

Sentiment Analysis and Deep Learning Based Cyber Bullying Detection in Twitter Dataset

Sherly T.T, B. Rosiline Jeetha



Abstract: When somebody, usually a teenager, abuses or harasses individual on the internet and other digital places, mainly on social networking platforms, this is termed as cyberbullying. Cyberbullying, like all types of bullying, produces psychological, emotional, and physical distress. Every individual's reaction to being bullied is diverse, but research has discovered certain common patterns. In a recent study, we introduced a technique called Hybrid Firefly Artificial Neural Networks (HFANN) to combat cyberbullying. Nevertheless, without considering the sentiment analysis features, accuracy of cyber bullying identification is lowered in this study. The Sentiment Analysis and Deep Learning based Cyber Bullying Detection (SADL-CDD) approach is used in the suggested research approach to address this issue. The punctuations, urls, html tags, and emoticons from the input tweet comments are removed first in this study project. Sentiment feature extraction is performed after pre-processing to improve classification accuracy. The Modified Fruit Fly Algorithm (MFFA) is used to choose the best features from the extracted features. Following feature selection, cyber bullying detection is carried out using a Hybrid Recurrent Residual Convolutional Neural Network (HRecRCNN). The experimental outcome of this study indicates the efficiency of the suggested approach. In comparison to current algorithms, the SADL-CDD method delivers improved classification performance with respect to reduced time complexity, greater precision, recall, f-measure, and accuracy.

Keywords: Cyber bullying, Artificial neural network, Deep learning, fruit fly algorithm, sentiment analysis, feature selection

I. INTRODUCTION

To this day, individuals all around the world use the internet as a means of communication [1]. Online tools, like SNSs, are highly famous socializing tool, particularly among adolescents, who have SNSs closely incorporated into their everyday routines since they may be a medium for users to communicate with one another without regard to time or location [2]. SNSs, on the other hand, may have bad implications if users abuse them, and one of the most prevalent negative behaviours on SNSs is cyber bullying, which is the subject of this study [3]. A person who engages in cyber bullying engages in threatening behaviour, harassment, etc toward another individual.

A group(s) or an individual(s) of humans who use telecommunication to frighten other people over communication networks is defined as cyber bullying [4]. Nevertheless, the majority of cyber bullying studies consider cyber bullying definition. "Willful and repeated harm done through the medium of electronic text" [5] is the cyberbullying definition. Flaming, harassment, denigration, impersonation, outing, boycott, internet stalking are all examples of cyber bullying [6]. Flaming is the most extreme kind of cyber bullying, whereas cyber stalking is the least severe. Flaming is an argument among two or more people about an occurrence that used harsh, insulting, or obscene language that happened over the internet [7]. Flaming is the most serious kind of cyber bullying as if an online battle between internet users occurs, it may be impossible to distinguish between cyber bully and victim at the moment. Harassment is defined as the transmission of damaging messages to a victim on a regular basis [8]. Denigration is when someone makes false statements, spreads falsehoods, or is harsh to a victim. Impersonation occurs when a cyber bully impersonates a target and posts negative information about that target with the purpose of bullying victim. When a cyber bully shares a victim's secrets or private information, it is known as outing. Boycotting is the deliberate exclusion of a person from social engagement on social media [9]. Cyber stalking, according to Willard, happens when a cyber abuser sends harmful communications frequently [10]. Because a cyber bully (cyber stalker) may be identified directly once they send annoying communications to a victim, cyber stalking is less severe than other types [11]. Many studies have demonstrated that ML algorithms can be used to predict and identify cyberbullying behaviour [12]. Because enormous amounts of data are created every second, methods may be trained efficiently. ML classification methods assist in the classification of text material into non-bullying and bullying categories. Bullying may be halted after the content has been classified. Deep learning techniques may also be used to enhance speed and decrease the amount of time spent on explicit feature extraction [13]. In this study, a sentiment analysis and deep learning-based approach for detecting cyberbullying is presented. Primarily, sentiment features will be collected from Twitter comments, and then an enhanced fruit fly search algorithm will be used to choose the best features. Lastly, the Hybrid Recurrent Residual Convolutional Neural Network (HRecRCNN) is used for classification, resulting in detection of cyber bullying. The following is the overall structure of the research project: This section provides a thorough introduction to cyberbullying. In section 2, several study techniques for predicting cyber bullying are addressed in depth. In section 3, the suggested research technique is discussed in depth, along with an appropriate architecture.

Manuscript received on September 27, 2021.

Revised Manuscript received on October 27, 2021.

Manuscript published on November 30, 2021.

* Correspondence Author

Sherly T.T*, Research Scholar, PG and Research Department of Computer Science, Dr. N.G.P. Arts and Science College, Coimbatore, Tamil Nadu, India

B. Rosiline Jeetha, Research Guide, PG and Research Department of Computer Science, Dr. N.G.P. Arts and Science College, Coimbatore, Tamil Nadu, India

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Section 4 provides an overall overview of the study effort based on simulation results. Lastly, in section 5, the general conclusion of the study project is presented based on the findings.

II. RELATED WORKS

Balakrishnan et al. [14] investigated an automated cyber bullying detection system based on psychological features of Twitter users like personalities, sentiment, emotion. Big Five and Dark Triad models were employed to identify user personalities, while ML algorithms such as NB, RF, J48 were utilized to categorize tweets into one of 4 groups: bully, aggressor, spammer, and normal. Twitter database contains 5453 tweets that were carefully annotated by human specialists using the hashtag #Gamergate. Cheng et al [15] developed a hierarchical attention network for detecting cyberbullying that considers these features of cyberbullying. T method has three main distinguishing features: 1) a hierarchical structure that mimics social media session's structure; (2) levels of attention mechanisms applied at the word and comment level, allowing the model to pay varied amounts of attention to words and comments based on context; 3) a cyber bullying detection job that also predicts the occurrence of cyber bullying. Visual features complement textual features in cyber bullying identification, according to Singh et al [16], and can assist enhance prediction outcomes. As a result, there is a great need to develop automated ways for detecting and preventing cyberbullying. Despite recent cyber bullying detection attempts have established advanced text processing algorithms for cyber bullying detection, visual data processing has yet to be used to automatically identify cyber bullying. Van Hee et al. [17] presented the gathering and fine-grained annotation of a cyberbullying corpus in English and Dutch, as well as a set of binary classification tests to see if automatic cyberbullying detection is possible. They utilised linear SVMs to analyse which information sources contribute the most to the job using a large feature set. Tests on a hold-out testset show that detecting cyberbullying-related posts may be done successfully.

In English and a few other languages, Haidar et al [18] presented strategies for detecting cyberbullying, but none had yet addressed cyberbullying in Arabic. ML and NLP are two techniques that help detect cyberbullying. This article builds on a prior work by describing a method for identifying and preventing cyberbullying. It begins with a detailed review of past work in the field of cyberbullying detection. After that, a method for identifying cyberbullying in Arabic content is shown and evaluated. Using DL-based models and transfer learning, Agrawal et al [19] thoroughly evaluated cyber bullying detection on diverse themes across many SMPs. The researchers used three real-world databases: Formspring (12k posts), Twitter (16k posts), Wikipedia (100k posts) to conduct comprehensive tests. Our research yielded a number of valuable insights on the detection of cyberbullying. Gencoglu et al. [20] presented a model training technique that may incorporate fairness restrictions and test our method using various databases. The authors show how to successfully reduce various forms of

unintentional biases without compromising model quality. The researchers think that their study advances the search for unbiased, transparent, ethical ML solutions for cyber-social health. The Big Five and Dark Triad systems were used by Balakrishnan et al [21] to create a cyberbullying detection algorithm formed on user psychology. Relying on connections between personality traits and cyberbullying, the model tries to detect bullying trends within Twitter groups. For cyberbullying categorization (i.e. aggressor, spammer, bully, normal), RF, a famous ML approach, was employed in combination with a baseline method comprising 7 Twitter features (number of mentions, followers and following, popularity, favourite count, status count, number of hash tags). Iwendi et al [22] conducted a study to assess the efficiency and performance of DL models in identifying insults in Social Commentary. BLSTM, GRU, LSTM, and RNN were the four DL algorithms employed in the experiments. Text cleaning, tokenization, stemming, Lemmatization, and the removal of stop words were among the data pre-processing procedures used. Clean textual data is supplied to deep learning models for prediction after data pre-processing. When contrasted with RNN, LSTM, and GRU models, the BLSTM model obtained good accuracy and F1-measure values. CONcISE, a new method for timely and accurate Cyberbullying detection on Instagram media SEssions, was developed by Yao et al [23]. They presented a sequential hypothesis testing framework that seeks to minimize the amount of features required in identifying every remark while yet retaining good classification accuracy. Only when a specific number of detections has been made, the CONcISE issue an alert. Raisi et al. [24] gave a limited seed vocabulary of bullying indicators, and the system extracts bullying roles of users and additional word indications of bullying from a vast, unlabeled corpus of social media interactions. The system aims to optimize the agreement among these estimates, i.e. participant-vocabulary consistency, by estimating whether every social encounter is bullying depending on who participates and what language is utilized. They tested PVC on three social media databases, proving its efficiency in detecting cyberbullying both quantitatively and qualitatively.

III. SENTIMENT ANALYSIS BASED CYBER BULLYING DETECTION

Harassment and cyberbullying are two examples of harmful internet conduct that are causing damage on people's lives. As a result of this phenomena, automated, data-driven approaches for assessing and detecting such actions are required. Cyber bullying detection is accomplished in this study by modifying the sentiment feature extraction and analysis technique. This is accomplished by extracting sentiment features and utilizing an enhanced fruit fly method to choose the most optimal features. Lastly, the Hybrid Recurrent Residual Convolutional Neural Network (HRecRCNN) is used to accurately predict cyberbullying. The operational flow for the suggested research study is depicted in Figure 1.



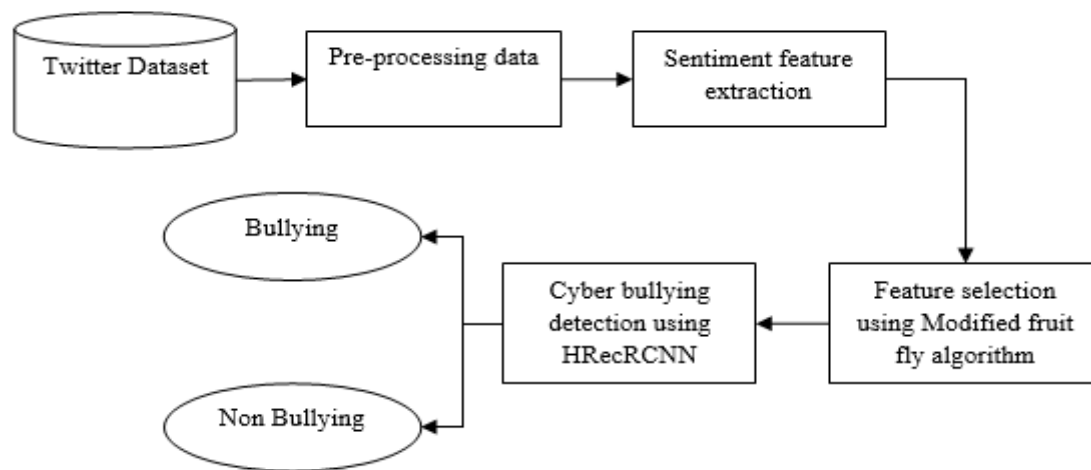


Figure 1. Overall operational flow of suggested research study

3.1. Twitter Dataset Pre-processing

Twitter is a micro-text-based social media platform. URLs, hash tags, emoticons, misspelling words, Internet slang, and improper grammatical expressions are common in tweets. The text was processed using Python's built-in regular expressions. All tweets and retweets (tweets shared or republished by another user) that were not in English were deleted. Furthermore, tweets with the hash tag "ebook" were deleted. Because certain spam or marketing accounts frequently link to clinical reports, the hash tag "ebook" was generally used. Aside from that, to denoise the tweets, the following steps were done:

- Remove duplicate tweets.
- A space was used to replace usernames and hash tags.
- Remove extra newline (\n), punctuations (excluding for "!" and "?").

Then came tokenization, stemming, and part-of-speech extraction. Tokenization is the process of breaking down a phrase into words. The extraction of lexical elements with comparable grammatical properties is known as POS. The act of minimizing inflected words to their root forms is known as stemming. For instance, "eating," "ate," "eaten," and "eats" can all be stemmed to "eat." These tasks were completed using Text Blob, a Python library for analyzing textual data.

3.2. Sentiment Feature Extraction

Here, feature extraction is carried out utilizing a variety of feature extractors to extract various types of features from input twitter content. Below is a list and explanation of some feature extractors:

TF-IDF:

Term Frequency and Inverse Document Frequency are two statistical techniques employed by TF-IDF. Inverse document frequency measure of how much information a word offers is the total number of times a particular term t occurs in the document doc versus (per) the total number of entire words in the document. It determines the importance of a particular word across the document. IDF displays the frequency of a specific term across all texts. TF-IDF may be calculated using the formula $tf * idf$. The product of TF and IDF gives us TF-IDF metric, that measures how frequently a

term is present in a document multiplied by how unique it is. In document j , for a word i :

$$W_{ij} = t_{fij} \times \log(N / d_{fi})$$

here

t_{fij} = total occurrences of i in j

d_{fi} = total documents having i

N = Total documents

Bag of words:

The bag of words method is the easiest way to transform text into structured features. The term "bag of words" simply divides the words in the review text into separate word counts. Machine learning methods are used extensively in the field of text analysis. Nevertheless, because most methods demand numerical feature vectors of a fixed size instead of raw text documents of variable length, the raw data, a series of symbols, cannot be supplied directly to them. To help with this, scikit-learn includes tools for the most common methods of extracting numerical features from text, such as:

- tokenizing strings and providing a numeric id to every potential token, such as by utilizing whitespace and punctuation as token separators.
- Counting the number of times tokens appear in every document.
- normalizing and weighting tokens that appear in the majority of samples / documents with decreasing significance tokens.

The following are the definitions of features and samples in this system:

- Every token's frequency of occurrence (normalized or not) is considered as a feature.
- A multivariate sample is a vector that contains all the token frequencies for a particular text.

A matrix having one row each document and one column per token (e.g. word) occurring in the corpus may therefore be used to depict a corpus of documents. The procedure of converting a group of text documents to numerical feature vectors is referred to as vectorization. Bag of Words or "Bag of n-grams" representation refers to this specific approach (tokenization, counting, and normalization).

Word occurrences are used to describe documents, but the relative location information of the words in the document is totally ignored.

N-gram:

In the fields of computational linguistics and probability, an n-gram is a continuous sequence of n components from a particular sample of text or speech. The components could be phonemes, syllables, letters, words, or base pairs, depending on the application. n-grams are typically retrieved from a text or speech corpus. There are two sorts of output from Extract N-Gram Features from Text module: Dataset with the results: A overview of the text that was analyzed, along with the n-grams that were extracted. The result includes columns that users did not choose the Text column option. The main idea behind n-grams is that they capture the structure of language from a statistical perspective, such as which letter or word is most likely to follow the current one. The more background one have to work with, the longer the n-gram (the greater the n). A character n-gram is a string of n characters in a row. The number of n-grams that may be created for a given document (typically 1, 2, 3, or 4) is the outcome of moving a window of n characters along the text. One character at a time is moved in the window. The number of times every n-gram appears is then counted.

Using n-grams has a number of advantages. One of them is that humans don't need to do word segmentation when they use n-grams. Furthermore, there is no requirement for a dictionary or any other language-specific approaches. On the other hand, extracting n-grams from a huge corpus will result in a high number of potential n-grams. Only a few of them will have high discriminate power and discriminating frequency values in vectors representing the texts.

Slang words:

The creation of a slang word dictionary will be done manually at first in this project. This built dictionary will be used as a learning tool. The data will be tokenized, and the tokenized terms will be compared to the slang dictionary. Slang words will be assigned to the terms that match. The subjects are chosen to serve as a foundation for examination. The similarities with other months' subjects are then computed, and the top three are output. The values are computed for every month such that the top three similarities' transitions may be examined on a monthly basis. The vector containing the feature words and their appearance probability is utilized to compute similarities. Nevertheless, because the tweets containing the target slang are being analysed, there is an issue in that the probability of slang appearing becomes quite high.

Emoticon lexicons:

Real-world emotion data is modelled as a combination of emotion-bearing and emotion-neutral (background) words. To explain the production of documents connoting emotion e_t , the generative model is:

$$P(D_{et}, Z | \theta_{et}) = \prod_{i=1}^{|D_{et}|} \prod_{w \in d_i} (1 - z_w) \lambda_{et} P(w | \theta_{et}) + z_w (1 - \lambda_{et}) P(w | N)^{c(w, d_i)}$$

here θ_{et} is emotion language model. N indicates background language model. λ_{et} indicates mixture parameter, Z_w is binary hidden variable that represents language model (θ_{et} or N) which generates word w. Additionally, $c(w, d_i)$ indicates how many times word w appears in document d_i . Estimation of parameters θ_{et} and Z is performed using expectation maximization (EM), that repeatedly increases complete data (D_{et}, Z) by alternating among E-step and M-step. E and M steps in this case are:

E-Step:

$$P(Z_w = 0 | D_{et}, \theta_{et}^{(n)}) = \frac{\lambda_{et} P(w | \theta_{et}^{(n)})}{\lambda_{et} P(w | \theta_{et}^{(n)}) + (1 - \lambda_{et}) P(w | N)}$$

M-Step:

$$P(\theta_{et}^{(n+1)}) = \frac{\sum_{i=1}^{|D_{et}|} P(Z_w = 0 | D_{et}, \theta_{et}^{(n)}) c(w, d_i)}{\sum_{w \in V} \sum_{i=1}^{|D_{et}|} P(Z_w = 0 | D_{et}, \theta_{et}^{(n)}) c(w, d_i)}$$

here n represents EM iteration value. EM is employed to estimate k mixture models' parameters related to emotions in E. Emotion lexicon UMMlex is learnt by employing k emotion language systems and background model N as:

$$UMMlex(w_i, \theta_{et}) = \frac{P(w_i | \theta_{et}^{(n)})}{\sum_{i=1}^k [P(w_i | \theta_{et}^{(n)})] + P(w_i | N)}$$

$$UMMlex(w_i, N) = \frac{P(w_i | N)}{\sum_{i=1}^k [P(w_i | \theta_{et}^{(n)})] + P(w_i | N)}$$

UMMlex is a $|V| \times (k+1)$ matrix, and k is number of emotions in the corpus. Table 2 represents an example of UMM terminology. On social media, one would notice the non-standard and creative expressions like :), nice!! that are frequently utilized to describe emotions. Such expressions frequently enhance text's emotional impact. For social media emotion analysis, modelling such expressions is essential. As a result, emoticons (such as: :) and concatenated expressions (such as: good!!) are tokenized as single words during the text pre-processing step to capture their connection with various emotions.

3.3. Optimal Feature Selection using Modified Firefly Algorithm

FOA is a novel heuristic approach that seeks the best solution to an objective function by simulating the foraging behaviours of fruit flies in nature. Because the fruit fly's olfactory and visual systems are still developing, it uses its sense of smell to conduct a wide variety of searches before sending food odor information to nearby drosophila when foraging. Individual flies will depend on visual function to move to a site where a fly has a higher concentration by comparison. Since the population's variety is diminished as a result of such features, the method that mimics them, like other bionic intelligent systems, has the flaw of being easy to slip into local optimization, that leads to issue of early maturity. Somatic cell theory and network theory were used to develop the immune system. It can also do self-regulation by producing antibodies that are comparable to those produced by the immune system. This technique specializes at local searches.

They can use this to enhance the basic fruit fly method by including it into the latter stages of execution. The novel method IFOA may be employed to compensate for shortcomings of the fruit fly method, which is disposed to falling into a local optimum state, and increase search efficiency. The immune system is a basic defensive system that prevents bacteria from invading living organisms and maintaining proper metabolism. To create biological variety, the system identifies gene kinds to generate various antibodies, encourages the emergence of new persons, and prevents the excessive production of individuals through regulatory processes. The feasible solution set was originally constructed using the FOA method, which is based on the conventional fruit fly method. The trigger criterion was then the number of evolutionary stasis steps t . IA search process was activated when $t > T$ (threshold of evolutionary stasis steps), and the process of immune factors in the immune method (correlating to individual drosophila) seeking antigens (correlating to food source in the drosophila method) producing antibodies (optimal solution) was utilized to expand the search space. In the feasible solution set acquired during IA search, the elite retention approach was implemented. And the resulting optimization solution was combined with initial feasible solution set to update the concentration of odour information in the search space, guiding the path search mechanism of additional fruit flies. The index to enter IA method is the threshold value of evolutionary stagnation step T . early implementation of IA method limits search capabilities of the method, as well as the convergence of suitable solutions. $T = 6$ was chosen as the trigger value for entering the IA method after several independent trials. If the trigger condition is met, the space optimization was done using a fixed immune factor redistribution probability. And, depending on their fitness levels, various people were assigned varying adaptive immune probabilities. The MFA procedure is as follows:

Algorithm: Modified Fruit Fly method

Step 1: set parameters. initialize Sizepop and Maxgen of population size, set population positions X_axis and Y_axis , and the number of evolutionary stagnation steps $t=0$;

Step 2: arbitrarily produce fruit fly population as stated by following formula;

$$Dist_i = \sqrt{X_i^2 + Y_i^2}$$

Step 3: Operate the population through following formula;

$$S_i = \frac{1}{Dist_i}$$

$$Smell_i = Fitness(S_i)$$

$$[bestSmell, bestIndex] = \min(smell)$$

$$SmellBest = bestSmell$$

Step 4: Record and maintain the best favor concentration value as said by following formula, and update the evolutionary iteration step number t ;

$$\begin{cases} \text{If } (bestSmell < Smellbest) \\ \quad Smellbest = bestSmell1 \\ \quad \text{else } (t = t + 1) \\ \quad Smellbest = bestSmell2 \\ \quad \text{end} \end{cases}$$

Step 5: check whether $t < T$ is true, if so, directly go to step 7; else as per following equation, the adaptive immunity probability of individual fruit fly was computed, and the immune operation was performed as per immune method. For individuals who did not do immune process, step 7 is considered.

$$P(i) = \frac{(Smellbest - Smell(i))}{((Smellbest - Smellworst)P^*)}$$

Step 6: Repeat steps 2-4 for iterative search of the new population attained by immunization

Step 7: $gen = gen + 1$; if $gen < Maxgen$, go to step 2. else end the iteration

3.4. Cyber Bullying Detection using HRecRCNN

Deep Learning (DL) approaches have proved to be extremely successful when adequate labelled data is available. Different improved DL approaches have been suggested in numerous medical imaging and computer vision modalities in the recent few years, which demonstrate higher performance than modern methods (Ball et al., 2017). Hybrid Recurrent Residual Convolutional Neural Networks (HRecRCNN) are a type of improved hybrid DCNN architecture that uses RCNN architecture, residual networks, and inception. When contrasted with alternative DL techniques like residual network, RCNN, and inception, this model provides superior recognition performance with same or fewer network parameters. It is a significant benefit of this method. The inception-residual units were used in this model, which was based on the Inception-v4 model. When HRecRCNN is compared to equivalent inception-residual networks, it outperforms them. The Inception V4 model is incorporated into the RecRCNN. As a result, the network was called HRecRCNN. classification performance is improved by utilizing Harmony Search Optimization to select the best parameter values. In the HRecRCNN model, there are stacks that incorporate both transition units and Inception Recurrent Residual Units (IRResU). The whole model is seen in Figure 2. At the output layer, there are different convolution layers, transition blocks, IRResUs, and softmax. The IRResU's visual representation is shown in Figure 3. The steps employed to train the system are illustrated in the top half of Figure 2, while the testing phase is represented in the lower section. The outcomes of this approach are evaluated using a variety of performance evaluation measures.

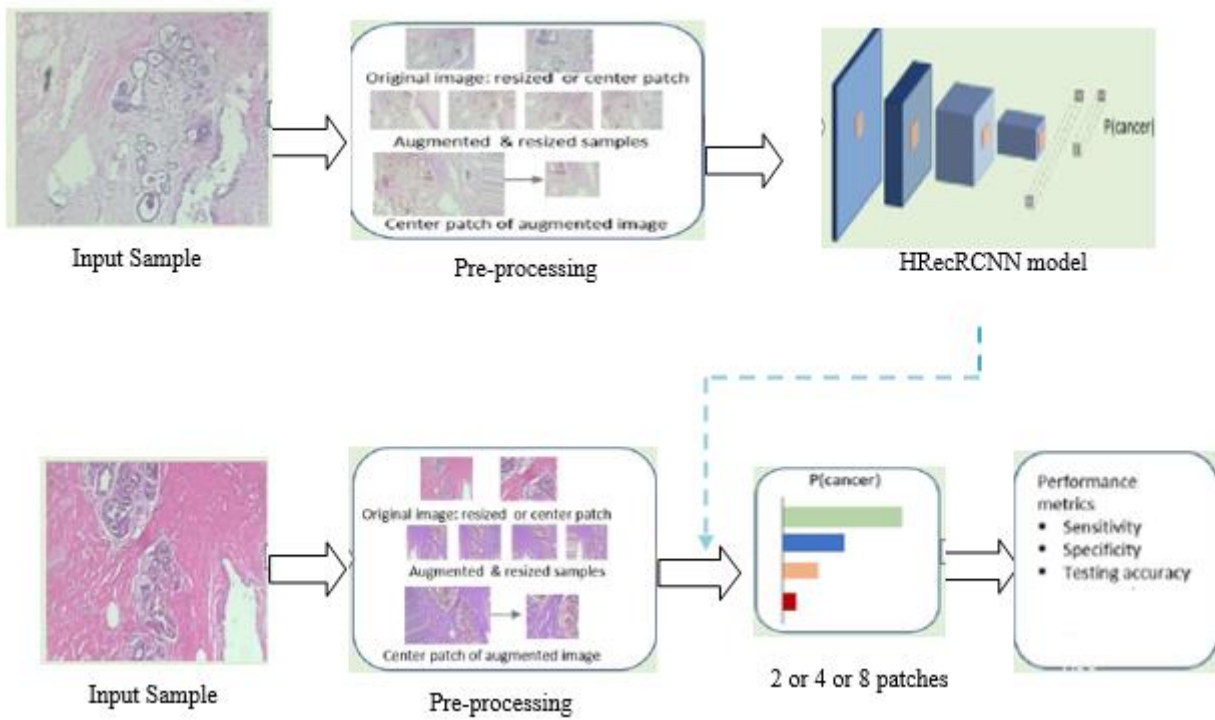


Figure 2: Breast Cancer Recognition Implementation Diagram utilizing HRecRCNN Model

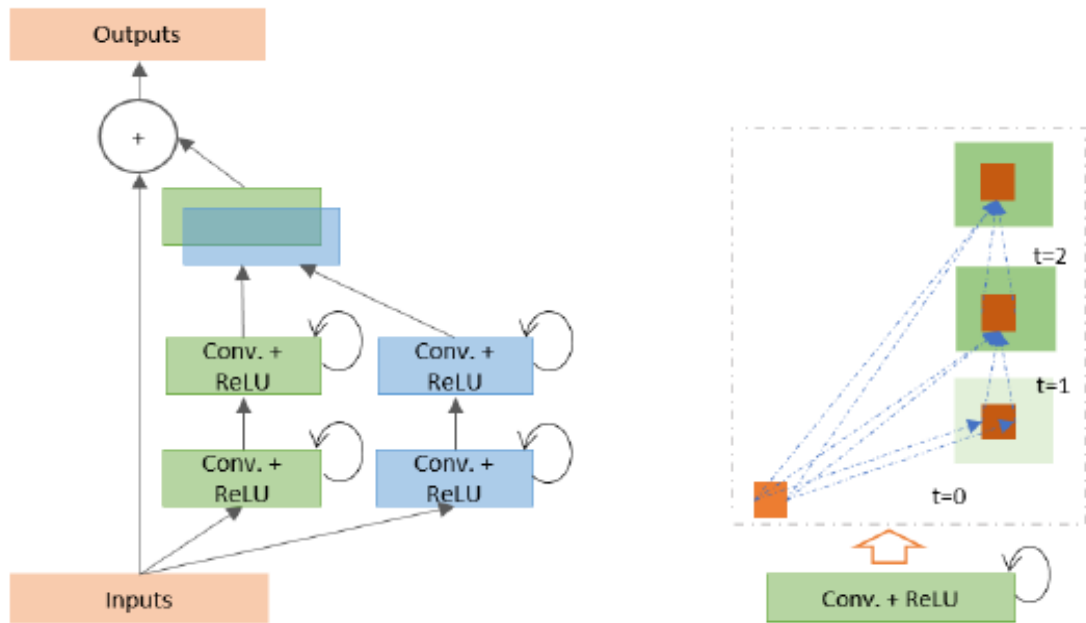


Figure 3: Displaying the Inception Recurrent Residual Unit

Figure 3 shows concatenated recurrent convolutional layers and inception units, as well as residual units. Before the output block, input features summation with inception unit output may be observed. The IRResU is a crucial component of the HRecRCNN framework, since it contains the residual layer, inception units, and Recurrent Convolutional Layers (RCLs). Inputs are supplied into input layers, which are then processed via inception units and RCLs are used. IRResU inputs are then combined with the outputs of the inception units. recurrent convolution operations were done in the inception unit using different kernel sizes. Because of recurrent structure inside the convolution layer, earlier time step outputs are combined

with present time step outputs. The current time step outputs are then utilized as inputs for the following time step. Similar processes are carried out based on assumed time steps. One feed forward convolution with two included RCLs, for example, is expressed as $t = 2 (0 \sim 2)$ in IRResU. Figure 3 depicts the functioning of the RCLs at varied time steps ($t = 2 (0 \sim 2)$) and ($t = 3 (0 \sim 3)$). IRResU is used to accumulate feature maps based on time steps.

This provides improved feature representation, and with similar number of network parameters, this approach may achieve greater performance.

The RCL procedures were conducted using discrete time steps, and they were represented using HRecRCNN. Consider that x_l input sample in HRecRCNN block's l^{th} layer comes from an input sample in the k^{th} feature map in RCL, and that unit (i, j) comes from an input sample in the k^{th} feature map. Furthermore, network output is expected to be $O_{ijk}^l(t)$ at time step t . (t). output is calculated using these data:

$$O_{ijk}^l = (w_k^f)^T * x_l^{f(i,j)}(t) + (w_k^r)^T * x_l^{r(i,j)}(t-1) + b_k$$

here the input for ordinary convolution layers is $x_l^{f(i,j)}(t)$, the weight value is w_k^f and input for l^{th} RCL is $x_l^{r(i,j)}(t-1)$, the weight value is w_k^r , and the bias is b_k . The HIRResCNN model's classification accuracy is improved by using HSO optimization methods to modify these weight and bias parameters.

$$y = f(O_{ijk}^l(t)) = \max(0, O_{ijk}^l(t))$$

Here f represents the activation function of a typical Rectified Linear Unit (ReLU). This model's performance is assessed using the Exponential Linear Unit (ELU) activation function: Inception units output y is denoted by $y_{1 \times 1}(x), y_{3 \times 3}(x)$, and average pooling layer output is expressed by $y_{1 \times 1}^p(x)$ for different sized kernels. The final result of the IRCNN unit is represented as $\mathcal{F}(x_l, w_l)$ and indicated as:

$$\mathcal{F}(x_l, w_l) = y_{1 \times 1}(x) \odot y(x) \odot y_{1 \times 1}^p(x)$$

Concatenation operation is depicted as \odot based on feature map axis or channel. The input of the HRecRCNN block is then combined with output of the IRCNN unit. The residual functioning of the HRecRCNN block is,

$$x_{l+1} = x_l + \mathcal{F}(x_l, w_l)$$

Where the input of the immediate next transition block is x_{l+1} , the HRecRCNN block's input samples are x_l, l^{th} . The kernel weights of the HRecRCNN block are denoted as w_l , while the output of the IRCNN unit's l^{th} layer is expressed as $\mathcal{F}(x_l, w_l)$. The HRecRCNN model's classification accuracy is improved by using the HSO optimization approach to tune the w_l and *bias* parameters.

The number of highlight maps and the components of the element maps for remaining units, however, are similar as HRecRCNN unit as indicated in Figure 5. The IRResU yields are subjected to cluster standardization. Finally, the yields of this IRResU are allocated to the contributions of the following progress unit as soon as possible (Liang et al., 2015). Based on state of progress unit in the model, various operations such as convolution, pooling, dropout are done in change unit. progress unit remembers initiation units.

Down-inspecting operations are carried out in progress units, where they do max-pooling jobs with a 33% fix and a 22% step. Because non-covering max-pooling has a negative impact on model regularization, they utilized covered max-pooling to regularize the system, which is important when planning a deep system engineering. late

use of a pooling layer aids in increasing the non-linearity of system's highlights by allowing higher-dimensional component maps to pass through system's convolution layers. This also makes it easier to limit the number of system boundaries to a minimum. advantage of using a 1×1 channel is that it aids to improve non-linearity of choice capacity without affecting the convolution layer. information size and yield highlights in the HRecRCNN units does not vary. Finally, at the end of the engineering, utilized a softmax, or standardized exponential capacity layer.

Softmax Classifier: Softmax operation for i^{th} class, with K distinct linear functions, weight vector W and input sample x is stated as

$$P(y = i|x) = \frac{e^{x^T w_i}}{\sum_{k=1}^K e^{x^T w_k}}$$

Several convolutional layers counts are employed in convolution blocks to evaluate this IRResCNN model, and layers counts are calculated as per time step t . In the start, a model with two convolutional layers is employed in breast cancer recognition, and it contains four IRCCN blocks, transition block, fully connected layer, hidden layer, and softmax layer at the conclusion.

This last hidden state is used to classify data. The formula for classification is shown below,

$$y = \text{softmax}(w_k^f h_T + b_k)$$

$$y = \text{softmax}(w_k^r h_T + b_k)$$

Here y represents the anticipated gait type, w_k^f represents the output weight, w_k^r represents the input weight, and b_k represents the output bias. Softmax classifier is used to detect image patterns. As a result of the classifier, a trained feature vector is generated. Every patch in the breast cancer image x_i is transformed into a fixed-length learned feature vector \hat{x} . This hidden layer is sent into the softmax classifier to provide classification performance. Softmax classifiers are a type of multiclass classifier that uses logistic regression for data classification. This is used to calculate every class probability for which the data is classified. As a result, the probability sum will equal one. The softmax function is used to normalize the data, and the exponentiation procedure is used to calculate the class probabilities. Fine tuning is the following training stage after training all network layers. The final stage of the classification process is fine tuning, that is used to improve model performance.

IV. RESULTS AND DISCUSSION

In the Twitter database, 4556 tweets on a variety of topics, including demonetisation, kids, mobile phones, Sachin, and Whatsapp, are searched and rated as positive and negative opinion tweets. The tweets that were analysed are included in Table 1. Precision, accuracy, recall, f-measure, and time complexity are just a few of the performance metrics investigated.



Sentiment Analysis and Deep Learning Based Cyber Bullying Detection in Twitter Dataset

The existing SSVM, v, new ECSO+HFANN, and HRecRCNN techniques are evaluated for implementing the above specified measures on Twitter database.

Table 1 - Tweets Collected through varied w search terms

Name of the File	Number of Tweets	Actual	
		Positive Opinion Tweets	Negative Opinion Tweets
demonetisation.txt	1003	498	505
kids.txt	984	402	582
mobilephones.txt	783	599	184
sachin.txt	994	483	511
whatsapp.txt	792	599	193

- False Positive (FP) → tweets with positive opinion were inaccurately labelled.
- True Negative (TN) → tweets with negative opinion were accurately recognized.
- False Negative (FN) → Negative opinion tweets were inaccurately detected.

Precision

Precision is computed like:

$$\text{Precision} = \frac{\text{True-positive}}{\text{True-positive} + \text{False-positive}}$$

Recall is a computation of correctness or quality, while precision is a measurement of completeness. generally, high precision indicates that an approach produced a huge number of relevant outcomes compared to irrelevant ones. number of true positives divided by total items recognized as pertaining to the positive class in a classification task is the accuracy of a class.

- True Positive (TP) → tweets with positive opinion were accurately detected.

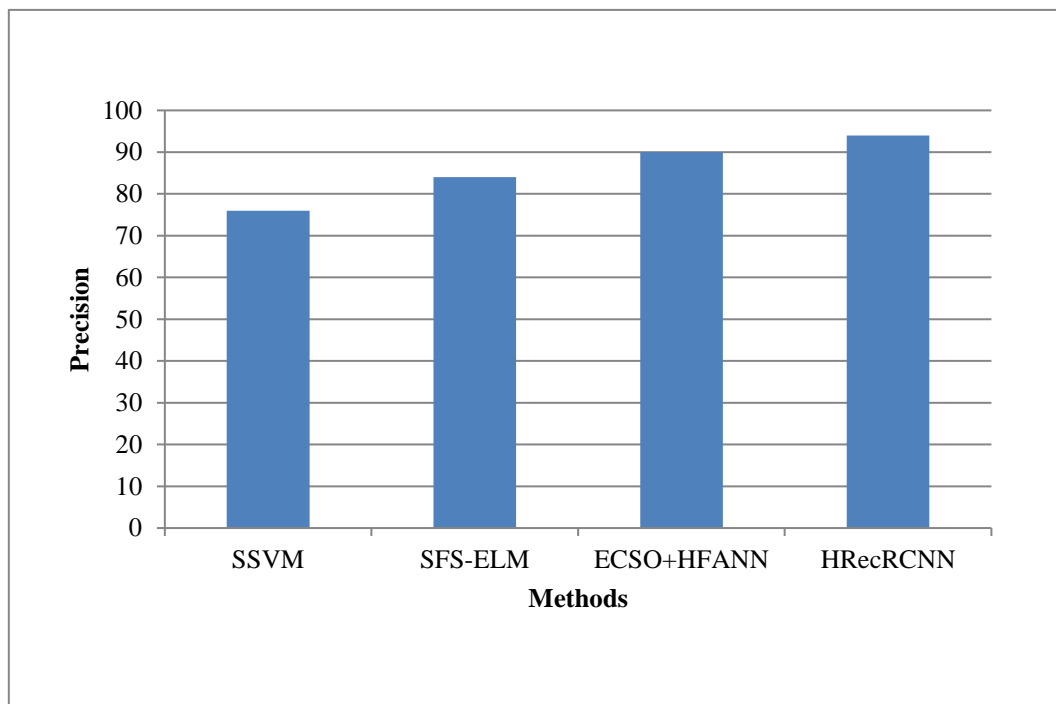


Fig 4 Precision

The comparative measure is evaluated for accuracy using existing and recommended techniques, as seen in Figure 4. x-axis displays the methods, while y-axis indicates accuracy values. Existing techniques such as SSVM, SFS-ELM, and ECSO+HFANN have lower precision than the proposed HRecRCNN method for the supplied Twitter database. As per the findings, the proposed HRecRCNN improves Cyberbullying detection accuracy by selecting the optimal features.

Recall

Recall is computed as:

$$\text{Recall} = \frac{\text{True-positive}}{\text{True-positive} + \text{False-negative}}$$

following is a representation of comparison graph:

The recall of a search is described as total relevant documents recovered divided by number of existing relevant documents, while precision is calculated as total relevant documents retrieved divided by actual documents retrieved by the search.

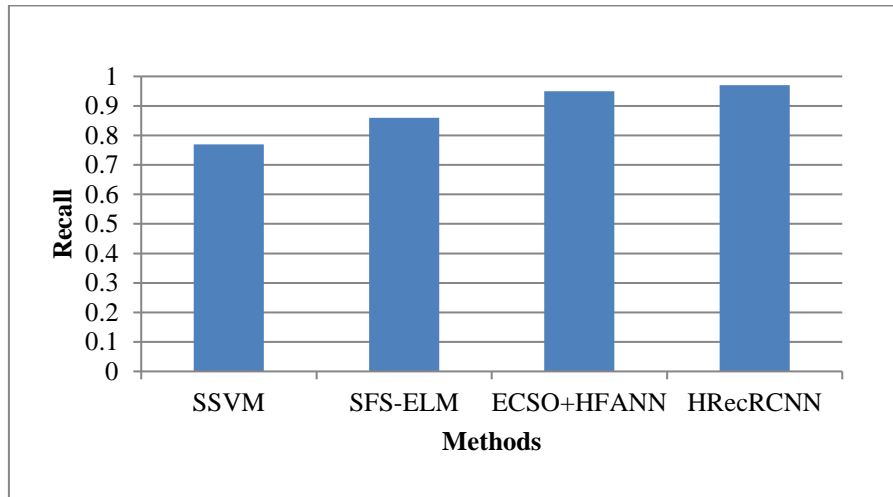


Fig 5 Recall

The comparative measure is evaluated with respect to recall using both existing and new approaches, as illustrated in Fig 5. On the x-axis, the methods are plotted, whereas the recall value is represented on the y-axis. Current approaches like SSVM, SFS-ELM, and ECSO+HFANN methods have reduced recall for the supplied Twitter database but the recommended HRecRCNN method has a greater recall. As per the findings, the proposed HRecRCNN improves

Cyberbullying detection accuracy by selecting the optimal features.

F-measure

F1-score is indicated as:

$$F1 - score = \frac{2 \times precision \times recall}{precision + recall}$$

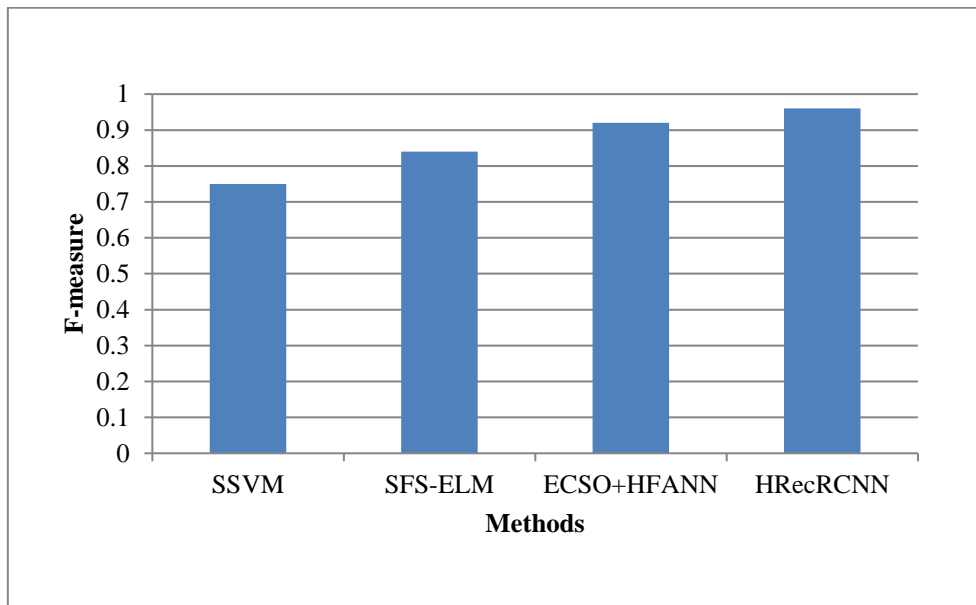


Fig 6. F-measure

The comparative metric is examined using both existing and recommended techniques with respect to F-measure, as illustrated in Fig 6. On the x-axis, the methods are presented, and F-measure value is presented on the y-axis. Current approaches like SSVM, SFS-ELM, and ECSO+HFANN methods provide lower F-measures for the supplied Twitter dataset, but the recommended HRecRCNN method produces higher F-measures. As per the findings, the proposed HRecRCNN improves Cyberbullying detection accuracy by selecting the optimal features.

Accuracy

It is calculated as overall correctness of the system and is calculated as total actual classification parameters ($T_p + T_n$) that is segregated by total classification parameters ($T_p + T_n + F_p + F_n$). This is calculated as:

$$Accuracy = \frac{T_p + T_n}{(T_p + T_n + F_p + F_n)}$$

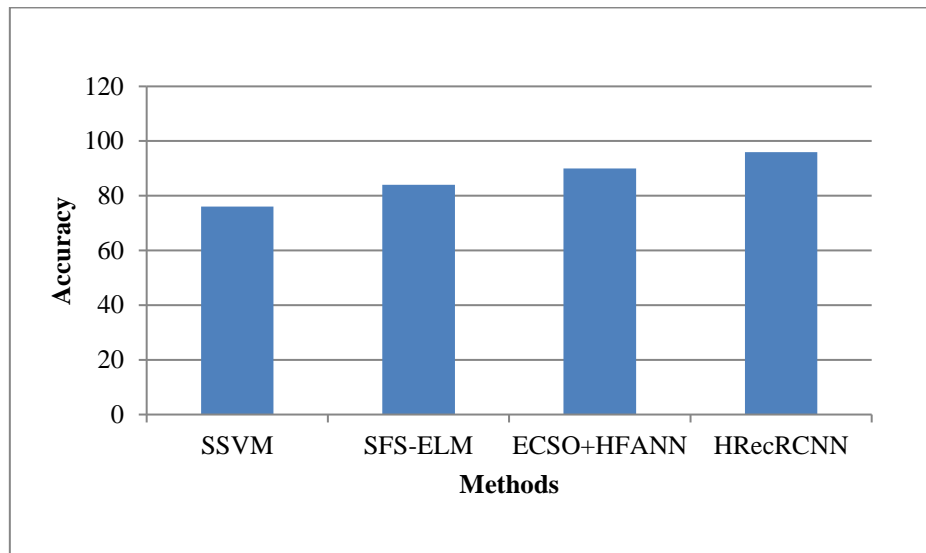


Fig 7. Accuracy

The comparative measure is evaluated for accuracy using both existing and recommended techniques, as seen in Fig 7. On x-axis, the methods are presented, and the accuracy value is displayed on y-axis. Existing techniques such as SSVM, SFS-ELM, and ECSO+HFANN have minimum accuracy than the proposed HRecRCNN method for the supplied Twitter database. As a result, as per the findings, the recommended HRecRCNN improves

Cyberbullying detection accuracy through the optimal selection of features.

Time complexity

Whenever the suggested method operates in less time, the system performs better.

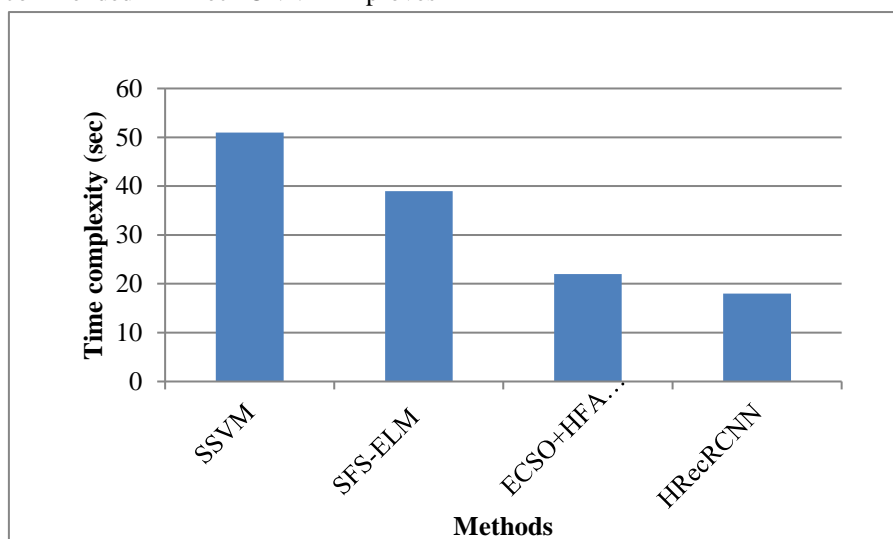


Fig 8 Time complexity comparison

The comparative measure is evaluated with respect to temporal complexity using both existing and novel approaches, as illustrated in Fig 8. On x-axis, the approaches are shown, whereas the time complexity value is represented on y-axis. Current methods such as SSVM, SFS-ELM, and ECSO+HFANN have a higher time complexity, however with the supplied Twitter database, the recommended HRecRCNN technique has a lower time complexity. As a consequence, the proposed HRecRCNN enhances Cyberbullying detection performance in the Twitter database by properly categorizing outcomes.

V. CONCLUSION

Cyberbullying detection was discovered to be the most researched topic in the social media environment. It is necessary to develop a cyberbullying detection mechanism in order to ensure safety and harassment protection. This is accomplished in this study by using the HRecRCNN technique. The punctuations, urls, html tags, and emoticons from input tweet comments are removed first in this study project.

Sentiment feature extraction is performed after pre-processing to improve classification accuracy. The Modified Fruit Fly Algorithm (MFFA) is used to choose the optimal features from the extracted features. Following feature selection, cyber bullying detection is carried out using a Hybrid Recurrent Residual Convolutional Neural Network (HRecRCNN). The test outcome of this study indicates the efficiency of the suggested approach. In comparison to current algorithms, the SADL-CDD method delivers improved classification performance with respect to reduced time complexity, greater precision, recall, f-measure, and accuracy.

REFERENCE

- Nowland, R., Necka, E. A., & Cacioppo, J. T. (2018). Loneliness and social internet use: pathways to reconnection in a digital world?. *Perspectives on Psychological Science*, 13(1), 70-87.
- Miller, D. (2020). Social networking sites. In *Digital anthropology* (pp. 146-161). Routledge.
- Cao, X., Khan, A. N., Ali, A., & Khan, N. A. (2020). Consequences of cyberbullying and social overload while using SNSs: A study of users' discontinuous usage behavior in SNSs. *Information Systems Frontiers*, 22(6), 1343-1356.
- Patchin, J. W., & Hinduja, S. (2015). Measuring cyberbullying: Implications for research. *Aggression and Violent Behavior*, 23, 69-74.
- Deschamps, R., & McNutt, K. (2016). Cyberbullying: What's the problem?. *Canadian Public Administration*, 59(1), 45-71.
- Kubiszewski, V., Fontaine, R., Potard, C., & Auzoult, L. (2015). Does cyberbullying overlap with school bullying when taking modality of involvement into account?. *Computers in Human Behavior*, 43, 49-57.
- Jane, E. A. (2015). Flaming? What flaming? The pitfalls and potentials of researching online hostility. *Ethics and Information Technology*, 17(1), 65-87.
- Watts, L. K., Wagner, J., Velasquez, B., & Behrens, P. I. (2017). Cyberbullying in higher education: A literature review. *Computers in Human Behavior*, 69, 268-274.
- Ali, W. N. H. W., Mohd, M., & Fauzi, F. (2018, November). Cyberbullying detection: an overview. In *2018 Cyber Resilience Conference (CRC)* (pp. 1-3). IEEE.
- Novitasari, N. F., & Hia, N. I. A. (2021). CYBERBULLYING IN MOVIE CYBERBULLY: AN ANALYSIS FROM THE PSYCHOLOGICAL PERSPECTIVE. *Celtic: A Journal of Culture, English Language Teaching, Literature and Linguistics*, 8(1), 44-64.
- Al-Rahmi, W. M., Yahaya, N., Alamri, M. M., Aljarboa, N. A., Kamin, Y. B., & Saud, M. S. B. (2019). How cyber stalking and cyber bullying affect students' open learning. *Ieee Access*, 7, 20199-20210.
- Vyawahare, M., & Chatterjee, M. (2020). Taxonomy of cyberbullying detection and prediction techniques in online social networks. In *Data communication and networks* (pp. 21-37). Springer, Singapore.
- Iwendi, C., Srivastava, G., Khan, S., & Maddikunta, P. K. R. (2020). Cyberbullying detection solutions based on deep learning architectures. *Multimedia Systems*, 1-14.
- Balakrishnan, V., Khan, S., & Arabnia, H. R. (2020). Improving cyberbullying detection using Twitter users' psychological features and machine learning. *Computers & Security*, 90, 101710.
- Cheng, L., Guo, R., Silva, Y., Hall, D., & Liu, H. (2019, May). Hierarchical attention networks for cyberbullying detection on the instagram social network. In *Proceedings of the 2019 SIAM international conference on data mining* (pp. 235-243). Society for Industrial and Applied Mathematics.
- Singh, V. K., Ghosh, S., & Jose, C. (2017, May). Toward multimodal cyberbullying detection. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (pp. 2090-2099).
- Van Hee, C., Jacobs, G., Emmerly, C., Desmet, B., Lefever, E., Verhoeven, B., ... & Hoste, V. (2018). Automatic detection of cyberbullying in social media text. *PLoS one*, 13(10), e0203794.
- Haidar, B., Chamoun, M., & Serhrouchni, A. (2017). A multilingual system for cyberbullying detection: Arabic content detection using machine learning. *Advances in Science, Technology and Engineering Systems Journal*, 2(6), 275-284.
- Agrawal, S., & Awekar, A. (2018, March). Deep learning for detecting cyberbullying across multiple social media platforms. In *European conference on information retrieval* (pp. 141-153). Springer, Cham.

- Gencoglu, O. (2020). Cyberbullying detection with fairness constraints. *IEEE Internet Computing*, 25(1), 20-29.
- Balakrishnan, V., Khan, S., Fernandez, T., & Arabnia, H. R. (2019). Cyberbullying detection on twitter using Big Five and Dark Triad features. *Personality and individual differences*, 141, 252-257.
- Iwendi, C., Srivastava, G., Khan, S., & Maddikunta, P. K. R. (2020). Cyberbullying detection solutions based on deep learning architectures. *Multimedia Systems*, 1-14.
- Yao, M., Chelms, C., & Zois, D. S. (2019, May). Cyberbullying ends here: Towards robust detection of cyberbullying in social media. In *The World Wide Web Conference* (pp. 3427-3433).
- Raisi, E., & Huang, B. (2017, July). Cyberbullying detection with weakly supervised machine learning. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017* (pp. 409-416).

AUTHOR'S PROFILE



Ms. Sherly.T.T is a researcher of Computer Science at Dr.NG.P. College of arts & science, Coimbatore, India. Sherly obtained her bachelor's degree from University of Calicut, Kerala, India in 2002 and her Masters in Computer Applications from Mahatma Gandhi university, Kottayam, Kerala in 2006. She obtained her M.Phil- Computer Science from Bharathiyar uinersity in the year 2013. She is currently a Ph. D. scholar under the supervision of Prof. Dr. B. ROSILINE JEETHA. Her research is centered on detection of cyber bullying text in social media platforms especially on twitter dataset. She has coauthored 10 publications including Data mining in KDD, forged assessment and analysis of online reviews, cyber bullying detection using SFS-ELM classifier, SSVM classifier, Hybrid Firefly Artificial Neural Network Algorithm For Cyber bullying Detection and changing trends in clod computing.



Dr. B. Rosiline jeetha, heads Computer Science Department, Dr.NG.P. College of arts & science, Coimbatore, India, an intellectual destination. Dr.NGP CAS is a part of Bharathiyar University, Ciomabtre. Dr. B. ROSILINE JEETHA is a leading and internationally respected researcher with awards for outstanding and innovative research. She also have membership in Professional Bodies like Peer Reviewer in Springer Sponsored International Conference on Informatics Engineering and Information Science ICIEIS, Malaysia, International Journal of Advance Research in Computer Science and Management Studies and Technical Committee member in IAJIT (The International Arab Journal of Information Technology). She received Most Active Reviewer Award in the year 2016 for the Journal "The International Arab Journal of Information Technology (IAJIT)". She received Real Heroes Award in Education from Youth Foundation Coimbatore in 2017 and Best Teacher Award from Lions Club, Coimbatore in 2019. She is the member of Board of Studies (BoS) of Computer Science, Kongu Arts and Science College, Erode. She is having 28 publications in International and national journals and numerous presentations in conferences.