International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Deep Learning Model for Detecting Bullying Comments on Online Social Media

[1]Renetha J B, [2]Bhagya J, [3]Deepthi P S

Submitted: 05/02/2024 **Revised**: 13/03/2024 **Accepted**: 19/03/2024

Abstract: Youth dominate the online world today and the vast majority access social networks. Around the world, cyberbullying is rampant on social media sites and it has become a serious issue for people of all age groups. The bullying content detection by analyzing textual data in social media dataset is one of the most important parts of this work. The use of Deep Learning in Natural Language Processing has become very prevalent for handling the problem of cyberbullying. A large real-world Twitter dataset is collected for cyberbullying analysis. This work aims to analyze cyberbullying across the social media platform using a deep learning model Long Short-Term Memory Recurrent Neural Network or LSTM RNN and to evaluate its performance. The cyberbullying analysis on Twitter dataset using LSTM RNN gives an accuracy of 86%.

Keywords and Phrases: *Hate speech, bully, word embeddings, accuracy*

1. Introduction

Cyberbullying on social media has become a serious problem in recent years. It is simple to bully someone on any social media platform since accounts and actions are not monitored. Emails, texts, chat rooms, blogs, pictures, video clips, and text messages might all be used in cyberbullying. It is an act that exists where digital devices like smartphones, computers, and tablets are used. The individual who is bullied is a victim, while the one who engages in cyberbullying may be a bully. Cyberbullying, or the use of abusive language over the internet, has become a serious issue for people of all ages. It has a negative impact on human personalities, resulting in emotional and psychological problems. It has been linked to a person's psychological and physical health deterioration. As a result of these actions, the abuse of women and children has grown.

The cyberbullying research centre, in collaboration with Cartoon Network, conducted a study called "Tween Cyberbullying in 2020" [1] that looked at bullying and cyberbullying behaviours among 1,034 tweens in the United States. According to their research, over 80% of people have been the subject, perpetrator, or witness of bullying. Bullying at school is also experienced by half of the tweens. 15% had been cyberbullied. It has a detrimental influence on the sentiments of more than two-thirds of tweens who have been bullied. Almost a third claimed it had a negative impact on their friendships. Cyberbullying had an impact on 13.1 percent of people's physical health, and 6.5 percent said it had an impact on their education. The majority of tweens have their own smartphones, and nine out of ten (90%) have used one or more of the most popular social networking and gaming applications in the previous year.

Therefore, the topic of cyberbullying has become a widely reported topic in the media in recent years.People are able to make anonymous comments with very minimal identifying information. However, this liberty comes with a price. Vulnerabilities in social media platforms boost cyberbullying's effect [2]. Because social media encourages individuals to communicate in more indirect and anonymous ways, it provides anonymity for certain people, making them feel safer even when they engage in bullying. Its impact on numerous social media platforms cannot be overlooked and it needs careful monitoring to keep these actions under control. Most countries do not have a clearly defined legal framework to combat cyberbullying. With the increase in innovation, the web has been a battlefield for cybercrimes.

Cyberbullying detection is a critical Natural Language activity, and the first stage is to process the text, analyze it, and extract information based on the end objective.. For efficient identification of bullying remarks on social media, numerous machine learning techniques have been utilized [3]. The demand for scalable, automated cyberbullying detection systems has risen dramatically as a result of the

[1]*Computer Science and Engineering LBS Institute of Technology for Women Trivandrum, Kerala, 695014, India APJ Abdul Kalam Technological University Trivandrum, Kerala, India Lourdes Matha College of Science and Technology Trivandrum, Kerala, 695574, India renetha.jb@lmcst.ac.in*

[2]*Computer Science and Engineering, LBS Institute of Technology for Women, Trivandrum, Kerala,695014, India*

bhagyajayapal@gmail.com

[3]*Computer Science and Engineering, LBS Institute of Technology for Women, Trivandrum, Kerala,695014, India*

deepthisath@gmail.com

web's vast scope. Deep Learning plays a key role in detecting cyberbullying on social media platforms.

In classification, deep neural network technology improves feature extraction. It also helps in the text classification for cyberbullying analysis. Finding improper bullying terms and categorizing those communications are the two elements of a Deep Learning-based automated cyberbullying detection system. Deep learning is one of the successful techniques for learning from data and generating a model that can automatically classify appropriate action. It can help us discover a trend in bullies' vocabulary and, as a result, create a model to detect bullying.

The bullying content detection by analyzing textual data from social media is the objective of this work. The textual comments must be pre-processed, with the message first being transformed into a fixed-length vector and then classified. A pre-trained word embedding model or our custom embedding are utilized thereby the pre-processed words are transformed into vectors. Following that, the Deep Learning model is trained and then tested to detect bullying. Finally, a trained algorithm can recognize bullying messages in any fresh data. The goal of this study is to use the Twitter dataset to detect cyberbullying in social mediaUsing the LSTM RNN, this research will look into detecting bullying from English textual comments. Experiments were conducted on the Twitter dataset. The text data from the comments is pre-processed before being transformed into vectors using the pre-trained Glove embeddings. Finally, the data is trained using an LSTM classifier, and the results are assessed.

2. Related Works

With the increased use of smartphones and mobile apps, a more accessible type of cyberbullying has emerged. There is a lot of research being done in the subject of NLP right now, especially for the identification of cyberbullying. The application of several machine learning and deep learning algorithms in textual datasets for bullying detection has been reported in recent research.

The Convolutional Neural Networks (CNN) to Classify Hate-Speech [4] is a model proposed by B. Gamback and U. K. Sikdar in 2017. The model used word2vec [5] for word embedding and character n-grams. The CNN model performed better for hate speech detection. Word2vec does not produce different vectors for the same words which are used in different contexts. Word2Vec has the distributed representation in low dimensional space and it only creates sparse matrix.

The toxic comment classification [6] by Georgakopoulos et.al in 2018 used CNN for classification and embedding method, Bag-of-Words. The CNN model outperforms the machine learning techniques. When utilizing bag-of-words to model sentences, the sequence of the words in the phrase is ignored. The semantics of the term are ignored by a large number of word models. In order to detect inflammatory and hate speech in tweets from South Africa, in 2020, Oriola, Oluwafemi, and colleagues [7] presented a model. Logistic Regression, Random Forest, Gradient Boosting, and Support Vector Machine (SVM) were employed as classifiers. They think the most informative criteria for detecting abusive speech to be English slur terms. For hate speech, SVM with character n-gram had the highest true positive rate, whereas it had the lowest true positive rate for offensive speech.

Rosa and Hugo employed SVM, Logistic Regression, and Random Forests in a comprehensive study [8] for automatic cyberbullying detection in 2019. Using 10 textual 21 emotive characteristics and word embedding, features were extracted from the Formspring dataset. The characteristics of the users are not discussed in depth in the study. One of the difficulties they confront is a lack of high-quality data. Akhter et al. [9] used Nave Bayes, Bayes Network, Hoeffding Tree, J48, Reduced Error Pruning Tree, Random Tree, Random Forest, Logistic regression, additive logistic regression, OneR, and JRip in their work proposed in 2020.

Alhawarat et al. developed an improved Arabic text classification deep model [10] in 2020, which employed a Multi-Kernel CNN model with n-gram word embedding. They used SVM and Nave Bayes with Term Frequency Inverse Document Frequency called TF-IDF. [11]. The Deep Learning model outperforms the previous experiments in accuracy. Waseem et al. [12] offered 16k tweets dataset annotated for hate speech identification in 2016. The dataset is made up of tweets gathered over a two-month period. They gathered 16,914 tweets and annotated 3,383 for sexist material sent by 613 people, 1,972 for racist content sent by 9 users, and 11,559 for neither sexist nor racist content sent by 9 users. In 2021, Bhagya J and Deepthi PS [13] suggested detecting cyberbullying using SVM and TF-IDF.

Andrew M. et al. [14] developed a supervised sequence learning model combining CNN and LSTM on the Twitter dataset in 2017. For the purpose of data training utilizing the suggested technique, the authors recommend that LSTM-RNN be used instead of CNN and RNN. To identify sexism and racism, they employed TF-IDF values, Bag of Words Vectors over Global Vectors, task-specific embeddings Fast Text, CNNs, LSTMs, and Gradient Boosted Decision Trees. In 2018, Agrawal, Sweta [15] et al. published a model, which included the datasets Formspring, Twitter, and Wikipedia. Four machine learning models were used including SVM, LR,NB and Random forest. For cyberbullying detection, Bayes with Character n-grams and word unigrams features were deployed.

The model proposed by Badjatiya et al [16] in 2017 investigates the usage of deep learning methods for hate

speech detection and explores char n-grams, word TF-IDF values, Bag of Words Vectors over Global Vectors, and task-specific embeddings learned using LSTMs, FastText, and CNNs. Fang, Yong, and colleagues [17] presented a cyberbullying detection model in social networks utilizing bidirectional gated recurrent units (Bi-GRU) in 2021. They used the Twitter dataset to detect cyberbullying. The GRU with self-awareness produces greater results.

Wu et al. [18] presented a modified TF-IDF based fastText for detection. The position weight is applied to the TF-IDF, and the extracted keywords are being used as an input to get noisy data filtering. This help to enhance model accuracy. Inclusion of position weight enhanced the TF-IDF algorithm; and new algorithm extracts keywords and uses them as input to achieve the goal. To categorize the input data, they utilized fastText to build a binary classifier.

Learning the word embedding from scratch is a difficult task for two reasons - lack of training data and high number of trainable parameters. The proposed LSTM RNN classifier classifies the cyberbullying contents in the dataset provided.

3. Proposed Methodology

The proposed model for cyberbullying analysis on social media uses textual dataset. The data from real-world Twitter dataset is used. Collected data is then pre-processed in the data pre-processing stage. The data splitting is the next phase. LSTM [19], a kind of RNN, classifies bullying and non-bullying remarks after dividing the data into the train and test sets and (RNN). The Figure 1 shows various steps followed in the proposed method.

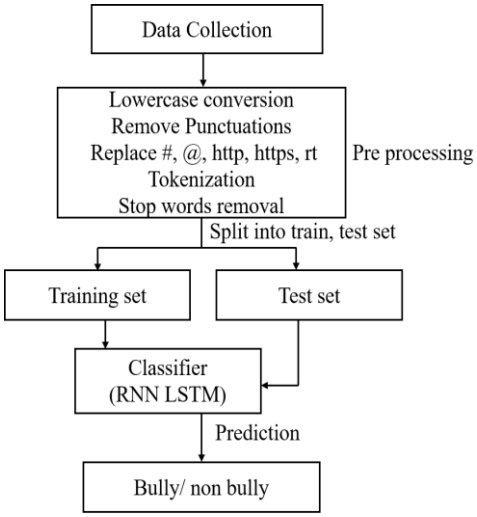

Fig 1: Proposed methodology

3.1 **Data Collection And Pre-Processing**

A large real-world dataset namely Twitter data is mainly used [12]. Twitter is one of the large microblogging platforms. The dataset addresses the topics of cyberbullying [20] such as sexism, racism. It includes more than 16K annotated tweets. Out of the 16K tweets, 1937 have been labelled racist, 3117 have been labelled sexist, and the rest have been labelled neither sexist nor racist. Twitter's dataset is a representative sample. It includes user ids, comments, and labels for sexism, racism, and none.

The pre-processing steps such as lowercasing the text data, removing punctuations, removing symbols such as hash tags that begins with # , the symbol used tro refer user, which starts with @ and tweets thar are again tweeted called re-tweets indicated using rt, http, https the hyperlinks, tokenization and stop words removal are done. All available data is converted to lowercase. The dataset contains symbols such as # and @. The hash tag is indicated by the # symbol, and the username is indicated by the @ symbol, followed by the username. Twitter retweets start with "rt" and hyperlinks start with http and https. These symbols have no effect on the detection of cyberbullying. As a result, at the data pre-processing step, these symbols are deleted.

Tokenization is basically the division of components or entire text document into smaller units, such as a single word or term. The word tokenization is used in the proposed model. It breaks a large sample of text into words. Stopwords are a set of words commonly used in a language. The insight behind removing stop words is that you can remove less informative words from the text and focus on the important words instead. These words do not affect the detection of cyberbullying.

A list of stopwords is maintained in the NLTK in Python in 16 distinct languages. With the help of NLTK, stop words in English are used in the work. After lowercasing the letters, punctuation removal, other symbol removal and stopwords removal, the word tokens are obtained.

After data pre-processing the word tokens are assigned to a word index. A vocabulary is generated with each token in a

sentence along with its word index. The padding of sequences is used to ensure that all sequences in the list are of the same length. This is done by adding zeros to the start or end of each sequence until each sequence is as long as the longest sequence in the dataset provided. To train a Neural network for NLP, you need the sequences of the same size to given as input to it. Each sequence then completes with the padding bit set to 0. Post-padding is done to produce a string of the same length. Here, padding is done by adding zeros to create the maximum length of the largest comment in the dataset.

3.2 Word Embedding

Dense word vectors, also known as word embedding, are a frequent and powerful technique to connect vectors with words. A word embedding is a form of word expression that allows similar-sounding words to be articulated in the same way. In a specified vector space, word is represented as a real-valued vector. The data from natural language is discrete. Word embedding is a small floating-point vector. With the help of word embedding, it computes a distance between the two vectors representing the two words and this distance between similar meaning words will be low with a good word embedding. Word embeddings may be obtained in one of two ways: either by learning the word embeddings for our task or by loading a pretrained word embedding model into your task. Pretrained word embeddings like Glove and Word2vec are the most popular.

One of the drawbacks of using the Word2Vec for generating word embeddings is that it does not take the occurrence statistic of a word in a large corpus. The Word2vec only consider the local context, it works according to the predefined window size which is previously defined. Stanford researchers created Global Vectors [21] for Word Representation in 2014. Its main is to consider the local statistics as well as global counts while generating word embeddings

The main objective of GloVe is to take in account the global statistics of words and keeping the linearity relationships between them. The factorization of a matrix of word co-occurrence statistics is used in this embedding approach. Rather than a full sparse matrix, Glove generates a vector space with significant substructures by simply training on non-zero items in a word-word global co-occurrence matrix. The Global Vectors were trained on 2 billion tweets with 27 billion tokens and 1.2 million unique words using 50-dimensional vectors. GloVe generates an embedding matrix that may be loaded into an Embedding layer. Glove creates a 50-dimensional embedding matrix for each word in the dataset once it is applied. If two words often appear within the same context, their meanings are strongly correlated with the GloVe embedding.

3.3 Data Splitting

The Corpus is split into two sets, train and test. The dataset splits into the ratio 80:20, where former is for training. In the proposed work a k-fold cross validation is applied for training and testing. Each iteration is usually named as folds, here setting the number of folds to 10 (k=10). From those 10 groups use one of them to test the model

3.4 Model Building

The LSTM classifier is used in proposed model to classify the bullying and non-bullying contents. The model is shown in Figure 2.

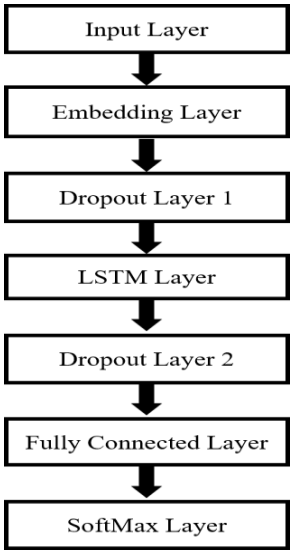

Fig 2 Model architecture

The Input layer receives a list of inputs. Each word obteined is tokenized and transformed to word embeddings. We may transform each word into specific sized vector of fixed length vector by using the embedding layer. The end result is a dense vector with real values. The embedding layer works with a set of words of a specific length. It starts

with weights which are randomized. Then it learns an embedding for training dataset word after word. By transforming the words to word embeddings, they are transformed into vectors of smaller dimension.

The major benefit of utilizing word embeddings is that they can capture context similarity and are quick and efficient for deep learning and NLP tasks owing to a reduced dimensionality. Instead of treating each word embedding separately, they are concatenated together into a single vector andis then sent to the hidden layer. The pre-trained word embedding model GloVe may be loaded using the Embedding layer. The GloVe embeddings are used to generate a 50-dimensional embedding matrix, which is then transferred to the next layer. To minimize overfitting, two drop out layers are applied before and after the LSTM layer. The dropout layer 1 has a dropout rate of 0.25. Using LSTM, the LSTM layer extracts uni-gram features at various locations in the input. The LSTM layer contains 50 units in this case.

The fully linked layer is the network's last layer, flattening and combining the high-level characteristics learnt by the preceding levels. The totally linked layer is also known as the thick layer. This layer also contains the regularization dropout layer. The dropout rate is set to 0.50 for layer 2. This layer's output is sent to the SoftMax output layer for prediction based on the number of classes (bullying and non-bullying in this case).Finally, the probability is calculated using the SoftMax activation function. This layer's output is sent to the SoftMax output layer for prediction. Just before the output layer, a neural network layer is used to implement SoftMax. They must both have the same number of nodes.

The probability is distributed throughout each output node using the SoftMax activation because it assigns the likelihood of belonging to a specific label in probabilities, in the range of 0 to 1, the softmax layer is the output layer. In the suggested model, the Adam optimizer is utilized, and a learning rate of 0.01 is used. Table1 lists all of the LSTM parameters utilized in the proposed model.

Table 1. LSTM parameters

Parameters used	Values
Number of units in LSTM layer	50
Number of dropout layers	2
Dropout rate of dropout layer1	0.25
Dropout rate of dropout layer2	0.50
Optimizer used	Adam
Activation function	SoftMax
Learning Rate	0.01

4. Results and Discussion

After pre-processing the Twitter dataset, the embedding matrix is created with the help of pre-trained GloVe embedding. The dataset is then divided in half at an 80:20 ratio and trained the dataset using the LSTM model. The accuracy-loss graph is obtained. The epoch is set to 20. Various performance measures were also evaluated. The accuracy loss graph is shown in Figure 3

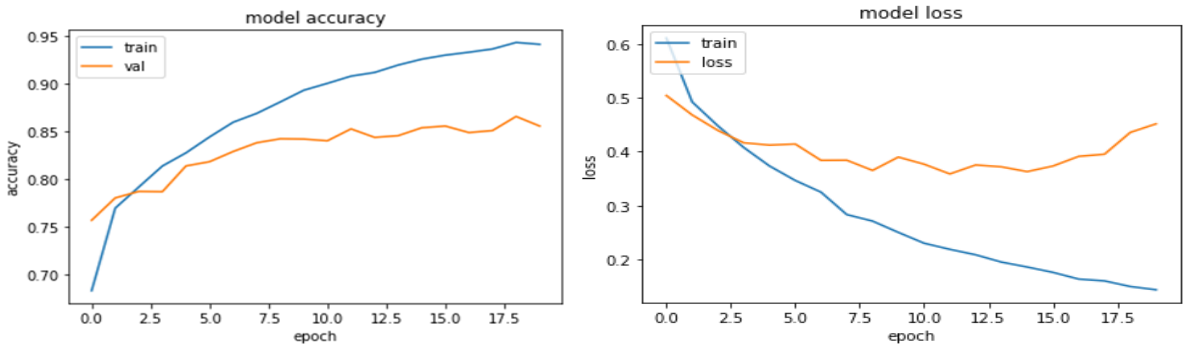

Fig 3. Accuracy loss graph

After 20 epochs the twitter dataset makes the predictions with bullying as 1 and non-bullying as 0. The accuracy-score, f-score, precision-score, bullying in category accuracy, bully count, non-bullying in category accuracy and non-bully count of Twitter dataset is then computed. The Twitter dataset result is shown in the Table 2.

Table 2. Twitter output

Accuracy-score	0.86474
F-Score	0.86477
Precision-Score	0.86501
Bullying in-category accuracy	0.871141
Bully count	2041
Non-Bullying in-category accuracy	0.44431
Non-Bully count	2188

The results show the bullying count as 2041 and the non-bullying count 2188 in the dataset. The classification report of Twitter dataset with necessary features of bullying and non-bullying comments and the overall accuracy of twitter along with the macro average and weighted average is shown in Table 3.

Table 3. Classification Report

	Precision	Recall	F1-score	Support
0	0.88	0.86	0.87	2188
1	0.85	0.87	0.86	2041
accuracy			0.86	4229
macro avg	0.86	0.86	0.86	4229
weighted avg	0.87	0.86	0.86	4229

Figure4 shows the ROC curve obtained.

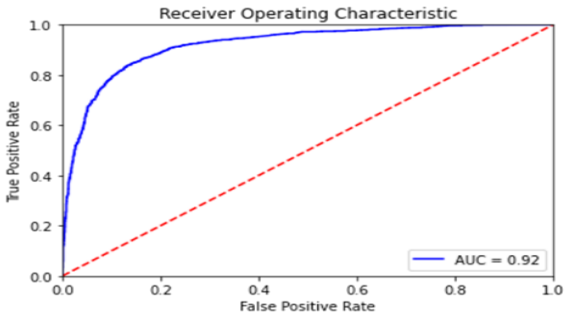

Fig 4. ROC curve

The comparison of several cyberbullying detection algorithms utilizing the Twitter dataset is shown in Table 4.

Table 4. Comparison with related works

Dataset	Models used	Precision	Recall	F1-Score
Twitter (16 k)	Bi-LSTM+ Attention [15]	0.698	0.686	0.692
	LSTM [16]	0.807	0.809	0.808
	Bi GRU [17]	0.828	0.831	0.829
	Bi GRU+ self-attention [17]	0.849	0.848	0.849
	LSTM+ GLoVe (Proposed work)	0.865	0.864	0.864

5. **Conclusion**

An LSTM RNN Deep Learning model has been built to detect cyberbullying on social media using the Twitter dataset. The word embeddings are produced with the pre-trained GloVe after the text datasets have been pre-processed, and the embedding matrix is obtained. The

dataset is divided into a training dataset and a test dataset, with an 80:20 split. In cyberbullying analysis, the LSTM model with the pretrained GloVe embedding performed well. There are extremely few posts tagged as bullying in the dataset used to detect cyberbullying. After the experimental analysis the LSTM worked fairly well on the Twitter dataset with the help of pre-trained model GloVe. Various performance measures were evaluated for twitter dataset. The LSTM model for cyberbullying analysis on social media using Twitter dataset is compared with some existing models and achieves better results. As a result, the Deep Learning-based LSTM model can distinguish between bullying and non-bullying material in the textual dataset.

References

[1] J.W. Patchin, S. Hinduja "Tween Cyberbullying in 2020", Cyberbullying Research Center and Cartoon Network, 2020

[2] Smit, D. M., "Cyberbullying in South African and American schools: A legal comparative study", South African Journal of Education, 35(2), 1076-1076, 2015.

[3] Reynolds, K., Kontostathis, A, Edwards, "Using machine learning to detect cyberbullying", In ICMLA, 241–244,2011.

[4] Gamback, B and Sikdar, U. K,"Using convolutional neural networks to classify hate-speech", Proceedings of the first workshop on abusive language online, 85-90, 2017

[5] Mikolov, T., Chen, K., Corrado, G and Dean, "Efficient estimation of word representations in vector space", arXiv preprint arXiv,1301-3781,2013.

[6] Georgakopoulos, Spiros "Convolutional neural networks for toxic comment classification", Proceedings of the 10th hellenic conference on artificial intelligence,2018.

[7] Oriola, O. and Kotze, E,"Evaluating machine learning techniques for detecting offensive and hate speech in South African tweets", IEEE Access 8, 21496-21509,2020.

[8] Rosa, Hugo, "Automatic cyberbullying detection A systematic review", Computers in Human Behavior 93, 333-345,2019.

[9] Akhter, M.P, Jiangbin, Z, Naqvi, I.R, Abdelmajeed, M. and Sadiq, M.T, "Automatic detection of offensive language for urdu and roman urdu", IEEE Access 8, 91213-91226,2020.

[10] Alhawarat, Mohammad, "A Superior Arabic Text Categorization Deep Model (SATCDM)", IEEE Access 8, 24653- 24661,2020.

[11] Aizawa, A.,"An information-theoretic perspective of tf–idf measures", Information Processing &Managemen, 39(1), 45-65,2003.

[12] Waseem, Zeerak, and Dirk Hovy. "Hateful symbols or hateful people? predictive features for hate speech detection on twitter." In Proceedings of the NAACL student research workshop, pp. 88-93. 2016.

[13] Bhagya J and Deepthi P S "Cyberbullying detection on social media using SVM", Inventive systems and control Springer Singapore, 17-27,2021.

[14] Banerjee, V, Telavane, J, Gaikwad, P and Vartak, "Detection of cyberbullying using deep neural network" In 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS) ,604-607,2019.

[15] Agrawal, Sweta and Amit Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms", In European conference on information retrieval Springer Cham 141-153,2018.

[16] Badjatiya, Pinkesh, Shashank Gupta, Manish Gupta, and Vasudeva Varma , "Deep learning for hate speech detection in tweets", In Proceedings of the 26th international conference on World Wide Web companion, 759-760,2017.

[17] Fang, Y, Yang, S, Zhao, B, and Huang, "Cyberbullying detection in social networks using Bi-gru with self-attention mechanism", Information, 12(4), 171,2021.

[18] Wu, Jiale, et al. "Toward efficient and effective bullying detection in online social network." Peer-to-Peer networking and Applications 13.5: 1567-1576,2020.

[19] Hochreiter, S, and Schmidhuber, "Long short-term memory, Neural computation", 9(8), 1735-1780, 1997.

[20] Dadvar, Maral and Kai Eckert, "Cyberbullying detection in social networks using deep learning-based models a reproducibility study", arXiv preprint arXiv, 1812.08046 , 2018 .

[21] Pennington, J, Socher, R, Manning , "Glove: global vectors for word representation", In EMNLP, 1532–1543,2014