I. J. Computer Network and Information Security, **2025, 3, 144-150**
Published Online on June 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2025.03.08

Multi-modal Cyberbullying Detection with Severity Analysis Using Deep-Tensor Fusion Framework

Neha Minder Singh*
Computer Science and Engineering, Oriental University, Indore, Madhya Pradesh, 453555, India
E-mail: neha.cse@orientaluniversity.in
ORCID iD: https://orcid.org/ 0009-0003-0771-3679
*Corresponding author

Sanjay Kumar Sharma
Computer Science and Engineering, Oriental University, Indore, Madhya Pradesh, 453555, India
E-mail: sanjaysharmaemail@gmail.com
ORCID iD: https://orcid.org/ 0000-0002-7965-6517

Received: 04 November 2023; Revised: 14 February 2024; Accepted: 20 May 2024; Published: 08 June 2025

Abstract: In today's internet age, cyberbullying is a serious threat to people. Bullies hurt their victims by using online social media sites like Facebook, Twitter, and so on. Since then, cyberbullying or online bullying has become an increasingly significant societal problem due to its psychological and emotional impact on the victim. To overcome this issue, the proposed method is used for the identification of social network cyberbullying using many modalities. The first step is data collection, the data are collected through text, image, audio and video. Firstly, the text data is pre-processed using tokenization, stemming, lemmatization, spell correction, and PoS tagging methods. After pre-processing, feature extraction is done using Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF) and glove modelling, and the feature is extracted using the convolutional dense capsule network (Conv_DCapNet) model. Second, image data is pre-processed using Gabor filtering and image resizing. Once the pre-processing is done, the feature is extracted using image modality generation through Self-attention based bidirectional gated recurrent auto-encoder network (SA_BiGR_AENet). Third, audio (acoustic) data is extracted using the Librosa library and Attentional convolutional BiLSTM. The next step is video extraction, in this video data are extracted using keyframe extraction and Residual 101 network. Finally, in the fusion method, all four extracted features (text, image, audio and video) are fused by the deep tensor fusion framework. After that, the sigmoid layer classifies whether the data is cyberbullying or not. Then, the notification of highly detected severe is sent to the user. Thus, the proposed method detects cyberbullying. Python tool is used for implementation, and the performance metrics of recall is 96%, F1-score for bullying class 92% and weighted F1-score 91% compared with existing methods.

Index Terms: Cyberbullying, Stemming, Dense Capsule Network, LSTM, Sigmoid Layer.

1. Introduction

Cyberbullying is a serious issue that has negative effects on teenagers. It is well known that bullying as a child can lead to psychological problems such as anxiety and despair [1]. Cyberbullying perpetration and victimization are high, which includes verbal or physical abuse. Numerous research shows that the global incidence of cyberbullying victimization is rapidly rising [2]. In seven European Union nations, the percentage of teenagers aged 9 to 16 who were victims of cyberbullying increased from 8% to 12%. Additionally, 7 to 50% of teenagers in the United States as well as Canada are victims of cyberbullying. The conflicting results of the cyberbullying study can be due to a variety of other factors in addition to the actual discrepancies. However, measurement and definition problems can be mainly responsible for the large variation [3, 4].

Cyberbullying is simply bullying that takes place using technology and the internet. [5]. Cyberbullying is described in some research as bullying via email and instant messaging, online chat rooms, or text messages received from a cell phone. Therefore, the conceptualization of the main concepts of the study is crucial as it not only determines what is measured but also influences the study's results [6, 7]. Since cyberbullying can speedily reach many people, its influences

are larger and longer-lasting than old bullying [8]. Online bullying doesn't seem to directly injure the victim physically. Still, it may potentially result in psychological diseases like despair, self-confidence issues, attention problems, and even suicidal thoughts [9, 10].

Cyberbullying has increased significantly over the past decade, particularly affecting children and young adults. The collected results show that this problem is rising rapidly, regardless of the country's level of development. Online bullies show higher sadness, fear and loneliness levels than traditional victims. Cyberbullying has also been linked to self-esteem issues and school absenteeism [11, 12]. The manifestations and risk factors of cyberbullying have changed significantly as young people's social media usage and behaviour patterns have evolved. In addition, since cyberbullying knows no national borders, it may not be a problem that only occurs in one country. In this respect, cyberbullying is a global problem, the solution of which requires increased international cooperation. Cyberbullying has a negative impact on children's safety, educational achievement and mental health, although Happiness and UNICEF have stated that no child is completely safe in today's digital world [13, 14].

The importance of interpreting multi-modal information in detecting cyberbullying is widely recognized. The (advanced) text processing remains the main emphasis of cyberbullying detection writing, and accuracy remains only moderate [15]. There are not many attempts to use visual indicators to detect cyberbullying. This is partly due to the extensive knowledge required to solve image or video editing challenges. Image analysis is now more accessible to a larger pool of HCI and data scientists due to the availability of modern computer vision application programming interfaces (APIs) known as Face++, Imagga, Microsoft's Project Oxford and CMU's Open Face [16]. Most recent studies used traditional machine learning (ML) models to identify cases of cyberbullying [17, 18].

Newer models created on deep neural networks (DNNs) have also been used to identify cyberbullying. DNN models for identifying cyberbullying have been extended to numerous social media sites. Other studies created models for identifying cyberbullying that, according to their reported results, are superior to traditional ML models and, more importantly, can be modified and applied to other datasets [19, 20]. In recent years, several methods are developed for detecting cyberbullying from multi-modal inputs. However, achieving better accuracy is becoming a great issue because of the reduced ability of existing methods. Also, the existing methods struggled to handle different modalities of inputs because of the complex patterns. In addition, feature learning ability is limited in the existing methods and hence achieved reduced detection performance. Thus, the proposed study aims to solve the mentioned limitations faced in the existing methods and achieve superior results while detecting cyberbullying from the multi-modal inputs like text, image, audio and video.

1.1. Motivation

Cyberbullying mainly affects computers, which is quite different from other forms of bullying. Bullying can be defined as thoughtful, harmful, recurrent, and an indication of power abuse. Using an IP address, every interaction may be tracked back, enabling law enforcement. A previous study has also used image analysis to detect bullying content. This article uses a multi-model approach to detect cyberbullying content.

The major contribution of the work is:

- Multi-modal cyberbullying is used to determine text, image, audio and video data.
- The input text data is pre-processed using four techniques, and feature extraction is done using LTF-MICF, glove modelling and Conv_DCapNet.
- Likewise, image, audio (acoustic) and video data are also extracted using Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF) as well as glove modelling techniques, which helps to reduce the complexity issues.
- To combine this extracted data, the Deep-tensor fusion method is used for fusion the features attained from multi-modalities and classification is done by adopting sigmoid layer.

1.2. Paper Organization

The remaining work of this paper is described as follows: Section 2 presents the related work in cyberbullying. Section 3 presents some proposed techniques used to detect cyberbullying. Section 4 contains the evaluation metrics and dataset used in this paper, and Section 5 presents the conclusion of this study.

2. Related Works

Kumari et al. [21] designed a multi-model cyber aggression detection functional optimization using the Firefly algorithm. In this existing work, a binary firefly-based optimization model and deep learning technique are developed to categorize social broadcasting pillars into non-aggressive, highly aggressive, as well as moderately aggressive modules. To assess the aggressiveness level of the posts, this model compares the images and the text. Using the three-layer CNN in parallel extracts the features of text posts, while using the pre-trained VGG-16 model extracts the image features. In addition, text and image features are united to obtain a hybrid article set that is then enhanced by a binary firefly optimization method. However, this technique requires more training to find the criminal on social media.

A1-Harigy et al. [22] presented a comparative analysis of automated cyberbullying detection. This latest report

presents a comparative study of the automated cyberbullying technique from a different angle, involving data pre-processing, data annotation, and feature engineering. In addition, text comprehension shows the impact of integrating emojis on emotion classification and the importance of emojis in conveying feelings and in the practice of cyberbullying detection. Also, the use of self-supervised learning as an annotation method for altered domains for cyberbullying detection is discovered. Using this technique, removing emojis from a sentence changes polarity and considers an incorrect outcome.

Murshed et al. [23] developed an identification of cyberbullying using deep learning and topic modeling in social media. This paper uses the FAEO-ECNN model to identify and categorize cyberbullying on social media platforms. To increase the accuracy of cyberbullying detection, this approach uses fuzzy-adaptive equilibrium optimization, clustering-based topic modeling, and augmented CNN. This existing work estimated the FAEO-ECNN contain two datasets: Real-World CB-Twitter and Cyberbullying Menedely datasets evaluated with state-of-the-art models known as Bidirectional, Long-Term Short-Term Memory and CNN-LSTM and RNN. Only English-language comments are checked here, while online operators usually use multilingual comments.

Maity et al. [24] preferred code-mixed meme cyberbullying detection using multiple models of a multitasking framework for feeling, emotion, and sarcasm detection. This existing work examines emotions, sarcasm, and feelings to find cyberbullying using multi-model code-mixed language memes. The MultiBully designations marked with emotions, sarcasm and sentiment are collected from the data set of the benchmark multi-model meme by the open-source platform Reddit. Image and text are the two models included in the dataset. Two multi-modal multitask models such as BERT+ResNET-Feedback and CLIP-CentralNet are used for cyberbullying detection. This technique has no other data sets, including the audio and video data.

Abishak et al. [25] offered cyberbullying detection on Instagram using unsupervised hybrid approaches. In this report, linguistic features such as irony, sarcasm, active or passive voice and idioms are extracted by the unsupervised cyberbullying detection technique. In addition, the clarifications received from Instagram are demonstrated through a learning network that learns multi-model sessions and multitasking to immediately assess and stimulate bullying strength. Unattended cyberbullying detection is used to detect abuse through neural networks. This technique only considers the input of multiple models such as text, image, audio and video. Using these techniques introduces large errors and correlates the classifiers.

Problem Statement

Cyberbullying detection has become one of the trending topics on social media. To detect the cyberbullying action, the Firefly algorithm is used to categorize the social media posts as aggressive and non-aggressive. Three-layer CNN and pre-trained VGG16 techniques play an important role in this. This technique only identifies cyberbullying in images and text. Emoji's-based cyberbullying is detected through the comparative analysis of automated cyberbullying techniques that include data annotation and data pre-processing. To overcome these problems, a proposed multi-model cyberbullying detection technique to instantly and accurately detect cybercriminals in social media.

3. Proposed Methodology

The Multi-Modal Framework is used in this study to propose an automated approach to identify multi-modal cyberbullying through social network. Data from numerous websites, including text, voice, image and video, is used for multi-modal cyberbullying detection. Firstly, the data collection, and the data such as text, image and audio are collected from various social network sites. This data is delivered as input to the proposed multi-model cyberbullying detection layer. The next step is modality generation, and the inputs are pre-processed, extracted and merged by proposed techniques. Figure 1 displays the architecture of the proposed method.

Modality generation pre-processes the text data using tokenization, stemming, lemmatization, spell correction, and PoS tagging. After the pre-processing, the feature extraction extracts the text data using Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF) and glove modeling. Next, features are extracted to pass through the Convolutional Dense Capsule Network (Conv_DCapNet) model aimed at text group. Next, the image modality generation is done by the Bidirectional gated recurrent auto-encoder network (BiGR_AENet) model and the images are extracted. The next step is audio model extraction, the acoustic model is extracted using Librosa library and given to the Attention convolutional BiLSTM. The next step is video modality generation, in this step, videos are extracted using keyframe extraction and fed to the Residual 101 Network. Finally, in the fusion method, the extracted features are given to the Deep-tensor fusion framework, and then the sigmoid layer will classify and detect the cyberbullying.

3.1. Data Collection

Gathering data is the first step in the proposed approach to detecting cyberbullying. This proposed technique collects the text, image, audio and video data using multiple models from different datasets. Then, the collected input data is passed through the multi-model generation. Some of the proposed studies on multi-model generation are explained below.

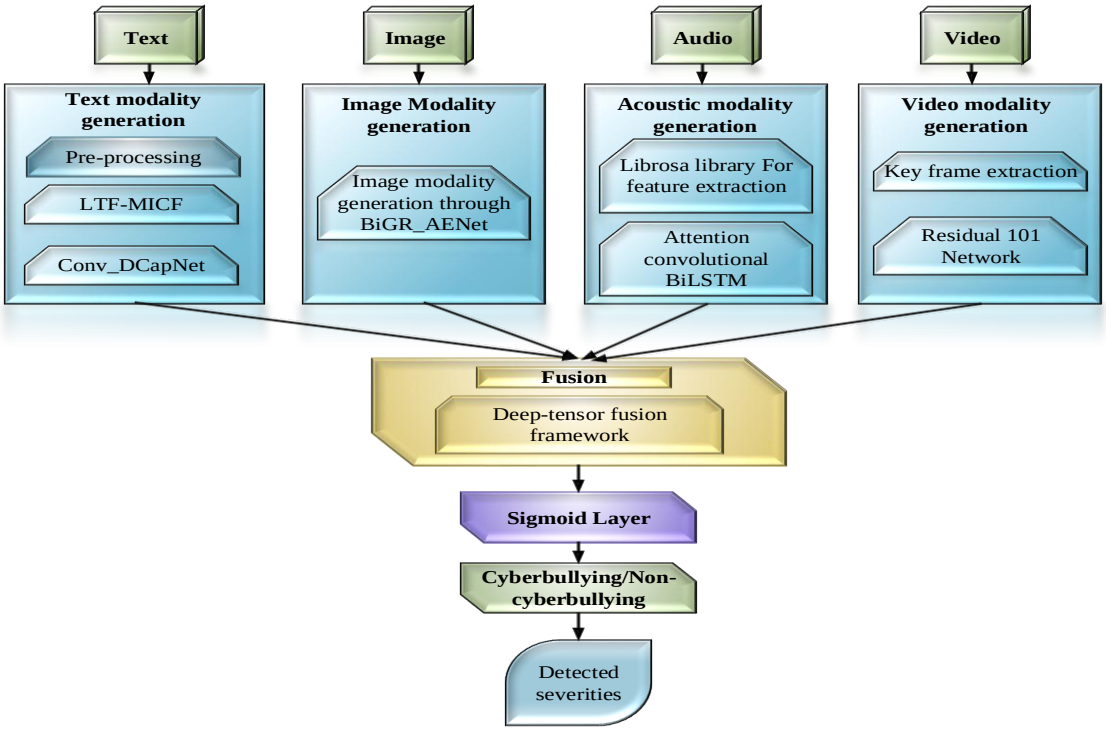


Fig.1. Architecture of the proposed method

3.2. Multi-model Generation

The creation of different models to analyse text, image, audio and video input files represents the multi-modal production method. The generation of multiple models can be done in four types of data, which are explained below.

A. Text Modality Generation

In text modality method, text is pre-processed and the feature extractions take place.

a. Text Pre-processing

Text pre-processing means preparing unstructured text data for analysis by cleaning and converting the text. The different text pre-processing steps are tokenization, stemming, lemmatization, spelling correction and PoS tagging used for dimensionality reduction. These text pre-processing steps are explained below.

- *Tokenization*: Tokenization involves replacing sensitive data, such as credit card numbers, with a token or other substitute value. The sensitive data often still needs to be kept safe in a single location with high security measures. The protection in a tokenization strategy is based on the security associated with the sensitive values, the algorithms and methods used to generate the replacement value, and the process by which it is mapped back to its original value.
- *Stemming:* Stemming, a natural language processing method, is used to condense words into their stem or base form. Stemming defines a component of linguistic research in extracting and querying morphological information using artificial intelligence (AI). Both natural language processing (NLP) and natural language understanding (NLU) depend on stemming.
- *Lemmatization:* Lemmatization often refers to the correct execution of tasks using a vocabulary, as well as the morphological evaluation of words to remove only inflectional endings or return the lemma that represents the word base and dictionary form.
- *Spelling correction*: One of the most important tasks in natural language processing is spell correction. It is used for many purposes, including text summarization, sentiment analysis, and search engines. It is also used to find and eliminate spelling mistakes during spelling correction.
- *PoS tagging*: Part-of-speech (PoS) tagging, also known as grammatical tagging in corpus linguistics, refers to the process of tagging a word within a text (corpus) as belonging to a specific part of speech, depending on its context and definition.

b. Text Feature Extraction

Feature Extraction based on text is the method for turning text data into numbers. In feature extraction, two steps are taken: Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF) and glove modelling. After extraction, the features were given to the Convolutional dense capsule network approach (Conv_DCapNet). Another

name for text extraction is text vectorization.

***Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF*)**: LTF-MICF methodology gather two weighting methods, Log Term Frequency (LTF)and Modified Inverse Class Frequency (MICF). The frequency of a term in a set of reviewing sentences was measured byL_q. However, L_q alone remains insufficient because a document can be heavily biased by the locations that looks more frequently. The use of class data gathered from reviews for supervised vLschemes has drawn increasing interest.

When calculating LTF centredvL, computes L_q for each duration through pre-processed dataset, then performs log standardisation on the data productivityL_q, denoted as LTF. The altered method ofjx_q, recognized as MICF, which is calculated for every term. Given this, the modification of $jx_q$is supported out since distinct class-specific notches aimed at every term must consume changing contributions to the complete period score.

$$LTF - MICF(l_k) = T_{lq} * \sum_{p-1}^{m} v_{kp}.[jx_q(l_k)] \tag{1}$$

In this sentence, v_{kp} refers to the precise weighting value for classx_c, which is well-defined as

$$v_{kp} = log\left(1 + \frac{f_j\vec{l}}{max(1, f_j\vec{l})}.\frac{f_j\vec{l}}{max(1, f_j\vec{l})}\right) \tag{2}$$

The approach used to allocate weight through the obtainable datasets known as weighting factor (WF). Here, $f_j\vec{l}$ represent the quantity of reviews (f_j) on a class x_c containing the term l_k, $f_j\vec{l}$ represent the quantity of $f_j$having the term t_{lq}in different classes. $f_j\vec{l}$ denotes the quantity of f_j in class x_c excluding the term l_k, and $f_j\vec{l}$ denotes the quantity of f_j lacking the terms l_k in other classes.

$$T_{lq}(l_k) = log\left(1 + L_q(l_k, f_j)\right) \tag{3}$$

Where $T_{lq}(l_k)$ is the entire occurrence of an expression l_k on a group of document reviewsf_j, i.e., the number of times the term l_k appears overall on the entire set of paper reviewsf_j.

$$j_{x_q}(l_k) = log\left(1 + \frac{c}{x(l_k)}\right) \tag{4}$$

Where $x(l_k)$ denotes the number of programmes which contain the terml_k, and c represent the entire number of classes in the group of review documents. So, the modality generation of text data is exhibited.

***Glove modelling*:** The GloVe model attempts to analyse a word vector such that the product of each word's points equals the logarithm of the possibility that such words will appear together or occur simultaneously. The following can be used to express the glove model.

$$v_j^R + \vec{v}_l + \vec{a}_l = log(Y_{jl}) \tag{5}$$

Where a_j and a_l are scalar biased representing the context of the$j$ and l -word, respectively, and $v$is the word vector as well as a is the contextual word vector. In Y_{jl} matrix,Y represents how many times a word j occurs in the l -word context. The following equation can be used to compute the weight's function, $g(Y_{jl})$.

$$g(Y_{jl}) = \begin{cases} \left(\frac{Y_{jl}}{y\,max}\right)^{\alpha}; & if\ Y_{jl} < ymax \\ 1; lainnya \end{cases} \tag{6}$$

Then, create the following model by including equations (5) and (6) in a cost function.

$$I = \sum_{j,l=1}^{w} g(Y_{jl})\left(v_j^R + \overrightarrow{v_j} + a_j + \overrightarrow{a_j} - log(Y_{jl})\right)^2 \tag{7}$$

So, the feature extraction is done using glove modelling. After feature extraction, the data are given to the Conv_DCapNet model.

Convolutional dense capsule network model (Conv_DCapNet): In this step, the extracted datas then passed to the convolutional approach of the dense capsule network. Here, the convolution layer, the max-pooling layer, and the dense

capsule layers are used to detect the modality of text data and generate the output. CNNs identify spatial features such as corners, textures, edges or more abstract shapes that best characterize the set or target class. Figure 2 shows the framework of the proposed Conv_DCapNet model.

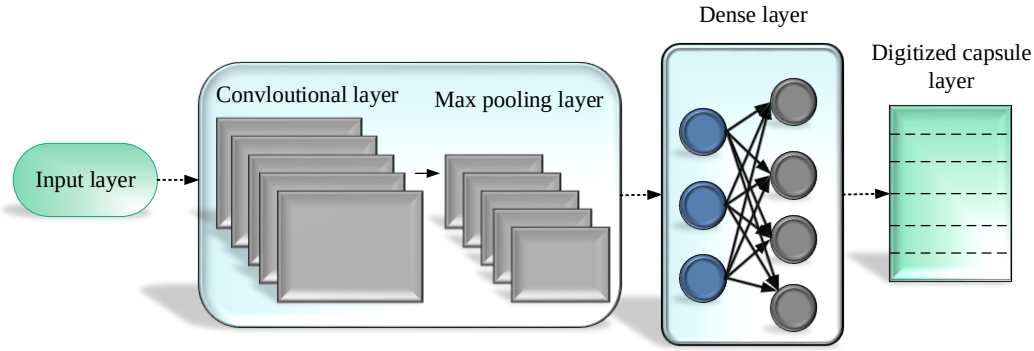


Fig.2. Flowchart of proposed Conv_DCapNet model

- *Convolutional layer*: A convolutional level is the basic module of a CNN and most of the feature processing takes place. This method requires input data, a feature map, and a filter, between other effects. The feature detector is a two-dimensional (2-D) array of values that represents a percentage of the image. The receptive field's size is also determined by the filter size, which is generally a 3x3 matrices, though it can differ in size.
- *Max-pooling layer:* Maximum pooling refers to selecting the most numerous elements from the region of the feature map that is included in the filter's analysis. The max-pooling level produces a feature map that incorporates the best salient aspects of the original feature map.
- *Dense layer:* Each neuron in the simple section of neurons, known as the dense layer, receives information from every cell in the layer. According to the results of convolutional layers, the image is classified using a dense layer. Such neurons are found in large numbers inside a layer. Dense connectivity and direct connections from one layer to all previous layers are the fundamental components of a dense network topology. The output of the dense layer is given as

$$i = \Phi^i \in T^{(v^i, v^i, a^i, m^i)} \tag{8}$$

$$\Phi^{i+1} \in T^{(v^{i+1}, v^{i+1}, a^{i+1}, 2m^{i+1})} \tag{9}$$

Where $v^{i+1}, a^{i+1}, m^{i+1}$ and v^i, a^i, m^i are same. Each block has $\frac{I(I+1)}{2}$ connections made between the I capsule's layers. DenseCaps accepts photos as input and uses capsules to represent the stream's data unit. The multiple levels capsule fusion representation of features is more suited for issues with image classification because it learns through feature reuse. So, the text modality generation is extracted. The next step is image modality generation.

B. Image Modality Generation

Imaging modality method is classified into three groups: ultrasound, radiation like X-rays, and MRI. In this proposed method, the images are pre-processed using Gabor filtering and image resizing. Once the pre-processing is done, the pre-processed images are given to the self-attention based bidirectional gated recurrent with auto decoder network (SA-BiGRU-AENet) model. These techniques are briefly explained below:

a. Image pre-processing

The word "image pre-processing" states to movements performed on images through the best basic level of concept. In this proposed method, image pre-processing is done by two steps: Gabor filtering and image resizing.

- *Gabor filtering*: Recently, Gabor filter techniques has increasingly gathered in image processing, especially for texture analysis. Elementary Gabor functions are sinusoidal modified Gaussian functions. It is shown that Gabor filtration is a successful method for displaying images and that the functional form of Gabor filters resembles the reception properties of basic cortical cells. The following equation can be used to express a two-dimensional (2D) equal Gabor filter in the following spatial domain:

$$F(a, b; \theta, g) = exp\left\{-\frac{1}{2}\left[\frac{a^2}{\delta_{a'}^2} + \frac{b'^2}{\delta_{b'}^2}\right]\right\} cos(2\pi g a') \tag{10}$$

Where $a' = a\cos\theta + b\sin\theta$ and$b' = b\cos\theta - b\sin\theta$. Where a' and b' are the space coefficients for the Gaussian envelope across the x as well as y axes, respectively, and g represents the frequency for the sinusoidal plane

waves along the direction θ away from the x-axis.

- *Image resizing:* Image Resizing refers to changing an image's size. Remapping can occur when adjusting lens distortion or inverting an image, whereas image scaling is required when the total number of pixels needs to be changed. It makes removing pixels in an image easier, which has numerous advantages. It also makes it easier to zoom images. The image size was chosen to be 256 x 256 x 3 pixels to simplify the complicated structure of the original image size. The input data type was created using bicubic interpolation to resize the images.

b. Image Modality Generation through SA-BiGRU-AENet Model

The proposed technique used the self-attention based bidirectional gated recurrent with auto decoder network (SA-BiGRU-AENet) model to generate image modality. These techniques are proposed for detecting cyberbullying action on social media. The self-attention model reduces the maximum path distance between the sites of two input and output composed of the altered layer types. Figure 3 represents the block diagram for SA based BI-GRU with AENet.

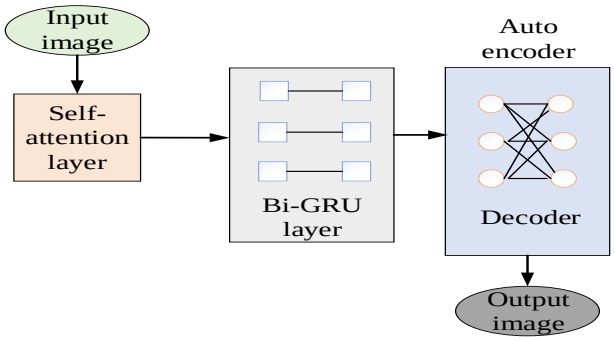


Fig.3. Block diagram for SA based Bi-GRU with AENet

Self-Attention layer (SA): The SA layer maps the input image to every convolution block, and the feature maps are removed on the preceding convolution layer. The attention value gained by all particular image with altered weights are expressed as,

$$N_n = \frac{1}{B(M_n)} \sum_q r(M_n, M_q) k(M_q) \tag{11}$$

Where, $n$denotes the position at which the reply was figured, $q$enumerates all positions in a particular image, a scalar is considered as $r$function that indicates the relevance between the intensity of the signal at the position $n$and$q$. At any position of $q$the illustration of the input signal is figured by$k$. The corresponding position of the signal is directly contributed to the current position signal. Figure 4 represents the framework of the self-attention layer.

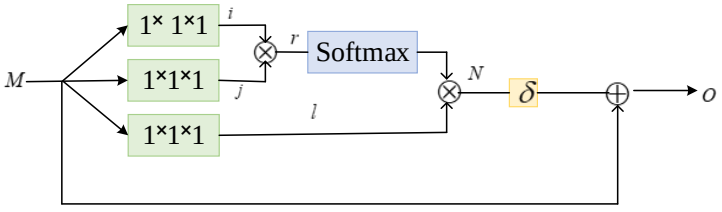


Fig.4. Self-attention layer framework

The contribution at a distance is determined by the significance and strength of the signal. The result was subsequently regularised, where the feature$B(M_n)$ is expressed as

$$B(M_n) = \sum_q r(M_n, M_q) \tag{12}$$

The utilities $r$and $k$were taken with a certain amount of flexibility. The linear utility for minimalism was defined by$k$.

$$k(M_q) = D_k M_q \tag{13}$$

To calculate the relevance between $M_n$and M_q there was a usual choice called the Gaussian function.

$$r(M_n, M_q) = d^{\vartheta(M_n)^P \lambda(M_q)} = d^{(D_i \times M_N)^P (D_j M_q)} \tag{14}$$

Here, weight matrices are D_i, D_j, D_lexecuted as $1 \times 1 \times 1$ convolution, which were cultured in the training network. In the attention layer, residual learning was hired as specified,

$$O_n = M_n + \delta N_n \tag{15}$$

Hence, the self-attention layer output (O_n) was composed of two modules: the first module is local dataM_n, from the convolutional feature maps, and the next module is non-local dataN_n, from the self-attention maps. Through training, the local and non-local modules in the results are discovered, and the offerings are balanced using the scale parameterλ. Like the convolutional network, SAT-Net had a learning performance and $\lambda$was fixed as 0. Self-attention layers progressively took results for achieving a flat transition to eminent self-attention CNN. During optimization $\lambda$was enriched. The data can be used effectively and perfectly integrated from widely separated spatial areas of attention.

Bidirectional Gated Recurrent Unit (Bi-GRU) layer: One type of recurrent neural system is the GRU. By combining the forget and input gates, GRU is used to create a more efficient structure of equal update gates. The word vector is simultaneously used in both the reverse and forward directions by the two halves of the Bi-GRU. The GRU calculates the output vector of static dimensions and the passing word vectors. This includes four different calculations:

The reset gate is the first part of GRU. For selecting the data at the previous program, $M_t$and$Q_t$represent output data, P_{l-1} represent the input of the preceding program andC_t represent the bias.

$$K_l = \lambda(M_t R_l + Q_t P_{l-1} + C_t) \tag{16}$$

The update gate is the second part of GRU. GRU is selected and upgraded the data in the current program by the update gate. Where the weight of data is $M_y$andQ_y, the input of the preceding program isP_{l-1}, $C_y$is the bias:

$$Y_l = \lambda\left(M_y R_l + Q_y P_{l-1} + C_y\right) \tag{17}$$

Next, the memory content of the applicant from GRU is taken into account for the calculation of the current program, where M and $Q$are weight data, $C$is the bias:

$$\hat{P}_l = tan\, l\,(M R_l + Q K_l P_{l-1} + C) \tag{18}$$

From the above outcomes, GRU estimates the output:

$$P_l = (1 - Y_l)\hat{P}_l + Y_l P_{l-1} \tag{19}$$

Auto-encoder network (AENet): Auto-encoder is an unsupervised learning method for neural networks. It absorbs an effective data illustration by training the network to disregard the noise signal from the input image. The encoder as well as decoder are the two parts of the auto-encoder. Assume $a$represent the first input, which belongs to $t$dimensional space, then the new input $b$which belongs to $u$dimensional space. An auto-encoder is a unique and complex three layered neural network, here the result $k_{R,c}(a) = (\tilde{a}_1, \tilde{a}_2, \cdots, \tilde{a}_t)^S$is equal to the input$a = (a_1, a_2, \cdots, a_t)^S$. The reconstruction error is$P$. For training purposes, a back propagation algorithm is used.

$$k_{R,c}(a) = i\big(l(a)\big) \approx a \tag{20}$$

$$P(R, c; a, b) = \frac{1}{2}\left\|k_{R,c}(a) - b\right\|^2 \tag{21}$$

Auto-encoder can be perceived as a method for changing the illustration. The result will be related to the sparse coding by limiting the nodes of the hidden layer$u$less than the nodes of the first input$t$ and the constraint of sparsity$k$. Then, a compressed illustration of the input is obtained that achieves the desired dimensionality reduction result. So, the image data is extracted using this proposed technique.

C. Acoustic Modality Generation

This method is used to extract audio data from the audio features. In this method, the Librosa library is used for audio feature extraction. Librosa refers to a Python library for analysing music and audio. Some features used for extracting acoustic generations are Constant-Q chromagram, Variable-Q chromagram, Compute a Mel-frequency cepstral coefficients (MFCCs) and mel-scaled spectrogram. The features were explained below.

- *Constant-Q chromagram:* It has connections to both the Fourier transforms as well as complex Morlet wave transform. Its structure lends itself to musical depiction. The waveform of the C major piano chords was transformed using a Constant Q algorithm.

$$\delta g_l = 2^{1/m} \cdot \delta g_{l-1} = \left(2^{1/m}\right)^l \cdot \delta g_{min} \tag{22}$$

If δg_l is the l -th filter's bandwidth, the lowest filter's centre frequency isg_{min}. Where m denotes the number of filters used in each octave.

- *Variable-Q chromagram*: The main difference between the variable-Q transform and the constant-Q transform is that the filter varies Q, called the variable-Q transform. Adding a high frequency offset similar to this is the simplest approach to creating the variable Q transform.

$$\delta g_l = \left(\frac{2}{\alpha\sqrt{g^{\alpha}{}_l + \gamma^{\alpha}}}\right) W \tag{23}$$

In terms of energy solution, using δ as a variable for transitional sharpness and α as a hyperbolic sine frequencies scale.

- *Compute a mel-scaled spectrogram*: The mel spectrogram differs from a normal frequency-time spectrogram in two essential ways. On the y-axis, the Mel scale is used instead of frequency. Also, colors are displayed using the decibel scale instead of amplitude.

$$n = 2595\, log_{10}\left(1 + \frac{g}{700}\right) \tag{24}$$

- *Mel-frequency cepstral coefficients (MFCCs):* Like Chroma or spectral properties, Mel-frequency cepstral coefficients (MFCCs) represent a group of characteristics. It is also used to simulate the features of a human voice. Mel-frequency Cepstrum (MFC) is the linear cos transformation for a logarithmic power spectrum on a nonlinear Mel scale during frequency.

Once the extraction is done, the input of the acoustic data is given to the Attentional Convolutional BiLSTM model. In CNN, the collection of shallow characteristics is crucial since it directly impacts the precision of high-level characteristics and the precision of the final categorization. Figure 5 shows the architecture of Attentional convolutional BiLSTM.

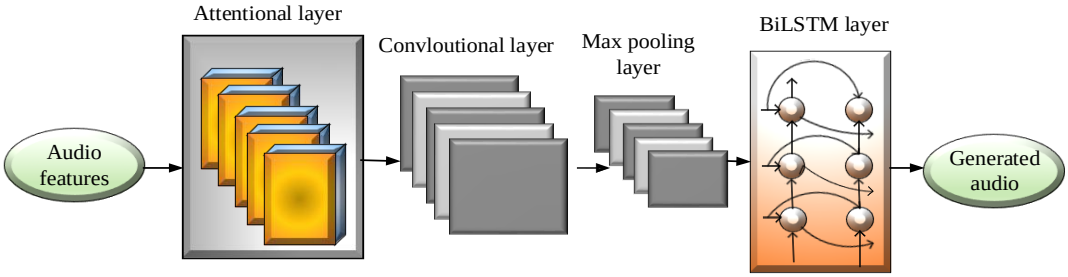


Fig.5. Flowchart of attentional convolutional BiLSTM

- *Attention model:* The attention model is a representation of the connection between inputs and outputs. To improve prediction accuracy and emphasize aspects of importance, the attention approach employs feature vectors to differentiate between different weights for attention. The calculation of the target attention weight (t_v) is mentioned below.

$$t_v = tan\, g\,(g_v) \tag{25}$$

The connected hidden layer variable g_v receives the resultant attention value. So, this model produced an attention weight and calculated the mean valueb_v, using probabilistic attention weight:

$$q_v = \frac{exp(t_v)}{\sum_{v=1}^{n} exp(t_v)} \tag{26}$$

$$b_v = \sum_{v=1}^{m} q_v \bullet g_v \tag{27}$$

- *Convolutional layer*: Convolutional layers collect the input and send the resultant information to the next layer. Convolution limits the number of open parameters, enabling a deeper network. Instead of regular convolution layers, depth-separated convolutions can speed up processing.
- *Max pooling layer*: The term "max pooling" refers to a pooling process that chooses the largest element within the feature map area where the filter covers. The result from the max-pooling layer from the feature map included the clearest features from the former feature map.
- *BiLSTM network*: Another name for the RNN version is LSTM. It consists of a storage module that continuously runs. The hidden state is represented asf_{a-1} , and the current input y_a are used to determine the forgetting gateg_a, memory gatej_a, and output gateu_a. It reduces the repeating neural network problem by preserving important information and forgetting irrelevant information.

$$g_a = log\ i\ stic\left(V_g y_a + Q_g f_{a-1} + x_g\right) \tag{28}$$

$$j_a = log\ i\ stic\left(V_j y_a + Q_j f_{a-1} + x_j\right) \tag{29}$$

$$Z_a = j * \tilde{Z}_a + g * Z_{a-1} \tag{30}$$

$$\tilde{Z}_a = tan\ f\ (V_z y_a + Q_z f_{a-1} + x_z) \tag{31}$$

$$u_a = log\ i\ stic(V_u y_a + Q_u f_{a-1} + x_u) \tag{32}$$

$$f_a = u_a * tan\ g\ (Z_a) \tag{33}$$

Where, V_g , Q_g , V_j , Q_j , V_z Q_z V_u Q_u represent weight matrix, x_g , x_j , x_z , x_u represent bias vector, Z_a and Z_{a-1} represent the most recent memory states. $tan\ g$ and $log\ i\ stic$are the functions that activate things. Forward LSTM and backward LSTM are combined to form BiLSTM and are mathematically expressed as

$$g_a = \vec{g}_a \oplus \tilde{g}_a \tag{34}$$

The acoustic features are extracted from this using this proposed method.

D. Video Modality Generation

The process of generating video modalities involves creating multiple video versions. The process usually requires specialized software that can automatically generate different versions of the video based on factors such as resolution, aspect ratio, and file format. Video modality generation is an important aspect of video production. It allows content creators to reach a wider audience and ensure their videos are accessible across different platforms and devices.

a. Keyframe Extraction

Current violence detection techniques use each frame extracted from the video for training. In a typical video sequence, several consecutive frames are copied. These duplicate frames increased the complexity and computational burden of the model. In this study, keyframe extraction eliminates identical consecutive frames and reduces computational costs by eliminating duplicate frames. All frames extracted from the video data are fed into the algorithm. The first setting of the video was treated as a keyframe as well as added to the list. The following frame is linked with the previous frame, and the similarity of two consecutive frames is calculated. The difference among binary frames is considered by non-zero charge using a uncertain matrix deduction. A non-zero responsibility can be compared to a threshold. The binary decision threshold consists of the upper and lower thresholds. Frames above the threshold are treated as keyframes.

b. Residual 101 Network

The Residual 101 network is a state-of-the-art technology that uses residual neural networks to achieve high levels of accuracy in image recognition tasks. This network architecture is designed to learn from surviving links, allowing complex image data to be processed effectively. Residual 101 is a deep learning model trained on large datasets, allowing it to detect a wide variety of objects with remarkable precision. It enables accurate identification of objects in real time and is an ideal solution for applications such as autonomous vehicles, security systems and medical imaging. The Residual 101 network is a powerful tool for image recognition and analysis. This architecture is based on the residual learning framework, which enables the training of very deep networks. The remaining 101 network consists of 101 layers, each performing a specific computation on the input data. This architecture has proven extremely effective for a variety of computer vision tasks, including image classification and object detection. The residual 101 network is a powerful deep learning tool. The architecture of the residual 101 network aims to address the vanishing gradient problem that can occur in very deep neural networks. The ResNet101 consists of 104 convolutional layers with 33 blocks, 29 of which are used

directly in previous blocks. The architecture is shown in Figure 6.

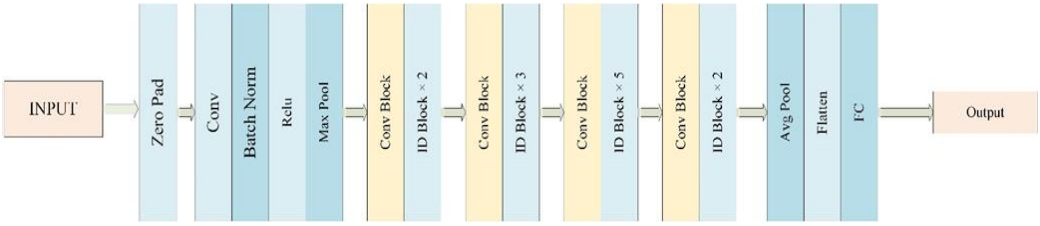


Fig.6. Architecture of ResNet 101 deep network

Consider $D_P = \{(\varphi_1{}^P, \phi_1{}^P), \ldots, (\varphi_j{}^P, \phi_j{}^P), \ldots, (\varphi_r{}^P, \phi_r{}^P)\}$ with a learning task $L_D, L_P, (\varphi_q{}^P, \phi_q{}^P) \in \Re$. Consider $D_K = \{(\varphi_1{}^K, \phi_1{}^K), \ldots, (\varphi_j{}^K, \phi_j{}^K), \ldots, (\varphi_q{}^K, \phi_q{}^K)\}$ with a learning task $L_K, (\varphi_r{}^P, \phi_r{}^P) \in \Re$ where (r, q) represent the training data sizes and $q << m$ as well as $\phi_1{}^D and \phi_1{}^K$ represent the labels in the training data. Then the Transfer learning (TL) is denoted as;

$$D_P \neq D_K, L_D = L_K \tag{35}$$

Therefore, the video data features are extracted using the proposed technique. Then, the extracted features are given to the sigmoid layer.

3.3. Fusion using Deep-tensor Fusion Framework

To improve the accuracy of the final regression estimation, this study uses multi-tensors over a network architecture framework to maintain unique dimensional semantic connections between modalities as well as interaction features. These cross-modal features of each group must simultaneously support the four modalities of text, image, audio, and video to create dynamic relationship modeling within the tensor space. Instead of dividing the modalities into four groups, this framework chose (TextA, ImageA, AudioA, VideoA) as one group and (TextB, ImageB, AudioB, VideoB) as the other group. In addition, by integrating feature vectors from multiple modalities, a multi-tensor fusion network can be used for each tensor.

$$Tensor_{yj}(C^x, C^y) = \begin{bmatrix} C^x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} C^y \\ 1 \end{bmatrix} \tag{36}$$

$$Tensor_{rt}(C^x, C^y, C^z) = \begin{bmatrix} C^x \\ 1 \end{bmatrix} \otimes \begin{bmatrix} C^y \\ 1 \end{bmatrix} \otimes \begin{bmatrix} C^z \\ 1 \end{bmatrix} \tag{37}$$

Figure 7 shows the overview of the Deep-tensor fusion framework.

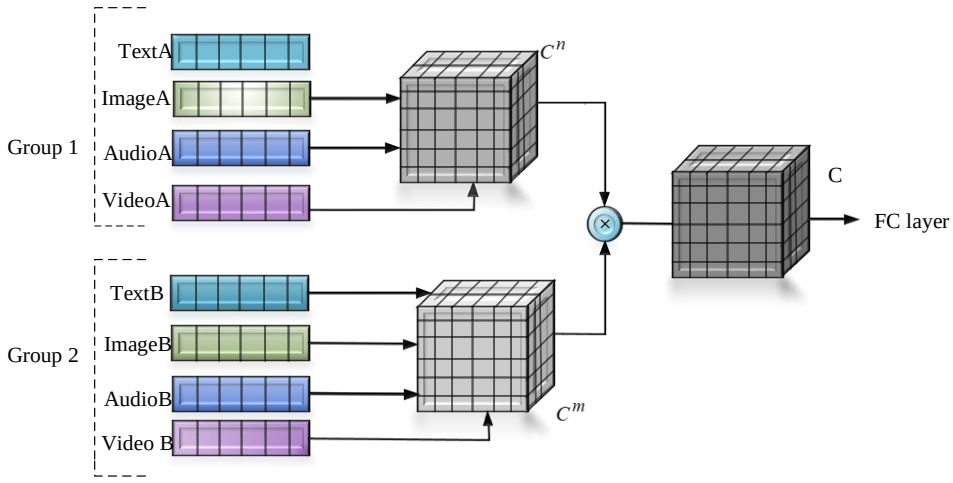


Fig.7. Deep-tensor fusion framework

If bimodal or multi-modal patterns may be achieved through tensor fusion, $\otimes$ denotes the outermost product of the vectors. The shared feature space is often invariant, and a multi-tensor fusing network is easy to integrate. The network framework can transmit feature data from one model to the next and completely capture the data through the modalities between bimodal or multi-modal modes. To ensure every modality in the final stage, the *j*vector must be patched into its feature vector during multi-level tensor fusion.

$$C = Tensor_{yj}(C^n, C^m) \tag{38}$$

$$C^n = Tensor_{rt}(\text{TextA, ImageA, AudioA, VideoA}) \tag{39}$$

$$C^m = Tensor_{rt}(\text{TextB, ImageB, AudioB, VideoB}) \tag{40}$$

Both compressed tensors can be fused to achieve a hierarchical fusion for a new perspective tensor. A tensor with spatial relationships is available through multi-level tensor fusion using high-dimensional relationships. The normalization function used by the modelling layer, $sigmoid$makes it easier to manage the output range.

$$J = 6 * sigmoid\big(ED(C; V_d)\big) - 3 \tag{41}$$

Here V_d is the weight, C is the entire tensor, and J is the predicted result of effect intensity. A fully linked neural network called ED is utilized with weights V_d that are dependent on the whole tensor$C$. These multi-tensor fusion networks feature a three-peak modality combined through model training as part of cross-modal modelling. Therefore, the extracted features in the text, image, audio and video data are merged with this proposed technique.

After the fusion method, extracted features are fed into the sigmoid layer. This layer is used to determine whether or not the input data represents cyberbullying. After that, degrees of severity are recognized. A notification is sent to the user if the error is very serious. Thus, the proposed method effectively detects cyberbullying.

4. Results and Discussion

This segment contains results as well as analysis of the proposed method, and the Python tool is used for simulation purposes. The effectiveness of the proposed method will be verified by relating the results obtained to other existing methods, such as Convolutional Neural Network (CNN), Genetic Algorithm (GA) and Gradient-Weighted Class Activation Mapping (Grad-CAM) [26]. The proposed technique is examined using various performance metrics.

4.1. Dataset Description

The proposed work includes text, image, audio and video data for cyberbullying detection.

Text:https://www.kaggle.com/datasets/andrewmvd/cyberbullyingclassification?select=cyberbullying_tweets.csv; Image: Dataset from online resources, Audio: Dataset from [27]. Video: Dataset from https://github.com/CMU-MultiComp-Lab/CMU-MultimodalSDK.

4.2. Evaluation Metrics

The performance metrics contained in the proposed methods are recall, F_1Score and accuracy. Table 1 represents evaluation metrics.

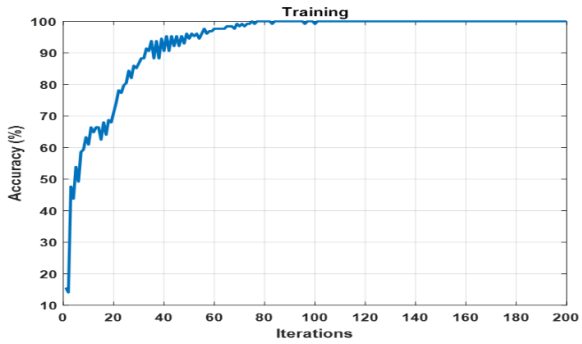
Table 1. Evaluation metrics

Recall	$R = \frac{A}{A+B}$
F₁ Score	$F_1 Score = 2 \times \frac{R \times P}{R+P}$
Accuracy	$Accuracy = \frac{(A+D)}{(A+D+B+C)}$
Specificity	$Specificity = (FPR = B/(B+D))$
Sensitivity	$Sensitivity = (TPR = A/(A+C))$

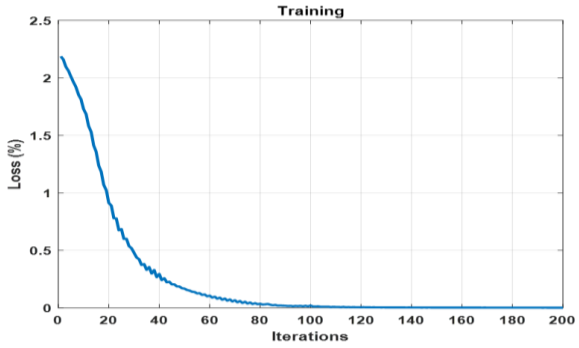
Here, Aindicates the True Positive value, D -True Negative value, B -False Positive value, C -False Negative value, R -Recall, P -Precision, FPR -False Positive Fraction, TPR -True Positive Fraction.

4.3. Performance Evaluation and Comparison Analysis

This segment contains simulation result of the proposed method. The proposed methods are compared with other existing methods like Convolutional neural network (CNN), Genetic algorithm (GA), Gradient-weighted Class Activation Mapping (Grad-CAM). Figure 8 shows the training accuracy and loss arcs of the proposed method.



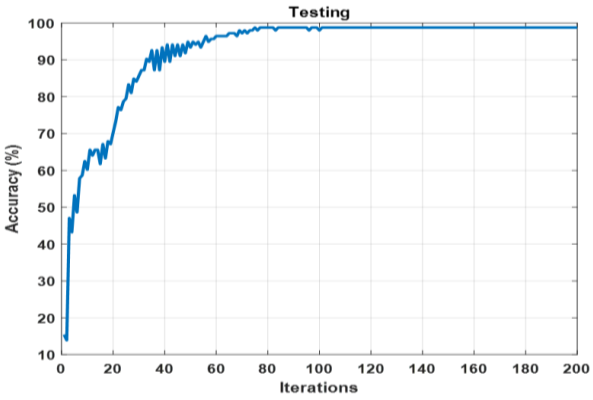
a) Training accuracy



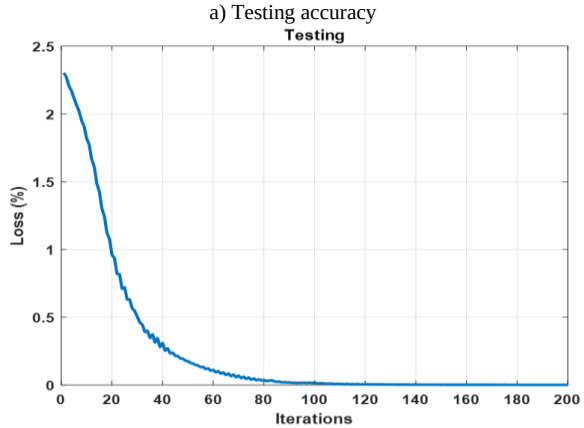
b) Training loss

Fig.8. Training accuracy as well as loss

The above figure explains the accuracy as well as loss arcs of the proposed work by changing the iterations. Figure 8 (a) shows that accuracy starts growing when the iteration is zero. When the iteration is 90, the accuracy touches its top and remains fixed for the succeeding iterations. Then, for Figure 8 (b), it is clear that the loss value starts reducing when the iteration is zero, and then the loss value gets low when the iteration is 90. The proposed method shows a low loss value for higher iterations. Figure 9 shows the testing accuracy as well as loss of the proposed method.



a) Testing accuracy



b) Testing loss

Fig.9. Testing accuracy and loss

Figure 9 (a) shows that when the iteration is 0, the accuracy starts increasing and peaks at 90. After this, the accuracy remains the same for subsequent iterations. The loss value can also be calculated by adjusting iterations in Figure 9 (b). At zero iteration, the loss value decreases, reaching its lowest at 90. It was believed that higher iterations would increase accuracy and minimize loss. Figure 10 shows the proposed method recall compared with existing methods.

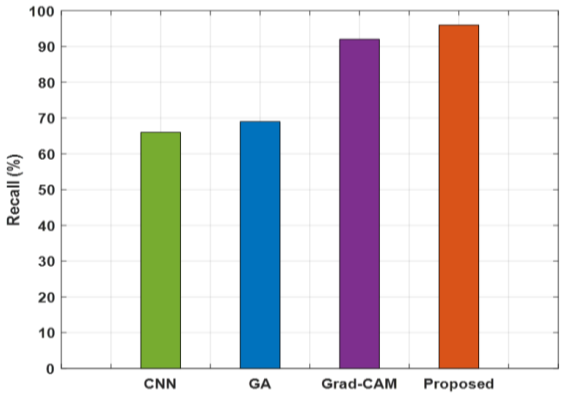


Fig.10. Recall the evaluation of the proposed method by the existing method

The above figure represents the recall assessment of the proposed method by existing methods like CNN, GA, and Grad-CAM. From this figure, the recall value of the proposed method is 96%, the value of CNN is 66%, the GA recall value is 69%, and Grad-CAM attains 92% of the recall value. Therefore, it is proved that the proposed method takes a advanced recall value comparing to existing methods. Figure 11 compares the F_1 Score for the bullying class of the proposed method with the existing method.

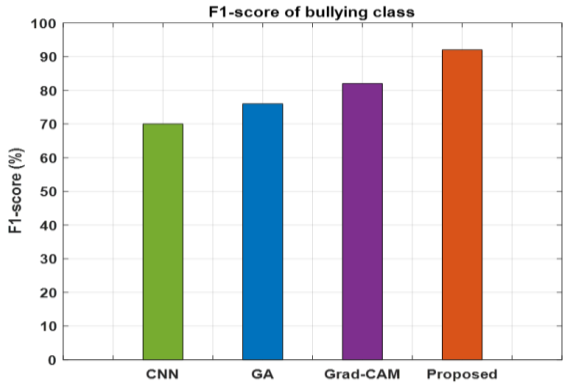


Fig.11. F_1 Score bullying comparison of the proposed method by existing method

The above figure shows the F_1Score for bullying comparison of the proposed method by existing methods like CNN, GA, and Grad-CAM. From this figure, the F_1Score value of the proposed method is 92%, the range of CNN is 70%, the GA is 76%, and Grad-CAM attains 82% of the F_1 Score value of bullying. Hence, comparing the F1Score value of bullying for the proposed method is higher than the existing techniques. Figure 12 represents the comparison of the weighted F_1Score of the proposed method with existing techniques.

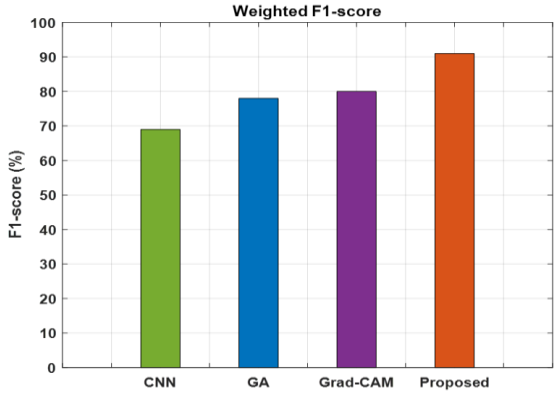


Fig.12. Weighted F_1Score comparison of the proposed with the existing method

The above figure represents the Weighted F1Score evaluation of the proposed method with existing methods like CNN, GA, and Grad-CAM. From this figure, the Weighted F1Score value of the proposed method is 91%, the value of CNN is 69%, the GA is 78%, and Grad-CAM attains 80% of the weighted F1 Score value. Therefore, comparing with the weighted F1Score value of the proposed method is higher than the existing methods. Figure 13 represents the ROC curve designed for the proposed method.

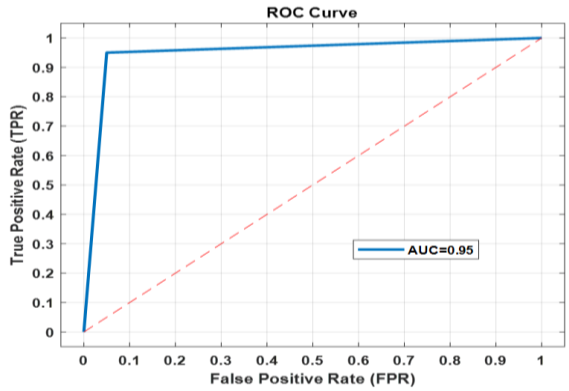


Fig.13. ROC curve for the proposed method

The figure above shows the ROC curve for the proposed method. This figure shows that when the false positive rate is zero, the true positive rate is also zero. Increasing the value of the false positive rate then increases the AUC value corresponding to the value of the true positive rate. If the false positive rate was bigger than zero as well as the true positive rate value is greater than 0.9, the AUC value remains at 0.95. Figure 14 represents the execution time of the proposed method.

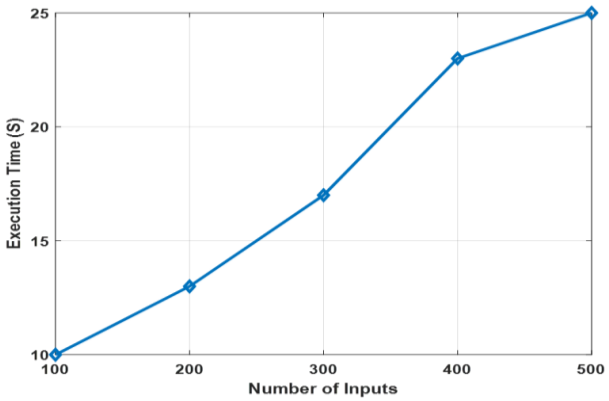


Fig.14. Proposed execution time (s)

The above figure illustrates the execution time (s) vs. the number of inputs. Changes in the number of inputs are used to measure the execution time. When the number of inputs is 100, the execution time is 10s, and when the input number is 200, the execution time is 13s. Also, for the number of inputs as 400, the execution time is 23s. Finally, when the input number is 500, it reaches 25s execution time. Here, when the number of inputs is increasing, the proposed execution time also increases.

5. Conclusions

In this work, a multi-model cyberbullying detection in social media is performed by using a novel automated deep tensor fusion framework. Here, the four different inputs like text, image, audio and video collected from the publicly available online sources. Initially, the text data is pre-processed by tokenization, stemming, lemmatization, spell correction, and PoS tagging methods. Then, for enhancing the ability of detection model, the most important features are extracted by using Log Term Frequency-based Modified Inverse Class Frequency (LTF-MICF), glove modelling and Convolutional Dense Capsule Network (Conv_DCapNet) model. Then, the image inputs are pre-processed using Gabor filtering and image resizing. Once pre-processing is done, features are extracted using a novel Self-attention based bidirectional gated recurrent auto-encoder network (SA_BiGR_AENet). On the other hand, features from audio data are extracted using Librosa library and Attentional convolutional BiLSTM. Next, for converting the video into frames, keyframe extraction is takes place and Residual 101 network is employed for extracting the audio features. Finally, all the extracted features are fused and the detection is done by proposing a new Deep-tensor fusion framework in which the

sigmoid layers are used for cyberbullying classification. For simulation, the proposed study used Python tool and the proposed technique show better results in terms of recall at 96%, F_1Score for bullying at 92%, weighted F_1Score at 91%, and the proposed AUC value is 0.95. Because of utilizing effective methods for processing each modality, the proposed study attains superior outcomes. The comparison analysis over other existing methods justified the strength of proposed work. As for future work, cyberbullying detection is intended to be tried in Indian regional languages. Also, Explainable Artificial Intelligence (XAI) will be employed in the future for understanding the models more precisely.

Reference

[1] Kumar, Akshi, and Nitin Sachdeva. "Multimodal cyberbullying detection using capsule network with dynamic routing and deep convolutional neural network." *Multimedia Systems* (2021): 1-10.

[2] Maity, Krishanu, Abhishek Kumar, and Sriparna Saha. "A Multitask Multimodal Framework for Sentiment and Emotion-Aided Cyberbullying Detection." *IEEE Internet Computing* 26, no. 4 (2022): 68-78.

[3] Yi, Peiling, and Arkaitz Zubiaga. "Session-based cyberbullying detection in social media: A survey." *Online Social Networks and Media* 36 (2023): 100250.

[4] Maity, Krishanu, Sriparna Saha, and Pushpak Bhattacharyya. "Emoji, sentiment and emotion aided cyberbullying detection in hinglish." *IEEE Transactions on Computational Social Systems* (2022).

[5] Suhas Bharadwaj, R., S. Kuzhalvaimozhi, and N. Vedavathi. "A Novel Multimodal Hybrid Classifier Based Cyberbullying Detection for Social Media Platform." In *Proceedings of the Computational Methods in Systems and Software*, pp. 689-699. Cham: Springer International Publishing, 2022.

[6] Lin, Hong, Patrick Siarry, H. L. Gururaj, Joel Rodrigues, and Deepak Kumar Jain. "Special issue on deep learning methods for cyberbullying detection in multimodal social data." *Multimedia Systems* 28, no. 6 (2022): 1873-1875.

[7] Kim, Seunghyun, Afsaneh Razi, Gianluca Stringhini, Pamela J. Wisniewski, and Munmun De Choudhury. "A human-centered systematic literature review of cyberbullying detection algorithms." *Proceedings of the ACM on Human-Computer Interaction* 5, no. CSCW2 (2021): 1-34.

[8] Roy, Pradeep Kumar, and Fenish Umeshbhai Mali. "Cyberbullying detection using deep transfer learning." *Complex & Intelligent Systems* 8, no. 6 (2022): 5449-5467.

[9] Alam, Firoj, Stefano Cresci, Tanmoy Chakraborty, Fabrizio Silvestri, Dimiter Dimitrov, Giovanni Da San Martino, Shaden Shaar, Hamed Firooz, and Preslav Nakov. "A survey on multimodal disinformation detection." *arXiv preprint arXiv:2103.12541* (2021).

[10] Zhao, Zehua, Min Gao, Fengji Luo, Yi Zhang, and Qingyu Xiong. "LSHWE: improving similarity-based word embedding with locality sensitive hashing for cyberbullying detection." In *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8. IEEE, 2020.

[11] Kiela, Douwe, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. "The hateful memes challenge: Detecting hate speech in multimodal memes." *Advances in neural information processing systems* 33 (2020): 2611-2624.

[12] Arif, Muhammad. "A systematic review of machine learning algorithms in cyberbullying detection: future directions and challenges." *Journal of Information Security and Cybercrimes Research* 4, no. 1 (2021): 01-26.

[13] Suryawanshi, Shardul, Bharathi Raja Chakravarthi, Mihael Arcan, and Paul Buitelaar. "Multimodal meme dataset (MultiOFF) for identifying offensive content in image and text." In *Proceedings of the second workshop on trolling, aggression and cyberbullying*, pp. 32-41. 2020.

[14] Woo, Wai Hong, Hui Na Chua, and May Fen Gan. "Cyberbullying Conceptualization, Characterization and Detection in Social Media–A Systematic Literature Review." *International Journal on Perceptive and Cognitive Computing* 9, no. 1 (2023): 101-121.

[15] Rezvani, Nabi, and Amin Beheshti. "Towards attention-based context-boosted cyberbullying detection in social media." *J. Data Intell* 2 (2021): 418-433.

[16] Gomez, Raul, Jaume Gibert, Lluis Gomez, and Dimosthenis Karatzas. "Exploring hate speech detection in multimodal publications." In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1470-1478. 2020.

[17] Wu, Fan, Bin Gao, Xiaoou Pan, Zelong Su, Yu Ji, Shutian Liu, and Zhengjun Liu. "FACapsnet: A fusion capsule network with congruent attention for cyberbullying detection." *Neurocomputing* 542 (2023): 126253.

[18] Bharti, Shubham, Arun Kumar Yadav, Mohit Kumar, and Divakar Yadav. "Cyberbullying detection from tweets using deep learning." *Kybernetes* 51, no. 9 (2021): 2695-2711.

[19] Bhowmick, Rajat Subhra, Isha Ganguli, Jayanta Paul, and Jaya Sil. "A multimodal deep framework for derogatory social media post identification of a recognized person." *Transactions on Asian and Low-Resource Language Information Processing* 21, no. 1 (2021): 1-19.

[20] Pramanick, Shraman, Shivam Sharma, Dimitar Dimitrov, Md Shad Akhtar, Preslav Nakov, and Tanmoy Chakraborty. "MOMENTA: A multimodal framework for detecting harmful memes and their targets." *arXiv preprint arXiv:2109.05184* (2021).

[21] Kumari, Kirti, and Jyoti Prakash Singh. "Multi-modal cyber-aggression detection with feature optimization by firefly algorithm." *Multimedia systems* 28, no. 6 (2022): 1951-1962.

[22] Al-Harigy, Lulwah M., Hana A. Al-Nuaim, Naghmeh Moradpoor, and Zhiyuan Tan. "Building towards automated cyberbullying detection: A comparative analysis." *Computational Intelligence and Neuroscience* 2022 (2022).

[23] Murshed, Belal Abdullah Hezam, Suresha, Jemal Abawajy, Mufeed Ahmed Naji Saif, Hudhaifa Mohammed Abdulwahab, and Fahd A. Ghanem. "FAEO-ECNN: cyberbullying detection in social media platforms using topic modelling and deep learning." *Multimedia Tools and Applications* (2023): 1-40.

[24] Maity, Krishanu, Prince Jha, Sriparna Saha, and Pushpak Bhattacharyya. "A multitask framework for sentiment, emotion and sarcasm aware cyberbullying detection from multi-modal code-mixed memes." In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1739-1749. 2022.

[25] Abishak, I. K. M., J. I. Sheeba, and D. S. Pradeep. "Unsupervised hybrid approaches for cyberbullying detection in Instagram." *International Journal of Computer Applications* 174, no. 26 (2021): 41-46.
[26] Pericherla, Subbaraju, and Ilavarasan Egambaram. "Cyberbullying detection on multi-modal data using pre-trained deep learning architectures." *Ingeniería Solidaria* 17, no. 3 (2021): 1-20.
[27] Sharon, Rini, Heet Shah, Debdoot Mukherjee, and Vikram Gupta. "Multilingual and multimodal abuse detection." *arXiv preprint arXiv:2204.02263* (2022).

Authors' Profiles

Neha Singh is a Research Scholar in Oriental University Indore, India. She is pursuing a PhD degree in Computer Science and Engineering. Her research areas are Networking, Cyber Security, and Machine Learning. She completed B.E (Information Technology), M.TECH (Computer Science & Engineering), PhD (Computer Science & Engineering). She has more than 9 years of Academic, IT Industry & Research Experience. She Published 7 Research Papers (3 paper in International Journals, 4 Paper in International Conference & and Guided 3 Graduate research projects).

Dr. Sanjay Kumar Sharma is a Professor at Oriental University Indore, India. He holds a PhD degree in Computer Science and Engineering. His research areas are Network Security, Cloud Computing, data mining, machine learning. He completed B.E (Computer Science & Engineering), M.TECH (Computer Science & Engineering), PhD (Computer Science & Engineering). He has more than 18 years of Academic, Administrative & Research Experience. He has a Membership in CSI (Computer Society of India), ACM (Association for Computing Machinery). He published more than 45 Research Papers (More than 32 paper in International Journals, 10 Paper in International Conference & 03 in National Conference and Guided 30 Post Graduate research projects).

How to cite this paper: Neha Minder Singh, Sanjay Kumar Sharma, "Multi-modal Cyberbullying Detection with Severity Analysis Using Deep-Tensor Fusion Framework", International Journal of Computer Network and Information Security(IJCNIS), Vol.17, No.3, pp.144-160, 2025. DOI:10.5815/ijcnis.2025.03.08