbullying.

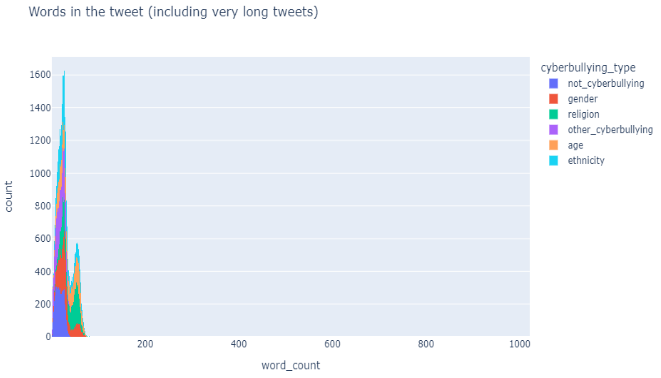

Fig. 3. Words in the tweets, including very long tweets.

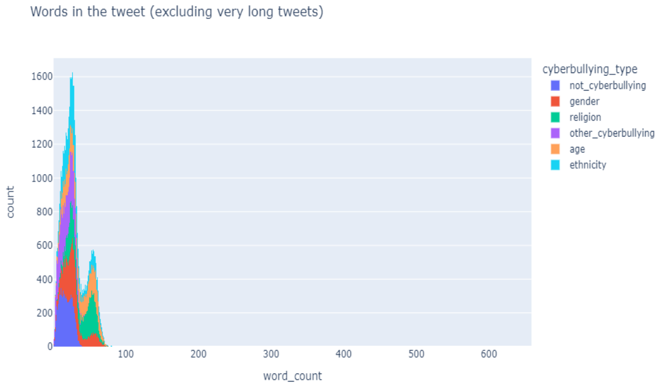

Fig. 4. Words in the tweets, excluding very long tweets.

In the Fig. 5 shows the percentage of ages that are exposed to cyberbullying, which schools, where the percentage reached less than 9000, unlike kids, which shows that the percentage is less than 1000, and here it shows us what ages are affected by cyberbullying.

Fig. 6 shows the religion that is most exposed to cyberbullying, which is the Muslim, with a percentage of less than 5,000, in contrast to radical which shows a percentage is less than 2,000.

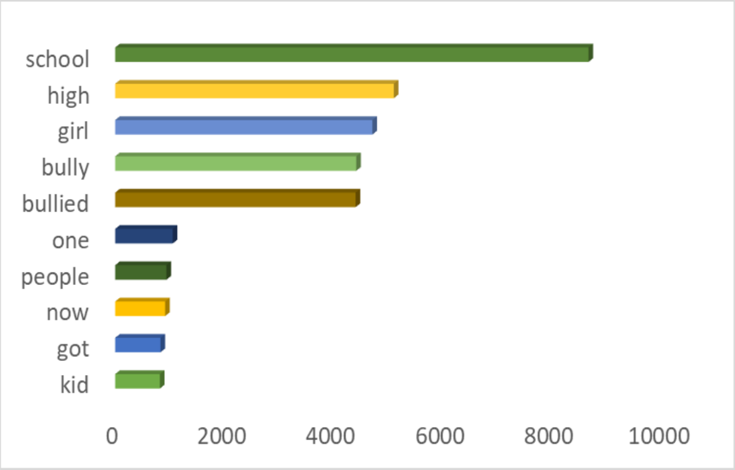

Fig. 5. Age-based cyberbullying.

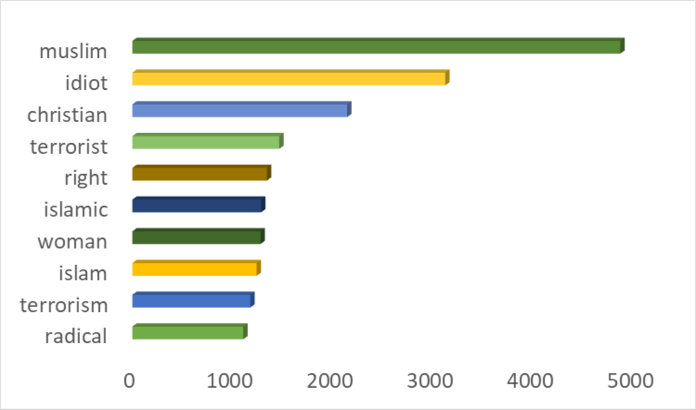

Fig. 6. Religion-based cyberbullying.

B. Evaluation Metrics

We also know that classification accuracy is the natural choice for statistics because identifying cyberbullying is a classification task, but we have a problem, which is the inequality in class for determining to cyberbully, and accuracy is not a reliable indicator, as well as retrieval and accuracy, as well as the F1 scale. To obtain all measures (TN) a matrix containing four categories of false positive (FP), false negative (FN), true positive (TP), and true negative values is used [18,19].

The fraction of correctly classified instances relative to all instances is known as accuracy. The Eq. (1) that follows determines it.

$$\text{Accuracy}=\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (1)$$

The recall is the proportion of correctly categorized examples that are positive when compared to instances that belong to the actual class. The Eq. (2) that follows determines it.

$$\text{Recall}=\frac{\text{TP}}{TP+TF} \quad (2)$$

The fraction of accurately categorized positive events relative to all projected positive instances is known as precision. The Eq. (3) that follows determines it.

$$\text{Precision}=\frac{TP}{TP+FP} \quad (3)$$

The F1 measure combines recall and precision, making it useful in situations when both are crucial. The Eq. (4) that follows determines it.

$$\text{F1measure}= 2*\frac{(precision*recall)}{\text{precision + recall}} \quad (4)$$

C. Methodology

1) Feature extraction: The sklearn package’s CountVectorizer (CV), which has a 2500 feature maximum, was utilized to extract features. Cyberbullying type was used as a label, and a CV was added as a feature to the original tweet text. The label Y’s size was 46017X6, and the feature X’s was 46017X2500. With a ratio of 75:25 for training and testing, the data was divided using the sklearn train test split. The sklearn StandardScaler was used to conduct the feature scaling. The dataset was split into training and testing with a ratio of 75:25, therefore the training set contains 34512 records.

2) KNN: The k-nearest neighbor’s algorithm is a supervised learning classifier that makes predictions or classifications about how a single data point will be grouped using proximity. Nevertheless, it can be used to address classification or regression problems. KNN compares data points based on how close or far apart they are from the query points. There are several ways to compute distance; one of the most popular is the Euclidean distance formula in Eq. (5).

$$\text{d (x, y)} = \sqrt{\sum_{i=1}^{n}(yi-xi\,)^2} \quad (5)$$

A straight line connecting the available location and the query point is measured by Euclidean distance.

KNN has the benefits of being simple to use, adaptable, and requiring fewer hyperparameters, but it also has memory and overfitting problems.

3) Support vector machine: Support Vector Machine (SVM), a dependable classification and regression technique, improves a model’s projected accuracy while preventing overfitting the training set. SVM is especially effective for analyzing data that has thousands of predictor fields. By transforming the data into a high-dimensional subspace, SVM may categorize data points even when they are not linearly separable. After a class divider has been found, the data are transformed to allow for the separator’s hyperplane.

4) Deep learning model development: We have developed a 6-layer Deep Learning model trained on the Twitter dataset for the identification of cyberbullying 5.2. This investigation used Python 3.8, Keras API, and Single-GPU TensorFlow 2.0 backend (i9, 10900k, 128 GB, 2666 MHz RAM). First, the Twitter dataset has been preprocessed. An embedding layer with the parameters vocab size and input length processes the input. The Embedding layer searches the vocabulary’s integer encoding for each word’s embedding vector. While the model is being trained, these vectors are learned, and the output array gains a dimension thanks to the vectors, (Batch, Sequence, and Embedding) are the measures that are obtained). The batch normalization layer was used to speed up and improve the stability of training artificial neural networks by normalizing the inputs to the layers by re-centering and re-scaling. The second layer was a convolution layer. By applying a filter to information, convolutional neural networks create a feature map that summarizes the existence of features recognized in the input. A max pool layer is made up of the third layer. The most significant value found in each patch of each feature map is determined using a pooling technique called max pooling. The results are feature maps that have been down-sampled or pooled, with the focus being placed on the feature that is most common in the patch as opposed to its average presence, as in the case of average pooling. A flattened layer, the fourth layer, condenses the input’s spatial dimensions to only its channel dimension. For instance, the coating will produce a flattened (H*W*C)-by-N-by-S array if given an H-by-W-by-C-by-N-by-S array as input. The dense layer was the last two layers.

The dataset is categorized using a thick layer based on the output of the preceding layers. A non-linear function known as an "activation function" is applied to the weighted average of the input that the neurons in each neural network layer calculate. The initial dense layer with L1L2 regulators employed the ReLu activation function, while the final layer with the softmax function was used for classification. Fig. 7 displays the CNN-LSTM architecture that has been suggested for use in malware detection.

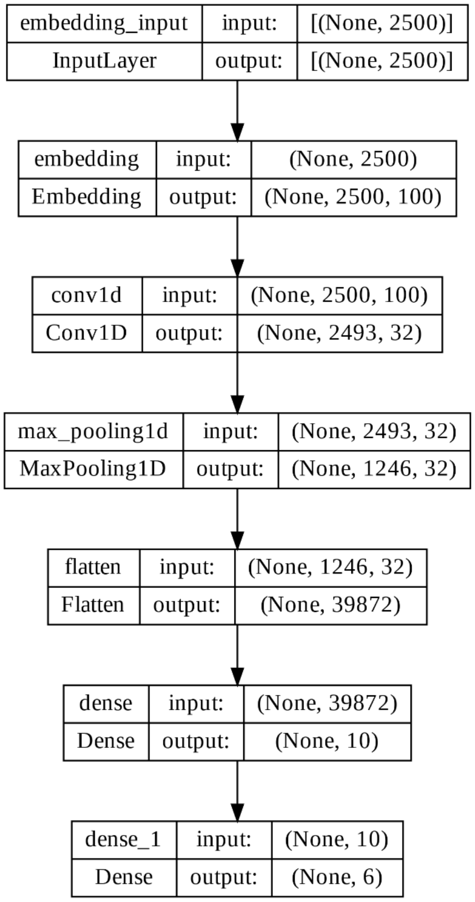
embedding_input | input: | [(None, 2500)]
InputLayer | output: | [(None, 2500)]
embedding | input: | (None, 2500)
Embedding | output: | (None, 2500, 100)
conv1d | input: | (None, 2500, 100)
Conv1D | output: | (None, 2493, 32)
max_pooling1d | input: | (None, 2493, 32)
MaxPooling1D | output: | (None, 1246, 32)
flatten | input: | (None, 1246, 32)
Flatten | output: | (None, 39872)
dense | input: | (None, 39872)
Dense | output: | (None, 10)
dense_1 | input: | (None, 10)
Dense | output: | (None, 6)

Fig. 7. Proposed deep learning architecture for cyberbullying detection.

The model compilation followed the layer addition. An optimizer, a loss function, and a metric function are required to evaluate the model's validity during compilation. The stochastic gradient descent variant Adam optimization technique was used for optimization. It has several benefits, such as fewer memory requirements and quick calculations. Given that the label comprises six classes, the categorical cross-entropy loss function was utilized to quantify the error rate between the actual and values for binary classification.

IV. Computational Complexity

Accuracy was then employed as an evaluation parameter. (6,739,628 params altogether, Trainable params: 6,739,628, non-trainable parameters: zero) as shown in Table III.

TABLE III. Number of Trainable and Non-Trainable Parameters for the CNN-LSTM Modle

Layer Type	Output Shape	#Params
embedding_1 (Embedding)	(None,2500, 100)	6315200
conv1d_1 (Conv1D)	(None,2493, 32)	25632
max_pooling1d_1 (MaxPooling 1D)	(None, 1246, 32)	0
flatten_1 (Flatten)	(None,39872)	0
dense_2 (Dense)	(None,10)	398730
dense_3 (Dense)	(None,6)	66

In Table III the embedding layer has 6,315,200 parameters, which are used to map each of the 2,500 input tokens to a 100-dimensional vector. The complexity of this layer is O (batch_size * input_length * embedding_size) = O(32 * 2500 * 100) = O(8e7). The 1D convolutional layer has 25,632 parameters and requires batch_size * 2493 * 32 multiply-adds to produce its output. The complexity of this layer is O (batch_size * input_length * kernel_size * num_filters) = O (32 * 2493 * 3 * 32) = O(7.5e6). The max pooling layer simply reduces the output size by a factor of 2, so it has negligible computational complexity. The flattening layer has no parameters and simply reshapes the output of the previous layer, so it has negligible computational complexity. The first dense layer has 398,730 parameters and requires batch_size * 39872 multiply-adds to produce its output. The complexity of this layer is O (batch_size * input_size * output_size) = O (1.27e9). The second dense layer has 66 parameters and requires batch_size * 10 multiply-adds to produce its output. The complexity of this layer is O (batch_size * input_size * output_size) = O (3.2e3). The overall computational complexity of the given model can be approximated as O (1.36e9). This means that the computational cost of training and inference for this model grows linearly with the size of the input data. Specifically, the dominant factors contributing to the computational complexity of this model are the number of parameters in the embedding and dense layers, as well as the size of the input and output data for each layer.

V. Results and Discussions

Here we review the accuracy results that were applied through the methods (KNN, SVM, Deep Learning) as shown in Table IV.

We note here that KNN gave fewer results than SVM and DL because of the approximation of the basic distribution of the data in a parametric way, and SVM assumes the existence of a super level that separates the data points from DL, as it is known that it works like a human neural network, as it is an effective classifier for cyberbullying detection tasks to extract text from how to create a 6-layer model.

Based on experiments, the deep learning method shows better results in accuracy for detecting cyberbullying than other methods (KNN, SVM), as deep learning is an effective classifier for cyberbullying detection tasks to extract text by creating a 6-layer model, which was A Twitter dataset drill was one GPU (i9, 10900K, 128GB 2666MHz RAM) running with Python 3.8, Keras API and Tensorflow 2.0 backend. The result indicated that deep learning achieved good state-of-the-art results on cyberbullying, and it is important to consider and analyze all results from all experiments.

To achieve our experiments, we applied a test to determine the appropriate resolution to deliver the best results for our model. Results using a deep learning algorithm to detect cyberbullying, in the first layer is embedding, which is the input processing and forms the output from (none, 2500, 100) and parameters 6315200 was made, the second layer is a wrapping layer and forms the output from (none, 2493, 32) and parameters 25632 was made, and the third layer takes the highest value found in Each patch and the results are reduced or aggregated feature maps that focus on the most prevalent feature in the patch rather than its average presence and form the output from (none, 1246, 32) parameters 0, the fourth layer is a flat layer, where the flat layer minimizes The spatial dimensions of the inputs to their channel dimensions and form the output from (39872, none) and parameters 0 were made, and in the last two layers is the dense layer to enrich the data set by calculating the inputs of neurons in each layer and form the output from the first dense layer (none, 10) Parameters 398730 was created, the output was created from the second dense layer (none, 6), and parameters 66 was created.

It was noted that the accuracy (KNN) was (0.90), which means that the model correctly predicted 90% of the comments classified as cyberbullying. Let's assume that we have tweets and there is bullying. We will discover this bullying by asking about some characteristics or types of words. Here, the discovery is classified or programmed, so that it identifies bullying words and the way KNN works. We now have a word, and it has two properties (word, bad word) Can it be predicted for its classification, we determine the value of the variable that will express the number of neighbors k and let its value be k=3, we calculate the value of the distance through the Eq. (6).

$$\mathrm{d}\ (\mathrm{x}, \mathrm{y}) = \sqrt{\sum_{i=1}^{n}(yi - xi\,)}^2 \qquad (6)$$

Table IV represents the comparison of the present investigation with the existing studies used tweets datasets. Table IV. offers a valuable comparison with other studies in terms of accuracy and limitations or future scope. This table reinforces the significance of present research and the need for continuous improvement in the field of cyberbullying detection.

TABLE IV. COMPARISON WITH THE OTHER STUDIES

Studies	Methods	Limitations/future scope	Accuracy
[6]	Classification NB	A small dataset with poor precision	70%
[7]	NB and random forests	Not reported	78%
[8]	KNN, RF, NB, and SVM	The dataset is not that large	60%
[9,10]	J48, RF, and NB	Constrained to a certain demographic with a fairly small dataset	60%
[11,12]	A probabilistic artificial neural network and ensemble	The dataset is not that large	91%
[13]	CNN uses SoftMax	The dataset is quite compact	78.3%
[14]	CNN, LSTM, and SVM	There is a minimal amount of data	92%
[15]	CNN, LSTM, and SVM	The dataset is quite compact	92%
[16]	BERT techniques and trained models	The data collection is not that large	95%
Present Study	KNN	Will add more data as future endevour	90%
Present	SVM	Will add more data as future endevour	92%
Study	DL Model	Will add more data as future endevour	96%

Through this method, we can limit or meet the words of cyberbullying. Fig. 8 shows us the accuracy ratio between the methods (KNN= 0.90, SVM= 0.92, Deep Learning= 0,97), and here it is clear to us that Deep Learning is higher in accuracy than the rest.

Fig. 9 shows us the high accuracy of training over the ages, and that the validation accuracy rises with the exercise of accuracy as a direct relationship. The higher the accuracy of training, the higher the validation of accuracy, as well as the training loss and the loss of accuracy the graph shows us that the indicator is going down over the ages, the worse the training accuracy, the greater the loss.

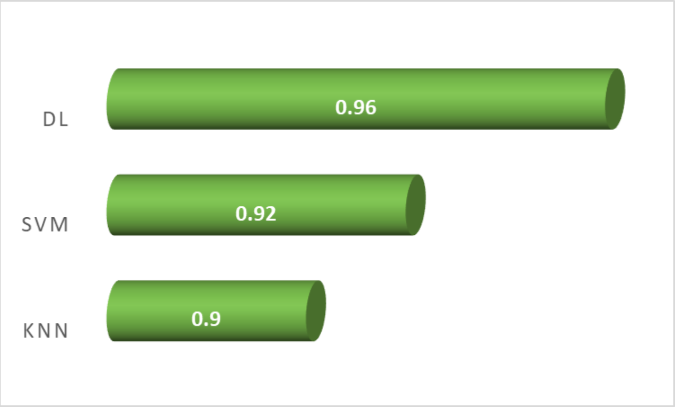

Fig. 8. The selected method and their accuracies.

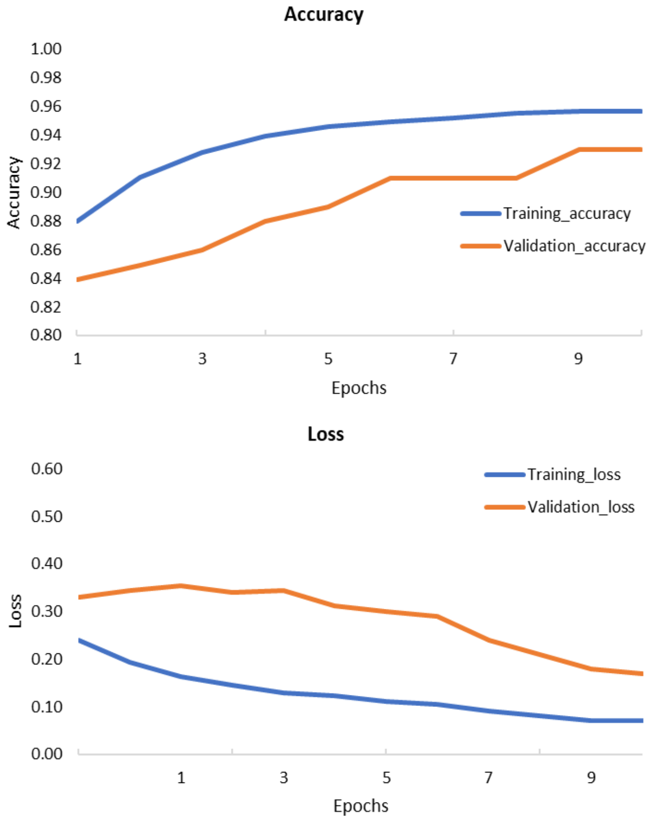

Fig. 9. Accuracy level through the epochs.

VI. Limitations and Future Scope

As it is known that the KNN algorithm is simple and easy to understand, and the result of this ease is that it does not provide predictions for words that are not circulated or known among people because it has no idea about it, so we must know the bad words and update them between periods to update the bad words, and the KNN algorithm needs. A large memory to store a group of words or data to predict them that may be used in tweets, which requires that the memory be very large to store words [20]. SVM works by separating data, but it is not suitable for a large group of data, because it is not implemented well if the data is overlapping, and the kernel greatly affects the way SVM works, which must choose the best kernel that fits the data, as there is another limitation is that it uses a lot of memory, as it needs to store the kernel matrix, which can be large for the data set, and also SVM lacks a probabilistic interpretation to make a decision, which is a defect in some applications, and the last limitation we have is that SVM is sensitive to choosing parameters as it is difficult to determine the best parameter values for a set of data [21]. Deep learning faces a limitation, which is that it understands the words received in the training data, as it is possible that the person does not know all terms or words in all dialects, which affects the mechanism of deep learning, and it is known to use deep learning models devices. These devices help reduce time and increase Effectiveness, however, are very expensive and consume a lot of energy. One of the limitations of deep learning is the method of learning. Sometimes the process may be disrupted. If the method is low, it is difficult to find a solution [18, 21].

We intend to use more data when we implement our method. We believe expanding our sample will enhance our approach performance. Large data sets are necessary for deep learning algorithms to work effectively. We’ll also attempt to expand the suggested structure by including numerous channels. The framework’s performance might be enhanced by employing more media when using a large dataset. The weights and other parameters of deep and massive neural networks can be improved with a large dataset [22,23]. Additionally, we intend to test our suggested framework using tweets in several languages.

Looking ahead, promising avenues include the Reliable Architecture-Oblivious Error Detection (RAED) algorithm, which enhances the reliability of computer systems and contributes to user and system well-being [24-26]. SHA-3's role in data integrity verification and the Advanced Encryption Standard (AES) indirectly aid in preventing cyberbullying by ensuring data privacy and creating a safe online environment. Additionally, leveraging algorithms like SIKE and CSIDH enhances security and privacy on social media, ultimately contributing to a safer digital landscape and protection against cyberbullying [27-30].

VII. Conclusions

The importance of online social networking has increased a part of our daily lives as it makes it easier to engage with others. However, detecting cyberbullying is an important topic to be examined due to the emergence of antisocial behavior faced by social media users due to issues such as hate speech, trolling, and cyberbullying on platforms such as Twitter and other social media experiences, which undoubtedly affects the psyche of victims. Social networks have become within the reach of everyone, especially children, and when children are exposed to electronic bullying, this affects the building of their personality and psyche. In this paper, reliable methods for detecting cyberbullying are presented. It shows how deep learning works and its high accuracy for detecting cyberbullying, and achieved the highest result, (0.96), by anticipating bad words, which helps reduce the spread of bullying in the means of communication, and as we mentioned another method, which is SVM, the result was good, which is (0.92). , but it is considered less than Deep learning, which affects the discovery of cyberbullying, and we also mentioned another method, which is KNN, which was the lowest result, as it achieved (0.90). The results show us the importance of this research and reliable methods to reduce the spread of cyberbullying.

Funding Statement

The authors would like to thank the Deanship of Scientific Research at Majmaah University for supporting this work under Project No. PGR-2023-659.

Acknowledgment

The authors would like to thank the Deanship of Scientific Research at Majmaah University for supporting this work under Project No. PGR-2023-659.

References

[1] M. U. S. Khan, A. Abbas, A. Rehman, and R. Nawaz, “HateClassify: A Service Framework for Hate Speech Identification on Social Media,” IEEE Internet Comput., vol. 25, no. 1, pp. 40–49, Jan. 2021, doi: 10.1109/MIC.2020.3037034.

[2] J. Wang, K. Fu, and C. T. Lu, “SOSNet: A Graph Convolutional Network Approach to Fine-Grained Cyberbullying Detection,” Proc. - 2020 IEEE Int. Conf. Big Data, Big Data 2020, pp. 1699–1708, Dec. 2020, doi: 10.1109/BIGDATA50022.2020.9378065.

[3] M. A. Haq, M. Abdul, R. Khan, and T. Al-Harbi, “Development of PCCNN-Based Network Intrusion Detection System for EDGE Computing," Computers, Materials & Continua”, vol.71, no.1, 2022 doi: 10.32604/cmc.2022.018708.

[4] O. Gencoglu, “Cyberbullying Detection with Fairness Constraints,” IEEE Internet Comput., vol. 25, no. 1, pp. 20–29, Jan. 2021, doi: 10.1109/MIC.2020.3032461.

[5] K. Dinakar, R. Reichart, and H. Lieberman, “Modeling the detection of textual cyberbullying,” AAAI Work. - Tech. Rep., vol. WS-11-02, pp. 11–17, 2011, doi: 10.1609/icwsm.v5i3.14209.

[6] H. Sanchez and S. Kumar, “Twitter Bullying Detection Knowledge Maps (KMs) View project The Vesperin System View project Twitter Bullying Detection”, Accessed: Mar. 29, 2023. [Online]. Available: https://www.researchgate.net/publication/267823748

[7] A. Saravanaraj, J. I. Sheeba, and S. P. Devaneyan, “Automatic Detection of Cyberbullying From Twitter,” IRACST-International J. Comput. Sci. Inf. Technol. Secur., vol. 6, no. 6, pp. 2249–9555, 2019, [Online]. Available: https://www.researchgate.net/publication/333320174

[8] M. A. Al-Garadi, K. D. Varathan, and S. D. Ravana, “Cybercrime detection in online communications: The experimental case of cyberbullying detection in the Twitter network,” Comput. Human Behav., vol. 63, pp. 433–443, Oct. 2016, doi: 10.1016/J.CHB.2016.05.051.

[9] V. Balakrishnan, S. Khan, T. Fernandez, and H. R. Arabnia, “Cyberbullying detection on twitter using Big Five and Dark Triad

features," Pers. Individ. Dif., vol. 141, no. January, pp. 252–257, 2019, doi: 10.1016/j.paid.2019.01.024.

[10] V. Balakrishnan, S. Khan, and H. R. Arabnia, "Improving cyberbullying detection using Twitter users' psychological features and machine learning," Comput. Secur., vol. 90, p. 101710, 2020, doi: 10.1016/j.cose.2019.101710.

[11] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on Twitter," WebSci 2017 - Proc. 2017 ACM Web Sci. Conf., pp. 13–22, Jun. 2017, doi: 10.1145/3091478.3091487.

[12] D. Chatzakou et al., "Detecting Cyberbullying and Cyberaggression in Social Media," ACM Trans. Web, vol. 13, no. 3, Oct. 2019, doi: 10.1145/3343484.

[13] B. Gambäck and U. K. Sikdar, "Using Convolutional Neural Networks to Classify Hate-Speech," Proc. Annu. Meet. Assoc. Comput. Linguist., pp. 85–90, 2017, doi: 10.18653/V1/W17-3013.

[14] A. Pradhan, V. M. Yatam, and P. Bera, "Self-Attention for Cyberbullying Detection," 2020 Int. Conf. Cyber Situational Awareness, Data Anal. Assessment, Cyber SA 2020, Jun. 2020, doi: 10.1109/CYBERSA49311.2020.9139711.

[15] S. Agrawal and A. Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 10772 LNCS, pp. 141–153, 2018, doi: 10.1007/978-3-319-76941-7_11/COVER.

[16] F. M. Plaza-del-Arco, M. D. Molina-González, L. A. Ureña-López, and M. T. Martín-Valdivia, "Comparing pre-trained language models for Spanish hate speech detection," Expert Syst. Appl., vol. 166, p. 114120, Mar. 2021, doi: 10.1016/J.ESWA.2020.114120.

[17] M. A. Haq, M. Abdul, and R. Khan, "DNNBoT: Deep Neural Network-Based Botnet Detection and Classification A ROBUST VIDEO STABILIZATION ALGORITHMS FOR GLOBAL MOTION ESTIMATION USING BLOCK MATCHING View project A Futuristic Analysis Approach of Neural Network for Intrusion Detection System. View project DNNBoT: Deep Neural Network-Based Botnet Detection and Classification", doi: 10.32604/cmc.2022.020938.

[18] M. Anul Haq and C. Author, "CNN Based Automated Weed Detection System Using UAV Imagery Coupling GPR Measurements and Modelling for Mountain Glacier Volume Assessment in India and Russia View project Understanding the Geomorphology of Martian Surface using MoM Datasets View project CNN Based Automated Weed Detection System Using UAV Imagery", doi: 10.32604/csse.2022.023016.

[19] M. Anul Haq and C. Author, "SMOTEDNN: A Novel Model for Air Pollution Forecasting and AQI Classification Modeling the snow properties for their classification and identification View project SMOTEDNN: A Novel Model for Air Pollution Forecasting and AQI Classification", doi: 10.32604/cmc.2022.021968.

[20] M. A. Haq et al., "Analysis of environmental factors using AI and ML methods," Sci. Reports |, vol. 12, p. 13267, 123AD, doi: 10.1038/s41598-022-16665-7.

[21] M. Anul Haq and C. Author, "CDLSTM: A Novel Model for Climate Change Forecasting Understanding the Geomorphology of Martian Surface using MoM Datasets View project Development of Glacial Lake Monitoring Techniques in The Uttarakhand Himalayas Using Geomatics Techniques View project CDLSTM: A Novel Model for Climate Change Forecasting", doi: 10.32604/cmc.2022.023059.

[22] M. Anul Haq, "DBoTPM: A Deep Neural Network-Based Botnet Prediction Model," 2023, doi: 10.3390/electronics12051159.

[23] M. A. Haq, M. A. R. Khan, and M. Alshehri, "Insider Threat Detection Based on NLP Word Embedding and Machine Learning," Intell. Autom. Soft Comput., vol. 33, no. 1, pp. 619–635, Jan. 2022, doi: 10.32604/IASC.2022.021430.

[24] M. M. Kermani and R. Azarderakhsh, "Reliable Architecture-Oblivious Error Detection Schemes for Secure Cryptographic GCM Structures," IEEE Trans. Reliab., vol. 68, no. 4, pp. 1347–1355, 2019, doi: 10.1109/TR.2018.2882484.

[25] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "Reliable hardware architectures for the third-round SHA-3 finalist Grøstl Benchmarked on FPGA platform," Proc. - IEEE Int. Symp. Defect Fault Toler. VLSI Syst., pp. 325–331, 2011, doi: 10.1109/DFT.2011.60.

[26] M. Mozaffari-Kermani and A. Reyhani-Masoleh, "A Structure-independent Approach for Fault Detection Hardware Implementations of the Advanced Encryption Standard," pp. 47–53, 2008, doi: 10.1109/fdtc.2007.15.

[27] B. Koziel, A. B. Ackie, R. El Khatib, R. Azarderakhsh, and M. M. Kermani, "SIKE'd Up: Fast Hardware Architectures for Supersingular Isogeny Key Encapsulation," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 67, no. 12, pp. 4842–4854, 2020, doi: 10.1109/TCSI.2020.2992747.

[28] M. Anastasova, R. Azarderakhsh, and M. M. Kermani, "Fast Strategies for the Implementation of SIKE Round 3 on ARM Cortex-M4," IEEE Trans. Circuits Syst. I Regul. Pap., vol. 68, no. 10, pp. 4129–4141, 2021, doi: 10.1109/TCSI.2021.3096916.

[29] A. Jalali, R. Azarderakhsh, M. M. Kermani, and D. Jao, "Towards Optimized and Constant-Time CSIDH on Embedded Devices," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 11421 LNCS, pp. 215–231, 2019, doi: 10.1007/978-3-030-16350-1_12.

[30] M. Mozaffari-Kermani, R. Azarderakhsh, and A. Aghaie, "Reliable and Error Detection Architectures of Pomaranch for False-Alarm-Sensitive Cryptographic Applications," IEEE Trans. Very Large Scale Integr. Syst., vol. 23, no. 12, pp. 2804–2812, 2015, doi: 10.1109/TVLSI.2014.2382715.