Received: 24 August 2024 | Revised: 16 December 2024 | Accepted: 25 February 2025 | Published online: 31 March 2025

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2025, Vol. 4(4) 451–465
DOI: 10.47852/bonviewJCCE52024162

BON VIEW PUBLISHING

Deep Learning Approaches for Detecting Cyberbullying on Social Media

Ghaith Jaradat[1], Mohammad Shehab[1,*], Dyala Ibrahim[1], Saif Najdawi[1] and Rami Sihwail[1]

[1]*College of Information Technology, Amman Arab University, Jordan*

Abstract: The widespread use of social media has brought many challenges, mainly due to a misconstrued interpretation of the right to freedom of expression. Cyberbullying is a particularly noteworthy issue with far-reaching global implications for both its victims and the wider community. It takes the form of bullying that happens on several social media websites. This paper's goal is to develop a deep learning model capable of recognizing cases of cyberbullying on social media. Four models, such as bidirectional long short-term memory (BiLSTM), convolutional neural network with bidirectional long short-term memory (CNN-BiLSTM), bidirectional long short-term memory with gated recurrent unit (BiLSTM-GRU), and artificial neural network (ANN), will be evaluated in a multiclass classification difficulty context. The results showed that the BiLSTM model outperformed the other models by achieving the highest accuracy in 91% of cases, while the CNN-BiLSTM and ANN models demonstrated relatively lower performance. In addition to determining the efficacy of the deep learning techniques, the work highlights the urgent requirement for strong systems to resist cyberbullying. By enhancing detection accuracy, the proposed model can contribute significantly to providing a safer digital environment for further studies in this field.

Keywords: cyberbullying, social media, tweets, machine learning, deep learning, classification

1. Introduction

Social media has become part and parcel of our lives, providing continuous communication and entertainment. It, however, poses some unique challenges, one of which is antisocial behavior like cyberbullying, stalking, and harassment. Hostile behavior may be physical, verbal, or even social bullying. This might negatively affect the mental health of a person [1]. On the other hand, cyberbullying is bullying on computers, cell phones, and tablets, usually on message boards or online discussion forums, and is targeted toward young people who happen to be weak due to immature life experiences and evolving social skills [2]. It may even result in a lack of self-esteem, thoughts of suicide, and emotional reactions like anger, anxiety, frustration, and despair among the victims. In the modern world, the lack of parental vigilance over children's misbehaving might result in cyberbullying [3].

This can include emailing or posting rumors, uploading embarrassing videos or photos, and posting derogatory or threatening messages. Since cyberbullying does not require one to make direct contact with the target of bullying, it has become widespread [4]. Therefore, efficient detection and prevention methods must be developed to combat cyberbullying. A machine learning (ML) algorithm is beneficial in learning the trends to detect cases of cyberbullying more effectively and efficiently than when being monitored by human beings [5]. The detection and prevention of cyberbullying is difficult because of its nature of occurrence in digital form [6]. The need for early detection of such an act stems from the need to provide instant prevention and aid to the victim. Traditional methodologies like manual monitoring are ineffective because of the hugeness of the volumes of data on social media. Primarily, the research focused on developing patterns for ML models capable of cyberbullying detection in online content [7, 8]. An explosive increase in the use of social media has brought about a pervasive problem of cyberbullying that profoundly affects the mental and emotional state of individuals. The usual targets of this offense are young individuals who repeatedly undergo emotional suffering, social isolation, and physical harm. Their vulnerability to cyberbullying is high because they have not experienced much in life, and their social skills are still growing.

The main challenge in handling the issue of cyberbullying is basically because of the nature of cyberbullying being online, therefore posing a problem in terms of detection and prevention [9, 10]. In other words, there must be more emphasis on the need for real-time cyberbullying detection, enabling effective remedial intervention and support to be meted out in time. Generally, the conventional way to detect cyberbullying includes tiresome and ineffective manual monitoring. The methods need to be revised with the massive volume of data about social media and their rapid interaction. Automated mechanisms that efficiently and effectively analyze remarkable quantities of data on social media are urgently required to detect cases of cyberbullying in real time.

This work proposes a new detection system to deal with the current problem. The approach taken by ML algorithms is that it must be possible to build a model that can learn from data to predict specific outcomes or, rather, to make a decision without modeling the process to accomplish this task and carry out the task itself. Such models should be trained to extract the usual manifestation patterns of cyberbullying from live web content, resulting in an approach that

***Corresponding author:** Mohammad Shehab, College of Information Technology, Amman Arab University, Jordan. Email: m.shehab@aau.edu.jo

is more scalable, efficient, and immediate in thwarting the offense of cyberbullying. The primary aim is to implement ML models that can detect potential cyberbullying incidents effectively and notify concerned authorities or platforms for immediate intervention, thereby minimizing the harm caused to individuals.

The remaining part of this paper has been organized as follows: Section 2 lists the related work, Section 3 describes the proposed methodology, Section 4 outlines the experimental results, and finally, Section 5 concludes the paper.

2. Literature Review

Cheng et al. [11] proposed a system for the detection of cyberbullying. They proposed proof of bias in models trained using sessions gathered from different social media, like Instagram. Their proposed strategy applied a reinforcement learning approach that would ensure the debiasing of the model in a context-aware and model-agnostic way. This strategy did not need any extra resources or annotations except for a predefined set of sensitive triggers usually used for recognizing instances of cyberbullying. Empirical evaluations have demonstrated that the proposed strategy is very effective in mitigating the effects of unintended bias while enhancing detection performance. However, this system needs to decrease time consumption.

Kumar and Sachdeva [12] proposed a complex neural network architecture for detecting cyberbullying over three kinds of social data: text, images, and infographics, which refers to text combined with images. The CapsNet–ConvNet architecture is an end-to-end design that merges the capsule network deep neural network, CapsNet, for dynamic routing to predict textual bullying content with the convolution neural network, and ConvNet, for predicting visual bullying content. The information is segregated inside the graphics using Google Lens as a feature of the Google Photos app to separate text from the image. Additionally, a perceptron-based decision-level late fusion strategy is used in multimodal learning to dynamically fuse predictions from discrete modalities and generate the final classification of bullying or non-bullying. The experimental evaluation of the proposed model was conducted on a multimodal dataset of 10,000 comments and posts extracted from YouTube, Instagram, and Twitter. Results highlight excellent performance with an achieved area under the receiver operating characteristic curve (AUC-ROC) of 0.98. However, the system needs to improve performance to detect malicious attacks.

Hooda et al. [13] presented a detection system of offensive comments for textual data using ML. Social media is vital to contemporary life, enabling global connectivity and facilitating communication with individuals worldwide. Social media allows for disseminating and propagating information, thoughts, and ideas. However, similar to any other technological advancement, it impacts individuals, either positively or negatively. They have transformed into a medium for disseminating animosity, derogatory remarks, and online harassment. Cyberbullying encompasses the phenomenon of bullying conducted via digital mediums, including but not limited to social media platforms, messaging applications, online gaming platforms, and mobile communication devices. The detection system aims to identify offensive comments within textual data. The dataset, comprising 35,000 comments, has been sourced from Kaggle. A comprehensive evaluation was conducted using various ML methods, and among them, the support vector machine demonstrated superior performance with an accuracy (ACC) of 92.2%, but the system needs to decrease the time consumption.

Certain adolescents engage in cyberbullying, which refers to a consistent pattern of online harassment toward others. A significant number of adolescents lack awareness regarding the potential hazards associated with cyberbullying, such as the increased likelihood of experiencing depression, engaging in self-harming behaviors, and even contemplating suicide. Due to its detrimental impact on an individual's psychological well-being, cyberbullying is a significant issue that requires attention. Obaid et al. [14] aimed to create a method for determining the extent of bullying using a deep learning algorithm and fuzzy logic. This task aimed to process and analyze Twitter data (consisting of 47,733 comments) obtained from Kaggle to identify and flag comments related to cyberbullying. The Keras comments were inputted into an extended short-term memory network consisting of four layers for classification. Subsequently, fuzzy logic was employed to ascertain the magnitude of the comments. The experimental findings indicate that the proposed framework offers an effective solution for detecting bullying, with ACC, $F1$-score ($F1$-S), and recall (REC) values of 93.67%, 93.64%, and 93.62%, respectively. However, the system needs to decrease the time consumption for detection.

The proposed models build upon foundational models (Bidirectional LSTM (BiLSTM), CNN-BiLSTM, BiLSTM-GRU, and artificial neural network (ANN)) by introducing architectural modifications that enhance performance in cyberbullying detection. Key enhancements include global max pooling to reduce dimensionality, dropout regularization to prevent overfitting, and the combination of BiLSTM with gated recurrent unit (GRU) to optimize efficiency and computational speed. The BiLSTM model exhibited the best performance, achieving a classification ACC of 91%, followed by the BiLSTM-GRU model with an ACC of 90%, ANN with 89%, and CNN-BiLSTM with 87%. These enhancements position the proposed models as robust solutions for detecting cyberbullying in social media environments, enabling safer online interactions [15–17].

3. Proposed Methodology

In this section, the flowchart for the proposed model will be presented, as depicted in Figure 1. The steps for the proposed model will be explained in the next subsections.

The dataset utilized in this paper is the cyberbullying dataset, accessible in a CSV file titled "cyberbullying_tweets.csv." The dataset contains 47,692 rows and two columns. The first column contains the text of the tweets, while the second column contains the label that indicates the specific type of cyberbullying present in each tweet. The "cyberbullying type" column contains six distinct values: "not cyberbullying," "gender," "religion," "other cyberbullying," "age," and "ethnicity." These labels denote the diverse forms of cyberbullying that can be found in the tweets [17]. The dataset is depicted in Table 1.

3.1. Data preprocessing

The preparation of data must, therefore, be preprocessed in order to use it optimally for the training and evaluation of deep learning models. In particular, the steps include removing stop words, stemming, converting to lowercase, removing punctuation and special characters, and lastly, tokenization. These procedures would fulfill the process of cleaning and standardizing the text data, hence helping the deep learning models. Preprocessing is crucial in building an effective model to be employed in the detection of instances of cyberbullying occurring on social media platforms. Text preprocessing can be applied to the "tweet_text" column to ensure better model ACC. These entail methods like tokenization, in which the text is disintegrated into instances of individual words. Tokenization will also remove stop words, apply them to stems, make all text lowercase for consistency, and take several other steps. In the same

Figure 1
The flowchart for the proposed model

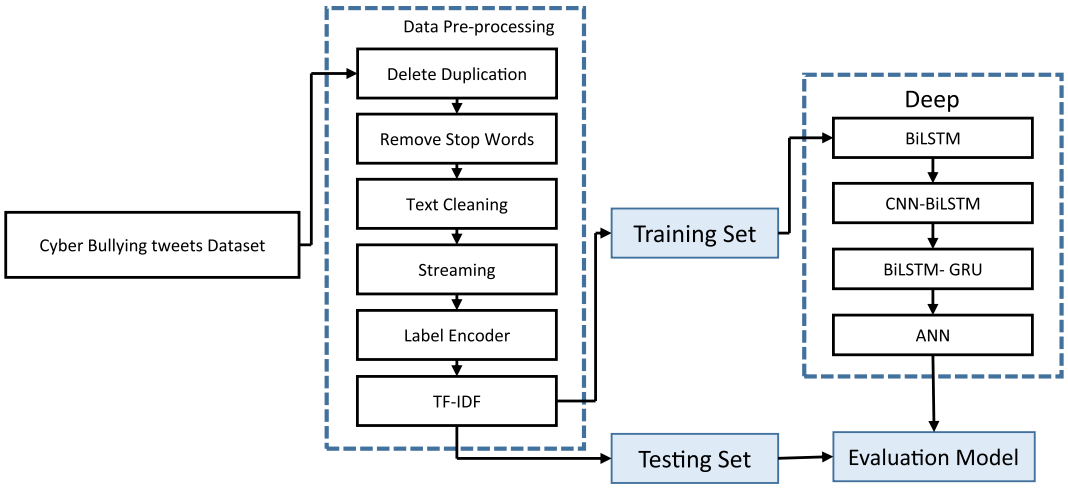


Table 1
The dataset description

No.	Tweet_text	Cyberbullying_type
0	In other words #katandandre, your food was cra. . .	not_cyberbullying
1	Why is #aussietv so white? #MKR #theblock #ImA. . .	not_cyberbullying
2	@XochitlSuckkks a classy whore? Or more red ve. . .	not_cyberbullying
3	@Jason_Gio meh. :P thanks for the heads up, b. . .	not_cyberbullying
4	@RudhoeEnglish This is an ISIS account pretend. . .	not_cyberbullying

fashion, missing or invalid data can be handled by removing such occurrences or substituting the same with passed-in placeholder values. Data preprocessing contains many techniques.

1) Drop duplication

Eliminating duplicates during data preprocessing involves identifying and removing rows that have identical values across all columns. Such duplicates may arise from errors during data entry, merging data from various sources, or technical issues in data collection. Removing these redundant entries is crucial to maintain data ACC, consistency, and reliability. Duplicates can distort the performance of deep learning models, resulting in inaccurate predictions and conclusions. Additionally, redundant data can unnecessarily inflate the dataset's size, complicating processing and analysis. Various techniques can be applied to remove duplicates, depending on the dataset's characteristics [18]. A commonly used approach is the "drop_duplicates" function in the Pandas library, which efficiently removes duplicate rows, ensuring that only unique entries remain.

2) Remove stop words

A vital portion of the information preprocessing handled in our show is the evacuation of halt words, which includes sifting out commonly utilized words that do not contribute noteworthy meaning to the dataset. These halt words, such as "the," "and," "a," "an," "in," "to," and "of," are habitually found in sentences but regularly do not include much semantic esteem. By prohibiting these words, ready to altogether diminish the dataset's estimate, driving to a speedier examination. Dispensing with halt words permits us to center on the key terms that pass on the central message of the text, making a difference in revealing basic designs and patterns. The method starts by compiling a list of words that are commonly utilized in the English dialect. We, at that point, apply this list to expel all occasions of these halt words from the "tweet_text" column in our dataset. This errand can be productively finished utilizing different normal dialect handling (natural language processing (NLP)) libraries, such as the Common Dialect Toolkit (NLTK) [19].

3) Text cleaning

Content cleaning is done to make consumable content that kills excess characters, images, and words. The step moves forward with PRE and makes the content more standard. A few methods utilized in cleaning content incorporate evacuating joins, notices, non-American Standard Code for Information Interchange (ASCII) characters, non-alphanumeric characters, and newline or carriage return characters. Evacuating joins and mentions from content is ordinarily tiring in most NLP ventures. These are unessential to the substance of the content and can add commotion. The joins and notices can be expelled to make the content information cleaner and more reasonable for investigation by finding URLs within the content and supplanting them with purge strings. Non-ASCII characters incorporate highlighted letters, remote characters, and emoticons that will cause issues amid the content examination. All non-alphanumeric characters like accentuation marks, images, and non-letters can be evacuated to put the content into a standard shape. All these characters can be supplanted by a space character, which may make the content basic and simple to handle. The newline and carriage return characters can be excluded to uniform the content and handle it more effectively. The final common information preprocessing procedure is removing a particular category or lesson from the dataset. Consequently, this process includes the determination of columns or occasions belonging to a place in the desired course and removing them from a given dataset to create a clearer or of superior quality.

4) Lemmatization

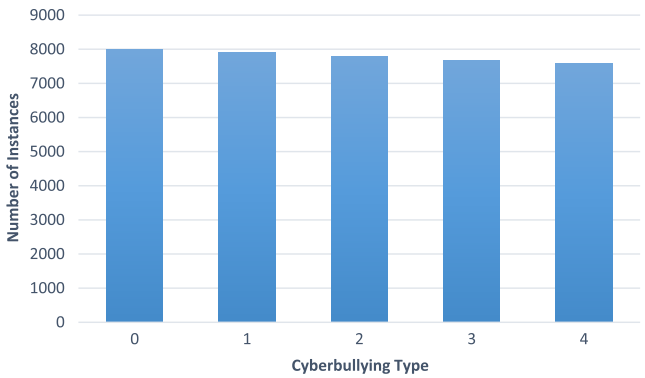
One technique in NLP is lemmatization, which simplifies words to their base or root form, known as the lemma. This process involves recognizing the root of a word and converting it into a

standardized form, enabling more effective analysis of textual data. The lemma serves as the standardized version of a word and is useful for examining how words are used in various contexts. Lemmatization also involves identifying the grammatical role of a word, such as whether it's a noun, verb, or adjective, and applying morphological rules to derive its lemma, for example, "walking" is reduced to "walk" and "better" to "good." By converting words to their root forms, lemmatization helps uncover patterns or connections in text data that might otherwise go unnoticed [20].

5) Label encoder

This is another deep learning preprocessing technique for converting categorical data into numerical values. It uses a different numerical identifier that identifies every category of the categorical variable [21]. The "cyberbullying_type" column is a categorical variable that includes "not_cyberbullying," "gender," "religion," "age," and "ethnicity." Each category will be labeled and encoded by giving them a unique numerical label. For instance, label 0 will be for "not_cyberbullying," 1 for "gender," 2 for "religion," 3 for "age," and 4 for "ethnicity," as shown in Figure 2.

Figure 2
Dataset after label encoder



3.2. Term frequency-inverse document frequency

One of the most popular techniques in NLP to measure the importance or significance of phrases in a document or a collection of documents is the term frequency-inverse document frequency (TF-IDF) method [22]. It creates a measure that considers the frequency of a term in a document and the frequency in the whole corpus. TF-IDF considers both of the above aspects and assumes that terms frequently occurring in a document yet scarce in the whole corpus are more important and influential in explaining the document's content. The term frequency (TF) part of the TF-IDF formula is elementary and computes the number of times a term appears in a text. It is a computation indicating the ratio of the frequency of a particular expression to the total number of expressions in the document:

$$TF(term,\ document) = \frac{The\ Freq.\ of\ Occ.\ of\ Term\ with\ in\ Document}{Total\ Number\ of\ Terms\ in\ Document} \quad (1)$$

The IDF portion of the calculation is the expression detailing the term's frequency in the entire corpus of documents. To determine how important a particular phrase is, the logarithm of the ratio of the total documents in the corpus to the number of documents containing that term is used:

$$IDF(term,\ corpus) = Log_e\left(\frac{Total\ num\ of\ doc\ in\ the\ corpus}{Num\ of\ doc\ that\ contain\ the\ term}\right) \quad (2)$$

The TF-IDF score assigned to a term within a document is obtained by multiplying the respective TF and IDF values associated with that term:

$$TF - IDF(term, document,\ corpus) = TF(term,\ document)\ *IDF(term,\ corpus) \quad (3)$$

The TF-IDF scores for each term in a text or corpus can be calculated to identify the most important and relevant terms for a particular topic.

The data preprocessing stage combines standard techniques like stop word removal, tokenization, and lemmatization with problem-specific approaches tailored for cyberbullying detection. Special attention is given to handling class imbalance and label encoding, both of which are critical for accurately classifying cyberbullying categories. These methods ensure that the models are not biased toward dominant classes and can accurately predict challenging categories like ethnicity-based and religion-based bullying, improving the fairness and robustness of the detection system.

3.3. Training and validation sets

After the dataset has undergone the necessary preprocessing, we will divide it into train and validation sets in an 80:20 ratio. This validation set will be used to estimate the model's performance and tune its hyperparameters. The training set will be used to train the model. This partitioning of the data is done randomly in such a manner as to retain the class distribution of the original dataset in both the training and validation sets. Thus, we can ensure that the model is trained on a sample of the data representative of the whole, and the performance is evaluated on a sample that is also representative. In this way, we can be confident that the model is trained based on a sample representative of the data and that the performance pertains to a representative sample.

The dataset was split into 80% for training, 10% for validation, and 10% for testing. This ratio ensures sufficient data for model training, while the validation set allows for effective hyperparameter tuning. The split was performed using a stratified method to maintain the class distribution across all subsets.

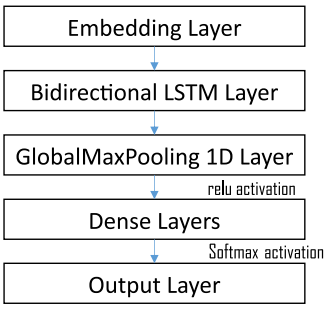
3.3.1. Deep learning models

1) Bidirectional LSTM model

The proposed BiLSTM model is structured in the architecture with a series of stacked, one-by-one, and forward layers for classifying tweets based on cyberbullying, as summarized in Figure 3. Keras's Sequential Application Programming Interface (API) implements the model that follows in the paper, which makes defining and training a model relatively easy.

The first layer of our model is the embedding layer, which will have an input size defined as one more than the length of the tokenizer word index, and the word embeddings will have a dimensionality of 128. We defined the input length as MAX_SEQUENCE_LENGTH, which we defined as the maximum length of the input sequences. The embedding layer learns what the word embedding is for the words inputted. The next layer is the BiLSTM. The LSTM model goes further to become a vital

Figure 3
BiLSTM model architecture

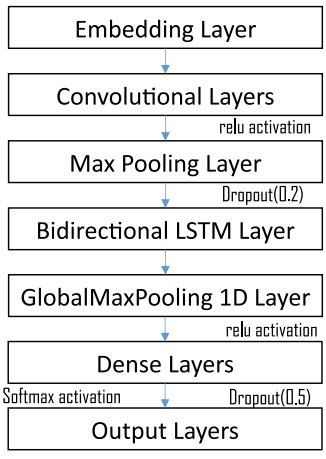


piece of design. The proposed architecture includes a BiLSTM layer consisting of 64 LSTM units set up to return sequences. Therefore, an input sequence is read from the start to the beginning and vice versa. This way, it can capture information from previous and future contexts. Then comes a GlobalMaxPooling1D layer. This layer takes the LSTM layer output sequence and reduces its dimensionality by selecting the maximum value across all the time steps. At the output, the LSTM layer makes a fixed-length vector from the sequence that contains essential features. The subsequent two layers within the model, after the GlobalMaxPooling1D layer, consist of dense layers. There are 64 units in the first dense layer, and the rectified linear unit (ReLU) activation function introduces the nonlinearity in the model to capture intricate patterns and representations. Subsequently, dropout regularization at a rate of 0.5 is applied after the initial dense layer to mitigate overfitting risks. The output layer, positioned as the fifth and final layer in the network, corresponds to the desired number of categories for tweet classification. The activation function utilized in the output layer, softmax, computes the probability of the input belonging to each class.

2) CNN-BiLSTM model

The BiLSTM model is enhanced by incorporating a CNN layer, which introduces extra-convolutional operations before the BiLSTM layer. By utilizing this combination, the model can effectively derive local and global contextual information from the input sequences, as depicted in Figure 4.

Figure 4
CNN-BiLSTM model architecture



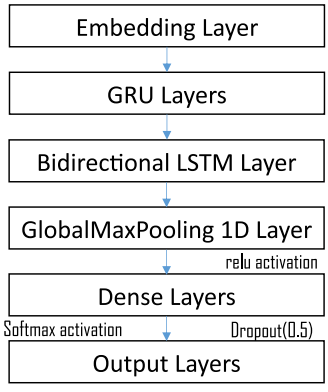
The second model's architecture commences with an embedding layer that acquires word embeddings for the input words. The input size is determined by adding 1 to the length of the tokenizer word index, and the word embeddings have a dimension of 128. The input length is defined as MAX_SEQUENCE_LENGTH, which indicates the maximum length of the input sequences. After the embedding layer, there is a Conv1D layer with 128 filters and a kernel size of 5. The convolutional layer applies convolutions to the input sequence, capturing local patterns and features by sliding a window of size 5. The model incorporates the ReLU as its activation function, which introduces nonlinearity. Following the Conv1D layer, a MaxPooling1D layer is included to perform downsampling. This is achieved by selecting the highest value within each sliding window of size 5. This process aids in decreasing the number of dimensions in the feature maps and capturing the most significant features. To mitigate overfitting, a Dropout layer is incorporated with a dropout rate of 0.2, which randomly deactivates 20% of the neurons during the training process. Following that, the BiLSTM layer is implemented, comprising 64 LSTM units and producing sequential outputs. This layer captures both long-term dependencies and contextual data from both preceding and subsequent contexts. The GlobalMaxPooling1D layer is appended following the BiLSTM layer, effectively reducing the output sequence to a constant-length vector by selecting the highest value across all time steps. This operation extracts the most significant and prominent characteristics from the LSTM layer. The model architecture incorporates two supplementary dense layers. The initial dense layer consists of 64 units and employs the ReLU activation function, thereby introducing nonlinearity. A Dropout layer with a dropout rate of 0.5 is appended to the dense layer to enhance the prevention of overfitting. The output layer comprises 5 units, corresponding to the number of classes we intend to assign the tweets to. The employed activation function is softmax, which yields the probabilities for each class.

3) BiLSTM-GRU model

The design of the third BiLSTM-GRU model is tailored for the classification of cyberbullying tweets. It makes use of the functionalities provided by BiLSTM and GRU layers, as depicted in Figure 5.

The input sequence, which represents the tweets containing cyberbullying, is initially fed into an embedding layer. The layer acquires the word embeddings for the input words by utilizing a vocabulary size equivalent to the length of the tokenizer's word index plus one and an embedding dimension of 128. It defines the input of sequence size with a defined sequence

Figure 5
BiLSTM-GRU model architecture



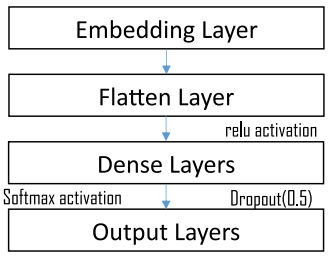
length, MAX_SEQUENCE_LENGTH. There is another sequential model with an output from the embedding layer into a BiLSTM. The BiLSTM sequence is created using 64 LSTM units, and in sequences, a GRU layer is appended after the Bidirectional LSTM.

The GRU layer uses 64 units to create sequence outputs. The paper introduces the GRUs, which are much closer to the LSTM units but are more straightforward to make. The output from the GRU layer goes into a GlobalMaxPooling1D layer, which takes the highest value across all the time steps and acts as a fixed-length representation of this sequence. Beyond GlobalMaxPooling1D are two dense layers. The initial dense layer consists of 64 units that added nonlinearity using the ReLU activation function. After this layer, a Dropout layer with a rate of 0.5 was added to compensate for the overfitting issue. The last layer, the output layer, consists of 5 units corresponding to the number of classes into which we will categorize the tweets. The implemented activation function is softmax, returning the probabilities for each class.

4) ANN model

Our ANN model architecture is designed to accurately classify any tweet as cyberbullying. Many layers in the model replicate this, set sequentially concerning processing the input data to generate predictions. As shown in Figure 6, these layers process the input information to predict the presence or absence of instances of cyberbullying in tweets.

Figure 6
ANN model architecture



The first component of this model will be an embedding layer, which forms word embeddings based on input words. The input size will be computed by adding 1 to the length of the word index formed by a tokenizer; simultaneously, the dimensionality of the word embeddings will be set to 128. The input length is defined to be MAX_SEQUENCE_LENGTH, which corresponds to the maximum allowed length of an input sequence. After the embedding layer, a Flatten layer is added to transform the multidimensional output from the previous format into a single-dimensional vector. This layer interfaces convolutional and fully connected layers. Afterward, two dense layers with full connections are added. The first dense layer has 256 units and the ReLU activation function, which adds nonlinearity to the network, letting it model complex patterns and representations. A Dropout layer is added after the first dense layer with a dropout rate of 0.5 to prevent overfitting. During training, dropout shuts off 50% of neurons randomly, forcing the model not to rely too much on some features, hence improving the generalization ability. The second dense layer of 128 units will have the ReLU as its activation function. This is followed by another Dropout layer, which has a rate of 0.5. These added layers significantly improve the model's potential to understand complex patterns but reduce the chances of overfitting. This output layer is a dense layer of five units because there are five classes in which tweets are classified. The softmax activation function is utilized, calculating the probability of input being assigned to each class. The model employs the highest probability to make predictions by selecting the corresponding class.

4. Experimental Results

In this section, the experimental results and discussion of the effectiveness of the proposed deep learning (DL) algorithms for the detection of cyberbullying activity over social media using DL techniques. Specifically, based on several performance metrics, we evaluate the performance of four DL algorithms, including BiLSTM, CNN- BiLSTM, BiLSTM-GRU, and ANN. Evaluation metrics are utilized to assess the efficacy of a deep learning model in accurately predicting the outcome of new data, particularly within the realm of cyberbullying detection on social media platforms.

1) Confusion matrix

A confusion matrix is a structured table that offers a succinct overview of the effectiveness of a deep learning classification algorithm when applied to a designated set of test data, wherein the true values are known [23]. For multiclass classification problems, where the model needs to predict among multiple classes, the confusion matrix is usually a square matrix that has one row and one column for each class [24]. The confusion matrix for multiclass classification can be depicted, as shown in Table 2.

Table 2
Confusion matrix for multiclass classification

Class	Predicted Class 1	Predicted Class 2	...	Predicted Class N
Actual Class 1	TN1	FP1	...	FP(N-1)
Actual Class 2	FP2	TN2	...	FP(N-2)
...	...	...	...	...
Actual Class N	FP(N-1)	FP(N-2)	...	TN (N)

In the context of classification evaluation, true negative (TN) signifies the number of instances accurately predicted to belong to a particular class that truly belongs to that class. False positive (FP) indicates the count of instances inaccurately predicted to belong to a class they do not actually belong to. False negative (FN) denotes the number of instances inaccurately predicted to not belong to a class despite actually belonging to it. True positive (TP) represents the count of instances accurately predicted to belong to a class they actually belong to.

2) Accuracy

ACC refers to the frequency with which the model accurately predicts the class of a given instance. The ACC is determined by dividing the number of accurate predictions by the total number of predictions and is represented as a percentage. The ACC formula is given in Equation (4) [25].

$$Acc = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{4}$$

3) Precision

Precision (PRE) is the ability of a model to accurately identify instances of cyberbullying. The cyberbullying detection rate is determined by dividing the number of correctly identified instances of cyberbullying by the total number of instances predicted as

cyberbullying and then multiplying by 100 to express it as a percentage. The equation represents PRE, as given in Equation (5) [26].

$$Pre = \frac{TP}{FP + TP} \tag{5}$$

4) Recall

Recall (REC), referred to as sensitivity or TP rate, quantifies the model's capacity to accurately detect all occurrences of cyberbullying. The cyberbullying detection rate is determined by dividing the number of correctly identified instances of cyberbullying by the total number of actual instances and is presented as a percentage. The REC formula is given in Equation (6) [27].

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

5) $F1$-score

The $F1$-S serves as a unified measure assessing the performance of a model through the computation of the harmonic mean of PRE and REC. PRE and REC are derived through a weighted average, where the weight is determined by the number of instances in each class. The formula for calculating the $F1$-S is given in Equation (7) [26].

$$F1_{Score} = 2 * \left(\frac{precision * recall}{precision + recall}\right) \tag{7}$$

4.1. Evaluation of the BiLSTM method

Figure 7 presents the loss and ACC outcomes throughout the training and validation phases of the BiLSTM model, providing insights into its performance over time.

The model is trained using the Adam optimizer and the categorical_crossentropy loss function for a total of 25 epochs. The goal is to maximize ACC while minimizing the loss. The loss value quantifies the difference between the predicted and actual values. During the training process, it was noted that the loss decreased considerably over the epochs, suggesting that the model is proficient in acquiring knowledge and enhancing its capacity to reduce errors. Likewise, the validation loss decreases, albeit at a slightly elevated level compared to the training loss. This indicates that the model exhibits strong generalization capabilities to unfamiliar data and consistently delivers reliable performance. The ACC metric quantifies the ratio of accurately classified instances. During the training process, the model notably improves ACC for both the training and validation datasets. The ACC values are high, with the training ACC surpassing 99% and the validation ACC exceeding 91%. This suggests that the model effectively acquires the patterns within the data and produces precise forecasts. The classification report of our BiLSTM model offers valuable insights into the model's performance across various categories of cyberbullying, as depicted in Table 3.

Table 3
BiLSTM classification report

Class	PRE	rec	$F1$-S	Support
0	0.80	0.80	0.80	1569
1	0.90	0.85	0.87	1568
2	0.94	0.94	0.94	1608
3	0.94	0.97	0.95	1578
4	0.97	0.98	0.98	1487
ACC	–	–	0.91	7810
macro avg	0.91	0.91	0.91	7810
weighted avg	0.91	0.91	0.91	7810

The BiLSTM model predicts cyberbullying accurately, with high PRE values for all classes. Class 4 (ethnicity) has the highest PRE (97%), followed by Class 3 (age) and Class 2 (religion). Class 1 (gender) has a good PRE (90%), and Class 0 (not_cyberbullying) has an 80% PRE. Class 4 has the highest REC (98%), followed by Class 3 (97% REC), Class 2 (94%), Class 1 (95%), and Class 0 (80% REC). Class 0 has an $F1$-S of 80%. The ACC is 91%, indicating the model's performance in predicting the correct cyberbullying type. However, the model made incorrect predictions, with 126 instances misclassified as "gender," 84 as "religion," 82 as "age," and 25 as "ethnicity."

Examining the confusion matrix enables us to evaluate the model's effectiveness in classifying various categories of cyberbullying. The model's ability to accurately predict each category can be assessed by examining its strengths and weaknesses. For example, the model demonstrates strong performance in accurately differentiating between the "ethnicity" and "age" categories, as

Figure 7
Accuracy and loss of BiLSTM

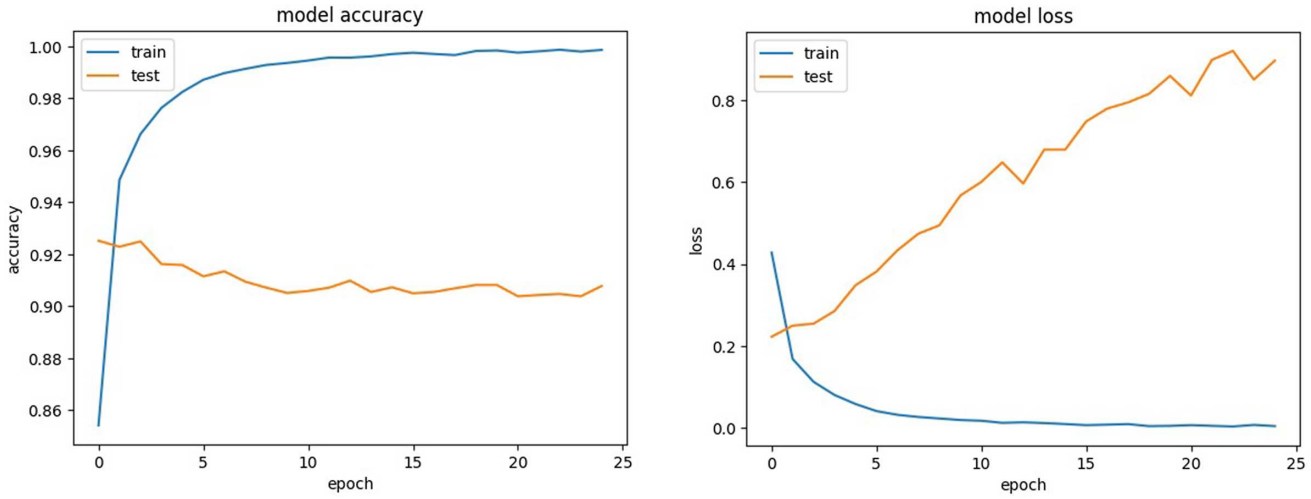
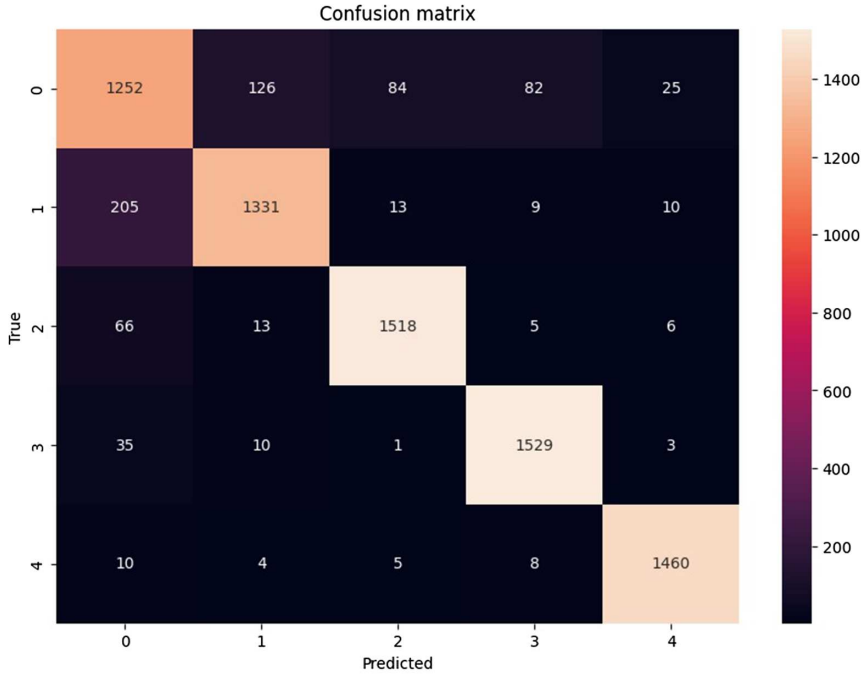


Figure 8
BiLSTM confusion matrix



evidenced by the large number of correct predictions in the corresponding diagonal cells. Nevertheless, it encounters more incredible difficulty in discerning the "not_cyberbullying" category from other forms of cyberbullying, as indicated by the elevated frequency of misclassifications in those instances, as shown in Figure 8.

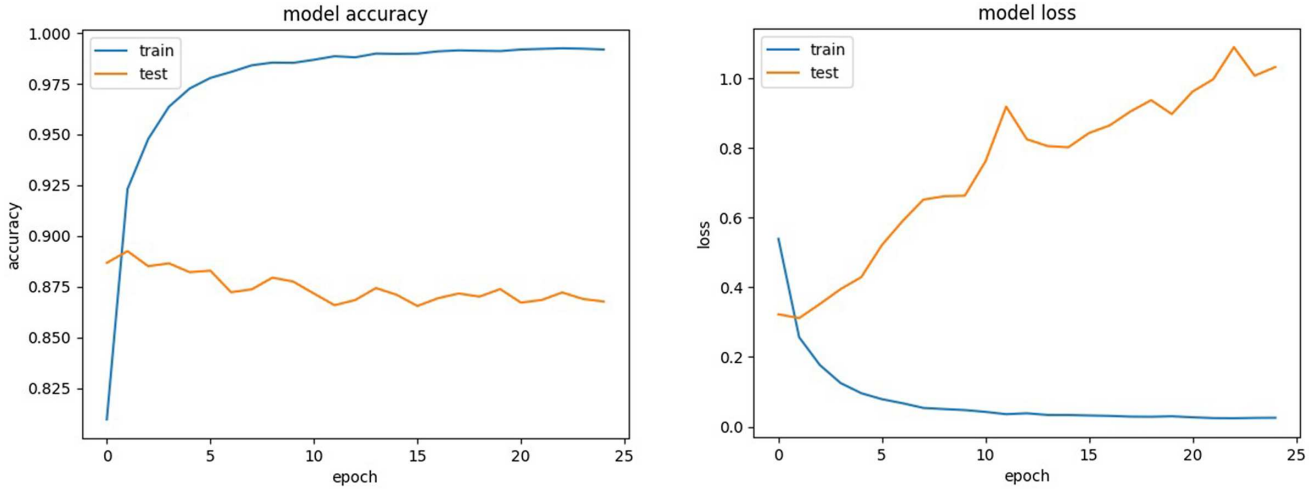
4.2. Evaluation of the CNN-BiLSTM method

During the training and validation phases of our CNN-BiLSTM model, we can track the progression of the model's performance by examining the loss and ACC metrics. The training encompasses 25 epochs, and the model's performance is evaluated on the validation set following each epoch. This process is depicted in Figure 9.

The model exhibits a favorable performance, achieving a training ACC of approximately 80.96% and a validation ACC of 88.67%. As the model undergoes further training, noticeable enhancements are observed in training and validation accuracies. At the end of the fifth training cycle or epoch, the model demonstrates an impressive training ACC of 97.25%, while the validation ACC slightly decreases but remains high at 88.21%.

The model continuously acquires new information throughout the training process and undergoes modifications. It is important to note that after a certain threshold, the training ACC increases, while the validation ACC levels off. This could indicate that the model is overfitting the training data, which may result in poor performance when applied to new data. Consequently, the model performs satisfactorily on both the training and validation sets. The efficacy of

Figure 9
Accuracy and loss of CNN-BiLSTM



the CNN-BiLSTM model was evaluated using a classification report presented in Table 4. The model's PRE for Class 0 is 75%, indicating that 75% of its predictions accurately identified this class. The model obtained a REC score of 0.71, accurately classifying 71% of the true instances belonging to this class. The *F*1-S, which combines PRE and REC, is 0.73. When assessing Class 1, the model exhibits a PRE of 0.84 and a REC of 0.82, indicating that it accurately identifies 82% of the actual Class 1 instances. The *F*1-S for this class is 0.83. Class 2 exhibits a PRE rate of 0.89, indicating that 89% of its predictions are correct for this specific class. Having a REC value of 0.93, it successfully identifies 93% of the real instances, resulting in an *F*1-S of 0.91. The PRE of the model for Class 3 is 0.91, indicating that it correctly predicts this class in 91% of cases. The model demonstrates robust performance in detecting instances of Class 3, with a REC rate of 0.93 and an *F*1-S of 0.92. Finally, in Class 4, the model attains a PRE of 0.94, indicating its capability to classify 94% of instances in this class accurately. Additionally, it exhibits a REC of 0.95, signifying its ability to identify 95% of instances in this class correctly. Additionally, it attains an *F*1-S of 0.94 for this particular class.

The model's ACC, as measured by the overall ACC, is 0.87. This means that it accurately predicts the cyberbullying type for 87% of the instances in the evaluation set. The classification report for CNN-BiLSTM is in Table 4.

Table 4
CNN-BiLSTM classification report

Class	PRE	REC	*F*1-S	Support
0	0.75	0.71	0.73	1569
1	0.84	0.82	0.83	1568
2	0.89	0.93	0.91	1608
3	0.91	0.93	0.92	1578
4	0.94	0.95	0.94	1487
ACC	–	–	0.87	7810
macro avg	0.87	0.87	0.87	7810
weighted avg	0.87	0.87	0.87	7810

Figure 10 displays the confusion matrix and offers a comprehensive evaluation of our CNN-BiLSTM model's ACC in predicting different types of cyberbullying. The display indicates the count of accurately and inaccurately classified instances for each class.

In Class 0, there were 1120 instances accurately classified as not_cyberbullying. However, 188 instances were mistakenly classified as gender-based cyberbullying, 115 instances as religion-based cyberbullying, 105 instances as age-based cyberbullying, and 41 instances as ethnicity-based cyberbullying. In Class 1, there were 209 instances accurately identified as gender-based cyberbullying. However, 1279 instances were incorrectly classified as not_cyberbullying. Additionally, there were 31 instances classified as religion-based cyberbullying, 16 instances classified as age-based cyberbullying, and 33 instances classified as ethnicity-based cyberbullying. In Class 2, there were 80 instances accurately identified as religion-based cyberbullying. In comparison, 22 instances were incorrectly labeled as not_cyberbullying, 1491 as gender-based cyberbullying, 5 as age-based cyberbullying, and 10 as ethnicity-based cyberbullying. In Class 3, there were 65 instances accurately categorized as age-based cyberbullying. However, 13 instances were incorrectly classified as not_cyberbullying, 17 as gender-based cyberbullying, 1472 as religion-based cyberbullying, and 11 as ethnicity-based cyberbullying. In Class 4, there were 27 instances accurately categorized as ethnicity-based cyberbullying, while 13 instances were incorrectly classified as not_cyberbullying, 21 as gender-based cyberbullying, 12 as religion-based cyberbullying, and 1414 as age-based cyberbullying. The confusion matrix indicates that the model achieved higher ACC in classifying tweets about gender-based cyberbullying, religion-based cyberbullying, and age-based cyberbullying, as compared to tweets not related to cyberbullying and ethnicity-based cyberbullying. Instances of misclassifications were noted, specifically in distinguishing between not_cyberbullying and various forms of cyberbullying.

4.3. Evaluation of the BiLSTL-GRU method

Figure 11 presents the training and validation results of our BiLSTM-GRU model, which underwent 25 training epochs. After each epoch, the model's performance was evaluated on both the training and validation datasets. The training procedure involved adjusting the model's parameters to minimize loss and improve ACC.

During the training process, we noticed a progressive reduction in the loss and a simultaneous improvement in the ACC for both the training and validation datasets. That means that the model learned very widely from the input data. The decreasing loss values demonstrate that the model's predictions are close to real labels during training. The model's ACC reached a very high value of 99.81% in the training dataset in the last epoch. From this, one can see an increase in the value of the ACC on the validation set up to 90.42%. Our training results and validation prove that our BiLSTM-GRU can accurately classify different kinds of cyberbullying. Table 5 is a detailed report of the evaluation performance of our BiLSTM-GRU model in detecting different forms of cyberbullying. This paper reports Class 0 with a PRE of 0.81 and a REC rate of 0.75, which means that it could detect 75% of real instances. The *F*1-S, in this case, is 0.78, which is a pretty balanced measure between PRE and REC. For the Class 1 analysis, the model gives it a PRE of 0.87. Using the rate by way of REC, it correctly classified 87% of gender-based cyberbullying cases with an *F*1-S of 0.87, which is also a successful blend between PRE and REC. For Class 2, it recorded a PRE of 0.94, thus indicating a very high ACC for the model's predictions. Furthermore, the REC indicated that this model was able to identify 95% of the cases related to religion-based cyberbullying.

As such, the *F*1-S of 0.94 presents a comprehensive measure of these metrics in this class. The PRE analysis for Class 3 revealed that 93% of the predicted instances were age-based instances of cyberbullying. A class 4 targeted REC of 0.97 detects 97% of real cases, hence *F*1-S of 0.95. A PRE of 0.97 also shows that the model performed very well in class 4. The *F*1-S was very high at 0.97 for ethnicity instances of cyberbullying, indicating that the targeted REC had identified instances of this class at 98%. This depicts a very balanced, highly accurate representation of this class in particular.

It attached equal importance to all classes with an overall ACC and macro average *F*1-S of 0.90, while the weighted average was 0.90, the proportion of instances per class. These metrics suffice to establish that the model performed excellently in detecting these varied forms of cyberbullying. The performance of the BiLSTM-GRU algorithm on the cyberbullying dataset is given in Table 6. In particular, it shows its performance for each class, as shown in Figure 12.

It correctly classified 1252 cases of cyberbullying but misclassified 126 instances as gender-based, 84 as religion-based, 82 as age-based, and 25 as ethnicity-based. This supports the view that the model can mistakenly classify not_cyberbullying as various forms of cyberbullying and, as such, may raise FP instances. The model

Figure 10
CNN-BiLSTM confusion matrix

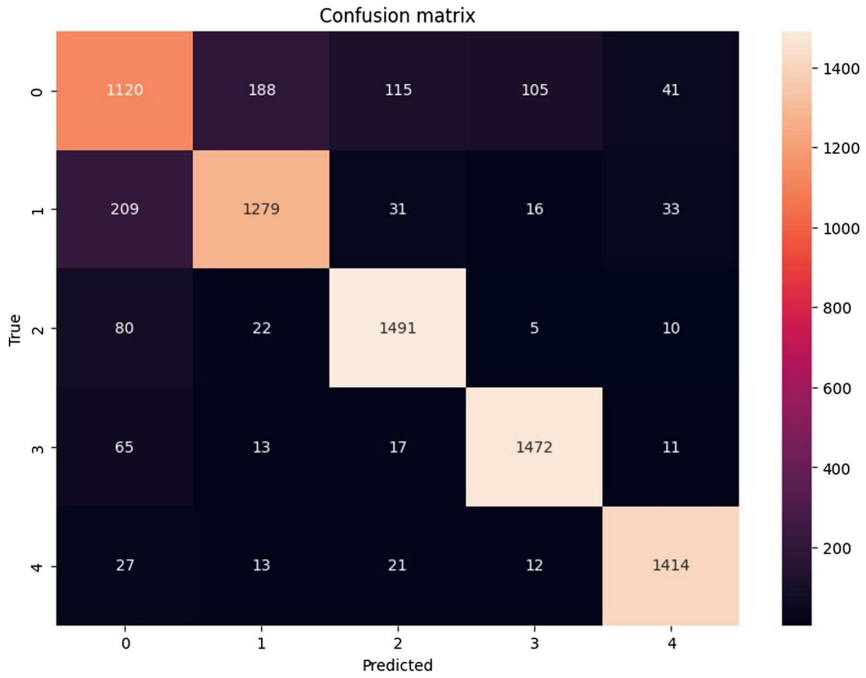


Figure 11
Accuracy and loss of BiLSTM-GRU

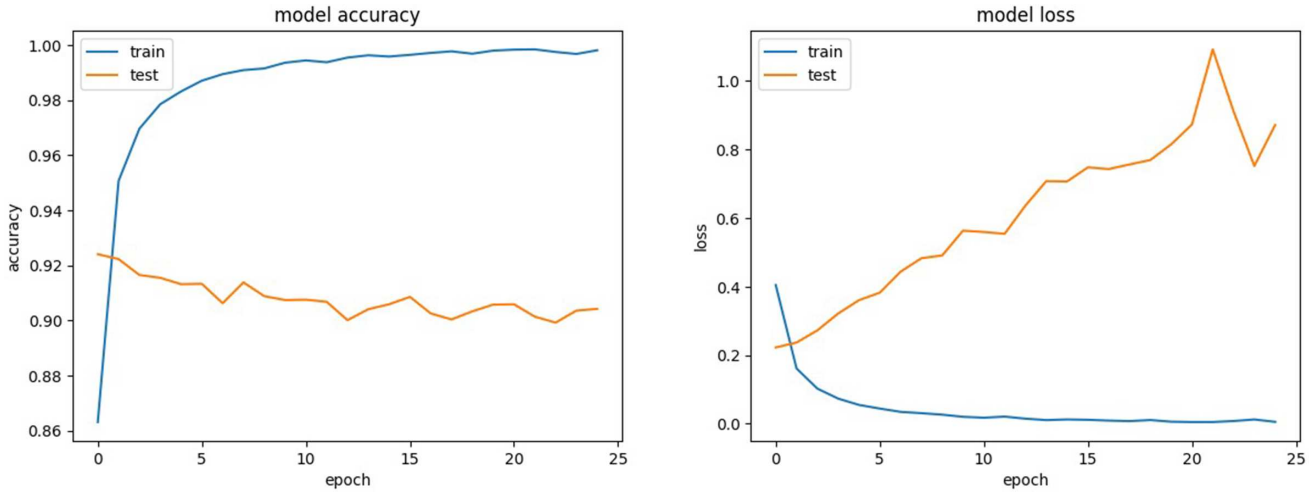


Table 5
BiLSTM-GRU classification report

Class	PRE	REC	*F*1-S	Support
0	0.81	0.75	0.78	1569
1	0.87	0.87	0.87	1568
2	0.94	0.95	0.94	1608
3	0.93	0.97	0.95	1578
4	0. 97	0.98	0.97	1487
ACC	–	–	0.90	7810
macro avg	0.90	0.90	0.90	7810
weighted avg	0.90	0.90	0.90	7810

also misclassified 1331 instances of gender, 1518 as religion, 1529 as age, and 1460 as ethnicity. These misclassifications indicate a clear line of demarcation between the forms, which takes much work to draw. In this regard, the confusion matrix will help point out areas of improvement and further analysis for refinement to better ACC and make the model more resilient in classifying the different kinds of cyberbullying.

4.4. Evaluation of the ANN method

These results for the training and validation of our ANN model are portrayed in Figure 13. In epoch one, it posted an ACC of 69.82% with a validation ACC of 91.65%. The model performed very well during the training process; at the end, it posted an ACC of 99.78% with a validation ACC of 89.15% in the last epoch. The

Figure 12
NB confusion matrix

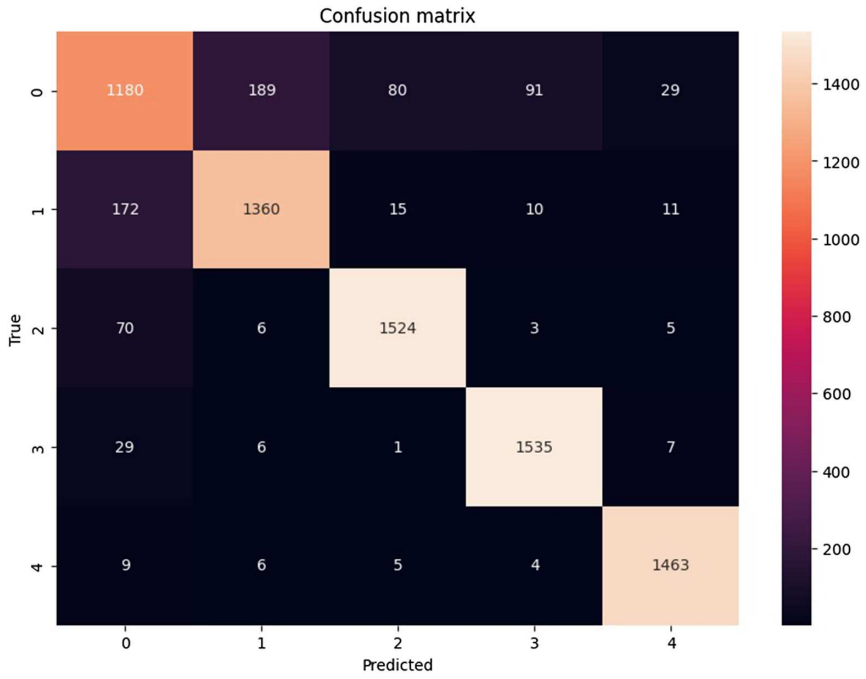
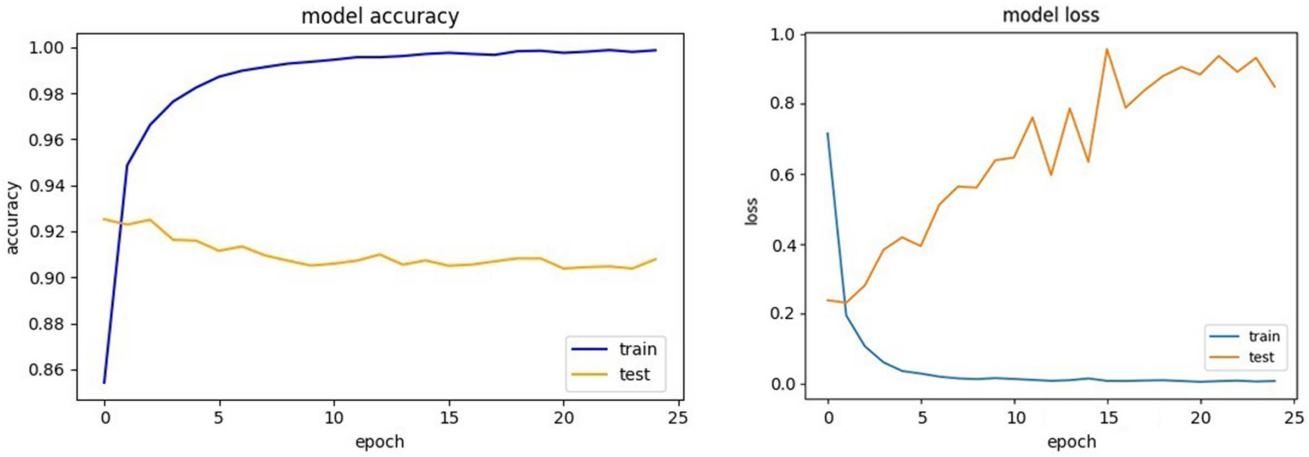


Figure 13
Accuracy and loss of ANN



loss values consistently decreased during training, thus proving that the model learned and got to tune its predictive capabilities. The loss started at 0.7151 in epoch one but dropped to a final value of 0.0077. The validation loss decreases from 0.2385 to 0.8492.

The ANN model's learning and generalization capabilities were commendable, reflected by its high ACC validation. On the other hand, there was a significant discrepancy in the ACC between the training and validation sets, which could have been indicative of overfitting problems. That happens when the model becomes too specialized for the training data and needs to generalize better on new, unseen data. Regularization or dropout can be used to handle overfitting. Moreover, users can also collect data that better represent different demographics and perspectives to improve performance and reduce bias. The classification report was balanced per category for expected occurrences versus actual occurrences. The overall ACC of the classification model was 89%, demonstrating consistent performance across various classes. Additional optimization is required to improve the ACC and resilience of the model. Table 6 shows the classification report.

The confusion matrix for the ANN model provides a detailed breakdown of its predictions for each class in the cyberbullying detection task, as shown in Figure 14.

The model accurately classified 1078 instances as TNs in Class 0. Nevertheless, 185 instances were erroneously classified as Class 0 (FN), suggesting that certain instances of Class 0 were mistakenly identified as different classes. In addition, 170 instances were inaccurately predicted as other classes (FP). In Class 1, the model accurately identified 1365 instances as TPs and correctly classified

Table 6
ANN classification report

Class	PRE	REC	*F*1-S	Support
0	0.82	0.69	0.75	1569
1	0.86	0.87	0.87	1598
2	0.91	0.94	0.93	1608
3	0.89	0.89	0.93	1578
4	0.97	0.98	0.97	1487
ACC	–	–	0.89	7810
macro avg	0.89	0.89	0.89	7810
weighted avg	0.89	0.89	0.89	7810

152 instances as TNs. Nevertheless, 25 instances were erroneously identified as Class 1 (FNs), and 13 were inaccurately categorized as different classes (FPs).

In Class 2, the model accurately identified 1517 instances as TPs and correctly classified 61 instances as TNs. Nevertheless, 24 instances were erroneously predicted as Class 2 (FNs), and 3 were mistakenly classified as other classes (FPs). In Class 3, the model accurately predicted 1551 instances as TPs and correctly classified 16 instances as TNs. Nevertheless, two occurrences were erroneously anticipated as Class 3 (FNs), while seven occurrences were inaccurately categorized as different classes (FPs). Eight instances were misclassified as TNs in Class 4; conversely, 1452 instances were correctly predicted as TPs. In addition, 10 instances were mispredicted as Class 4 – the FNs – and 9 instances that were misclassified into other classes were the FPs.

The confusion matrix presented here thus replicates the high ACC of our ANN model in accurately classifying TPs and TNs in most categories. However, in some cases, some misclassifications result in FP and FN results. A deeper analysis and understanding of the trends and characteristics of misclassified instances are essential in identifying possible ways to improve the model. We intend to correct these misclassifications to increase our artificial neural network model's ACC and performance in classifying cyberbullying.

5. Comparison Between Suggested Models

This section checks comparative performance for multiclass classification problems using four different models: BiLSTM, CNN-BiLSTM, BiLSTM-GRU, and ANN. The target variable consisted of five different classes, and the performance was measured using three different metrics: Acc, macro average, and weighted average, as shown in Table 7. Our findings show that BiLSTM has the best ACC, macro average, and weighted average among the four models. The models achieved an ACC rate of over 91% in correctly classifying instances into the dataset's five different classes, demonstrating their high PRE in classification. The BiLSTM-GRU model showed excellent performance, achieving an ACC of 90%. In contrast, the CNN-BiLSTM and ANN models exhibited inferior performance scores relative to the other models, achieving 87% and 89% ACC, respectively. Table 7 presents a comparison of the proposed models.

The potential role of ChatGPT as a zero-shot classifier for cyberbullying detection. While large-scale language models (LLMs) like ChatGPT have shown promise in text classification tasks, we highlight that they may not outperform domain-specific

Table 7
Comparison of model performance

Model	Acc	Macro Avg	Weighted Avg
BiLSTM	0.91	0.91	0.91
CNN-BiLSTM	0.87	0.87	0.87
BiLSTM-GRU	0.90	0.90	0.90
ANN	0.89	0.89	0.89

Figure 14
ANN confusion matrix

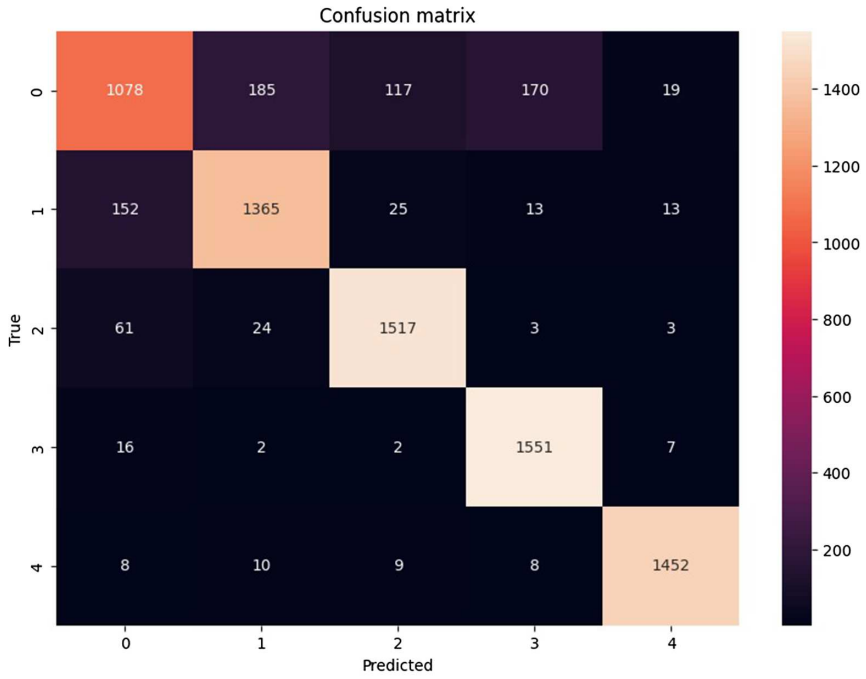
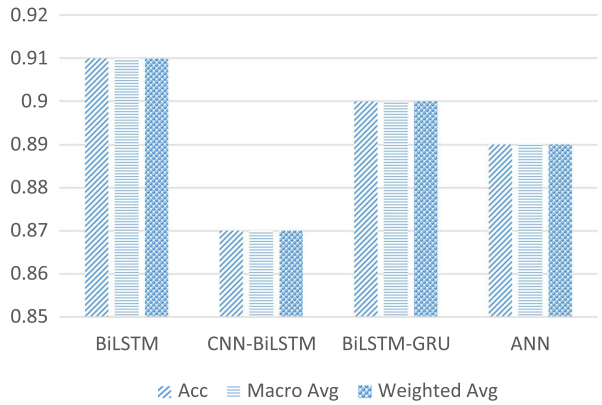


Figure 15
The performance comparison between proposed models



models like BiLSTM, which are trained on task-specific datasets. We conclude by recommending future research on combining LLMs with local models for hybrid detection systems.

Figure 15 presents the comparison between models. Our analysis indicates that the BiLSTM model exhibits high ACC and effectively classifies instances into five distinct classes. Nevertheless, the selection of a particular model for our thesis would be contingent upon several factors, including the computational resources at our disposal, the model's interpretability, and the application's specific demands.

Our study evaluated multiple models to determine their effectiveness in a particular task, such as detecting cyberbullying. We compared the ACC of different models, namely, BiLSTM, CNN-BiLSTM, BiLSTM-GRU, and ANN. The evaluation results revealed that these models attained commendable ACC scores, ranging from 0.87 to 0.91. To provide context for our findings, we compared our results to existing models previously discussed in the literature. As an example, the study was conducted by Cheng et al. [11], Kumar and Sachdeva [12], Hooda et al. [13], and Obaid et al. [14].

Based on Table 6, the BiLSTM model demonstrated the best performance across all metrics, particularly excelling in ACC, macro average, and weighted average, suggesting that it generalizes well across both dominant and minority classes. The BiLSTM-GRU also achieved high performance, while the CNN-BiLSTM and ANN models had lower scores due to slight overfitting and class imbalance issues. The use of macro and weighted averages ensures that the models are evaluated holistically, considering both dominant and minority classes.

Table 8 provides a comparative analysis of the classification performance of the BiLSTM model with the approaches proposed by Cheng et al. [11], Kumar and Sachdeva [12], and Obaid et al. [14]. Each study addresses the cyberbullying detection challenge using distinct model architectures and datasets.

The proposed BiLSTM model outperforms prior studies by demonstrating higher classification ACC and significantly improving the identification of challenging bullying types, particularly ethnicity- and religion-based cyberbullying. By employing a bidirectional memory mechanism and optimizing for multiclass imbalance, our model addresses critical limitations faced by earlier works. Additionally, our focus on text-only analysis provides a computationally simpler alternative to hybrid models. This enhanced capability has implications for the development of real-time, fair, and unbiased cyberbullying detection systems.

6. Conclusion

With the growing accessibility of the Internet, the consequences of cyberbullying are becoming more pronounced, inflicting both psychological and physical harm on its targets. To efficiently identify and tackle this problem, our primary emphasis was on thoroughly understanding cyberbullying, encompassing the typical victims and the resulting outcomes. We can only identify and address instances of cyberbullying successfully if we have a comprehensive understanding of it.

This paper has showcased the efficacy of deep learning models, specifically BiLSTM, CNN-BiLSTM, BiLSTM-GRU, and ANN, in accurately detecting cyberbullying on social media platforms. The developed model has demonstrated favorable outcomes in categorizing instances into distinct classes associated with cyberbullying, facilitating prompt interventions, and reducing potential harm. The results of this study aid in establishing a more secure digital space, safeguarding individuals against the detrimental consequences of cyberbullying, and providing insights for formulating policies and regulations to tackle this widespread problem.

As cyberbullying evolves with the advent of new communication technologies and social media trends, the strategies for detecting and mitigating it must also adapt. Future investigations may investigate the integration of real-time information streams and progress common dialect-preparing strategies to upgrade the responsiveness and PRE of discovery frameworks. Additionally, the consideration of cross-platform investigation might uncover designs and behaviors that are not clear when analyzing information from a single stage, subsequently advertising a more all-encompassing approach to combating cyberbullying. Moreover, creating models that are delicate to social and etymological subtleties may make strides in the worldwide appropriateness of these location frameworks. Finally, an intriguing collaboration including analysts, sociologists, and technologists seems to lead to the creation of comprehensive systems that not only identify cyberbullying but

Table 8
Comparative analysis of the classification performance

Model	Dataset	Accuracy	Strength
RL [23]	Instagram social media	88.2%	Debiasing of predictions using reinforcement learning
CapsNet-ConvNet Hybrid [11]	Multimodal (text + images)	89.8%	Handles multimodal data (text and images) using capsule networks
LSTM + Fuzzy Logic [13]	Twitter (47,733 tweets)	93.6%	Fuzzy logic integration for measuring bullying severity
BiLSTM	Cyberbullying Tweets (47,692)	91.0%	High accuracy for ethnicity- and religion-based cyberbullying

also address its root causes, advertising preventive measures and bolstering frameworks for casualties.

This study presents a novel approach to multiclass cyberbullying detection using advanced deep learning models (BiLSTM, CNN-BiLSTM, BiLSTM-GRU, and ANN). Our models outperform existing benchmarks, particularly in the challenging categories of ethnicity- and religion-based cyberbullying, where the BiLSTM model achieved 97% PRE and 98% REC. The study highlights how bidirectional context-aware learning enables models to identify subtle forms of bullying. Moving forward, four critical research directions should be pursued to enhance the effectiveness and fairness of cyberbullying detection systems: exploring transfer learning with models like BERT and ChatGPT to leverage the generalization capabilities of LLMs; cross-platform analysis to develop generalized, multi-platform cyberbullying detection models; real-time detection to create real-time, low-latency models capable of preventing harm as soon as it occurs; and explainable AI to ensure transparency and fairness in cyberbullying detection, especially in high-stakes environments where legal compliance and human rights are involved.

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data are available on request from the corresponding author upon reasonable request.

Author Contribution Statement

Ghaith Jaradat: Conceptualization, Methodology, Formal analysis, Data curation, Writing – original draft, Writing – review & editing, Visualization, Project administration. **Mohammad Shehab:** Conceptualization, Methodology, Software, Validation, Formal analysis, Data curation, Writing – original draft, Writing – review & editing, Supervision, Project administration. **Dyala Ibrahim:** Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Project administration. **Saif Najdawi:** Methodology, Validation, Writing – original draft, Writing – review & editing, Visualization. **Rami Sihwail:** Software, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization.

References

[1] Al-Garadi, M. A., Hussain, M. R., Khan, N., Murtaza, G., Nweke, H. F., Ali, I., ..., & Gani, A. (2019). Predicting cyberbullying on social media in the significant data era using machine learning algorithms: Review of literature and open challenges. *IEEE Access*, *7*, 70701–70718. https://doi.org/10.1109/ACCESS.2019.2918354

[2] López-Vizcaíno, M. F., Nóvoa, F. J., Carneiro, V., & Cacheda, F. (2021). Early detection of cyberbullying on social media networks. *Future Generation Computer Systems*, *118*, 219–229. https://doi.org/10.1016/j.future.2021.01.006

[3] Abaido, G. M. (2020). Cyberbullying on social media platforms among university students in the United Arab Emirates. *International Journal of Adolescence and Youth*, *25*(1), 407–420. https://doi.org/10.1080/02673843.2019.1669059

[4] Teng, T. H., Varathan, K. D., & Crestani, F. (2024). A comprehensive review of cyberbullying-related content classification. *Expert Systems with Applications*, *244*, 122644. https://doi.org/10.1016/j.eswa.2023.122644

[5] Akhter, A., Acharjee, U. K., Talukder, M. A., Islam, M. M., & Uddin, M. A. (2023). A robust hybrid machine learning model for detecting Bengali cyber bullying in social media. *Natural Language Processing Journal*, *4*, 100027. https://doi.org/10.1016/j.nlp.2023.100027

[6] Shehab, M., & Alzabin, L. R. (2025). Evaluating the effectiveness of stealth protocols and proxying in hiding VPN usage. *Journal of Computational and Cognitive Engineering, 4*(2), 186–194. https://doi.org/10.47852/bonviewjcce42023642

[7] Sultan, T., Jahan, N., Basak, R., Jony, M. S. A., & Nabil, R. H. (2023). Machine learning in cyberbullying detection from social-media image or screenshot with optical character recognition. *International Journal of Intelligent Systems and Applications*, *15*(2), 1–13. https://doi.org/10.5815/ijisa.2023.02.01

[8] Balakrisnan, V., & Kaity, M. (2023). Cyberbullying detection and machine learning: A systematic literature review. *Artificial Intelligence Review*, *56*, 1375–1416. https://doi.org/10.1007/s10462-023-10553-w

[9] Pawar, R., & Raje, R. R. (2019). Multilingual cyberbullying detection system. In *2019 IEEE International Conference on Electro Information Technology*, 40–44. https://doi.org/10.1109/EIT.2019.8833846

[10] Ali, A., & Syed, A. M. (2020). Cyberbullying detection using machine learning. *Pakistan Journal of Engineering and Technology*, *3*, 45–50. https://doi.org/10.51846/vol3iss2pp45-50

[11] Cheng, L., Mosallanezhad, A., Silva, Y. N., Hall, D. L., & Liu, H. (2021). Mitigating bias in session-based cyberbullying detection: A non-compromising approach. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2158–2168. https://doi.org/10.18653/v1/2021.acl-long.168

[12] Kumar, A., & Sachdeva, N. (2022). Multimodal cyberbullying detection using capsule network with dynamic routing and deep convolutional neural network. *Multimedia Systems*, *28*(6), 2043–2052. https://doi.org/10.1007/s00530-020-00747-5

[13] Hooda, R., Jaiswal, A., Bansal, I., Jain, M., Singh, P., & Sachdeva, N. (2022). Detection of offensive comments for textual data using machine learning. In *Advancements in Interdisciplinary Research: First International Conference*, 213–223. https://doi.org/10.1007/978-3-031-23724-9_20

[14] Obaid, M. H., Guirguis, S. K., & Elkaffas, S. M. (2023). Cyberbullying detection and severity determination model. *IEEE Access*, *11*, 97391–97399. https://doi.org/10.1109/access.2023.3313113

[15] Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1746–1751. https://doi.org/10.3115/v1/d14-1181

[16] Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for

statistical machine translation. *arXiv Preprint:1406.1078*. https://doi.org/10.48550/arXiv.1406.1078

[17] Orelaja, A., Ejiofor, C., Sarpong, S., Imakuh, S., Bassey, C., Opara, I., . . . , & Akinola, O. (2024). Attribute-specific cyberbullying detection using artificial intelligence. *Journal of Electronic & Information Systems*, *6*(1), 10–21. https://doi.org/10.30564/jeis.v6i1.6206

[18] Wijayanti, E. B., Setiadi, D. R. I. M., & Setyoko, B. H. (2024). Dataset analysis and feature characteristics to predict rice production based on eXtreme Gradient Boosting. *Journal of Computing Theories and Applications*, *1*(3), 299–310. https://doi.org/10.62411/jcta.10057

[19] Jefriyanto, J., Ainun, N., & Al Ardha, M. A. (2023). Application of Naïve Bayes classification to analyze performance using stopwords. *Journal of Information System, Technology and Engineering*, *1*(2), 49–53.

[20] Pramana, R., Debora, Subroto, J. J., Gunawan, A. A. S., & Anderies (2022). Systematic literature review of stemming and lemmatization performance for sentence similarity. In *2022 IEEE 7th International Conference on Information Technology and Digital Applications*, 1–6. https://doi.org/10.1109/ICITDA55840.2022.9971451

[21] Low, M. X., Yap, T. T. V., Soo, W. K., Ng, H., Goh, V. T., Chin, J. J., & Kuek, T. Y. (2022). Comparison of label encoding and evidence counting for malware classification. *Journal of System and Management Sciences*, *12*(6), 17–30. https://doi.org/10.33168/jsms.2022.0602

[22] Al-Obaydy, W. N. I., Hashim, H. A., Najm, Y. A., & Jalal, A. A. (2022). Document classification using term frequency-inverse document frequency and K-means clustering. *Indonesian Journal of Electrical Engineering and Computer Science*, *27*(3), 1517–1524. https://doi.org/10.11591/ijeecs.v27.i3.pp1517-1524

[23] Luque, A., Carrasco, A., Martín, A., & de las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, *91*, 216–231. https://doi.org/10.1016/j.patcog.2019.02.023

[24] Yilmaz, A. E., & Demirhan, H. (2023). Weighted kappa measures for ordinal multi-class classification performance. *Applied Soft Computing*, *134*, 110020. https://doi.org/10.1016/j.asoc.2023.110020

[25] Shannaq, F., Alshorman, A., Al-Sayyed, R., Shehab, M., & Alomari, W. (2024). Weighted service broker algorithm in cloud environment. *Informatica: An International Journal of Computing and Informatics*, *48*(7), 163–176. https://doi.org/10.31449/inf.v48i7.5617

[26] Aldhyani, T. H. H., Al-Adhaileh, M. H., & Alsubari, S. N. (2022). Cyberbullying identification system based deep learning algorithms. *Electronics*, *11*(20), 3273. https://doi.org/10.3390/electronics11203273

[27] Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: An experimental review. *Journal of Big Data*, *7*(1), 70. https://doi.org/10.1186/s40537-020-00349-y

How to Cite: Jaradat, G., Shehab, M., Ibrahim, D., Najdawi, S., & Sihwail, R. (2025). Deep Learning Approaches for Detecting Cyberbullying on Social Media. *Journal of Computational and Cognitive Engineering*, *4*(4), 451–465. https://doi.org/10.47852/bonviewJCCE52024162