ECTI Transactions on Computer and Information Technology
Journal homepage: https://ph01.tci-thaijo.org/index.php/ecticit/
Published by the ECTI Association, Thailand, ISSN: 2286-9131

The Challenges and Approaches during the Detection of Cyberbullying Text for Low-resource Language: A Literature Review

Md. Nesarul Hoque[1], Puja Chakraborty[2] and Md. Hanif Seddiqui[3]

ABSTRACT

Objective: The primary intent of this paper is to review related studies that are more corresponding to the detection of five variants of cyberbullying text, such as abusive, hateful, aggressive, bully, and toxic comment or texts of Bengali language as a sample of low-resource language, to gain a comprehensive understanding of the challenges and state-of-the-art approaches used to identify these types of text.

Materials: We have searched the associated articles on cyberbullying text detection in the Bengali language published from 2017 to July 2021 since there was no research being detected before the year 2017 on this domain-specific paradigm. After that, we scrutinize the different levels of aspects by inspecting the title, abstract, and entire text to enlist the subsequent research in this review study.

Results: After applying different levels of filtering, from the initial search results, 28 domain-centric papers are considered out of 2,745 documents. At first, we deeply analyze the context of each study and then narrate a clear comparative review in case of research challenges and approaches, as well as providing the direction for the future work on the road to the detection of cyberbullying text for the Bengali language.

Conclusion: In this paper, we discuss five variants of cyberbullying text, such as abusive text, hateful speech, aggressive text, bully text, and toxic comments over the web, and their detection process by studying existing literature in this domain. We present advice on dataset preparation, preprocess and feature extraction tasks, and classifier selection that may aid in comprehensive research for better detection.

Article information:

Keywords: Abusive, Hate Speech, Aggressive, Cyberbullying, Toxic Comment, Low-resource Language, Bengali

Article history:
Received: March 28, 2022
Revised: December 24, 2022
Accepted: April 1, 2023
Published: April 29, 2023 (Online)

DOI: 10.37936/ecti-cit.2023172.248039

1. INTRODUCTION

With the advancement of information technology, the number of internet users is growing exponentially. In this circumference, different micro-blogging and social media sites have become popular as a platform for sharing personal feelings and opinions. Among the other social media platforms, Facebook plays a prominent role in people's interaction, where about 2.9 billion monthly active users have been accounted for. In the same way, people mostly prefer YouTube (2.4 billion) and Instagram (1.4 billion) as video-sharing and photo-sharing platforms, respectively[1].

In Bangladesh, Internet use is growing in a dramatic pattern. Up to March 2021, the total number of Internet subscribers has counted as 116.4 million[2] among the total population of about 167 million[3]. With the excessive use of the internet, social networking sites like Facebook, YouTube, Instagram, etc.,

[1,3] The authors are with Department of Computer Science and Engineering, University of Chittagong, Chittagong-4331, Bangladesh., E-mail: mnhshisir@gmail.com and hanif@cu.ac.bd
[1] The author is with Department of Computer Science and Engineering, Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Gopalganj-8100, Bangladesh., E-mail: mnhshisir@gmail.com
[2] The author is with Department of Computer Science and Engineering, Premier University, Chittagong, Bangladesh., E-mail: puja.cse@std.cu.ac.bd
[1]https://backlinko.com/social-media-users
[2]http://www.btrc.gov.bd/content/internet-subscribersbangladesh-march-2021
[3]https://www.worldometers.info/world-population/bangladesh-population/

have been firmly integrated with human life in recent years. Approximately 47 million (about 27% of the total population) Facebook users are being noticed, in which most of them (about 68.8%) are male[4]. Regarding the most significant number of active users, Dhaka was the second among all the cities in the world in 2017[5].

Because of the unbounded use of various microblogging and social media sites, cyberbullying is rapidly growing by sending abusive, aggressive, hateful, bullying, and toxic comments or texts. Here, abusive comments rise to hate, sexism, racism, and cyberbullying, resulting in a psychological hamper on people, specifically children [1,2]. On the other hand, bullying texts and hate speeches on the internet increase suicidal tendencies [1,3]. In the case of linguistic aggression, it may damage an individual's social status or dignity [4]. At the same time, toxic comment expresses anti-social behavior that may hamper or unstable the online environment [5]. In Bangladesh, the rate of cyberbullying is relatively high, with around 80% of victims being women or girls aged 14 to 22[6].

In this research paper, we have studied the Bengali text as a sample of low-resource language. This Bengali language is the official and national language[7] [8] in Bangladesh, where more than 98% of people use this as their native language[9]. In addition, this language is considered the sixth most popular language globally, with approximately 270 million people using it as their mother tongue or second language[10].

In summary, the five variants of cyberbullying texts, including abusive, hateful, aggressive, bullying, and toxic comments, can cause physical or psychological harm to a person or a community and may provoke someone to destabilize society. Even though numerous studies on Bengali texts, there is still much more to explore. This study addresses the following research questions:

- How do the five variants of Bengali cyberbullying texts are co-related?
- What challenges do researchers confront during the detection of cyberbullying content?
- Which factors do we need to consider when preparing a dataset?
- Which pre-processed tasks are required after the collection of a dataset?
- Which features and feature extraction techniques should be considered?
- Which classifier models should be chosen for optimal performance?

To answer the above questions, we have searched the related papers through Google Scholar[11] that directly deal with detecting five variants of cyberbullying text for the Bengali language. After that, by intuitive analysis of these papers, we have addressed the following issues:

- Co-relation of five variants of Bengali cyberbullying text.
- Identifying the challenges encountered during the recognition of five variants of cyberbullying text in Bengali.
- Providing instructions for the preparation of a dataset.
- After obtaining a dataset, specify the necessary pre-processing steps.
- Providing suggested features and approaches for feature extraction for a more effective detection system.
- Proposing appropriate classifier models to construct an intelligent detection system.

We have structured the remaining part of the paper as follows: In ***Section 2***, we discuss existing review papers, highlighting their advantages and disadvantages. Then, we propose a review method and imitate a literature search technique to select and extract our prospective study in ***Section 3***. After that, we describe working procedures with pros and cons for each extracted literature in ***Section 4***. The comprehensive discussion about the domain we have illustrated in ***Section 5***, and the top-ranked approaches by incorporating dataset description, pre-processing, feature extraction, and classifier algorithms from the prospective articles are highlighted in ***Section 6***. ***Section 7*** conveys the challenges during the detection of five variants of Bengali cyberbullying text, and ***Section 8*** notifies some exciting points on data set preparation, pre-processing, and feature extraction tasks, as well as the selection of classifiers with a suggestive nature for future research. Finally, we present the concluding remarks and subsequent research in ***Section 9***.

2. RELATED WORK

Since the detection of Bengali cyberbullying text is a recent field of interest for researchers, a very minimal amount of review papers is recognized in this regard. The majority of review articles focused on high-resource languages like English. As far as we know, this is the first cyberbullying review article regarding Bengali texts. In the following, we have discussed some survey papers that deal with the detection of cyberbullying typed text:

Dhanya and Balakrishnan [6] reviewed 32 articles regarding hate speech detection in 11 Asian lan-

[4] https://www.napoleoncat.com/stats/facebook-users-in-bangladesh/2021/04/
[5] https://www.thedailystar.net/bytes/dhaka-2nd-among-citieslargest-active-facebook-users-1391377
[6] https://www.thedailystar.net/country/news/80-cyberbullyingvictims-are-women-cyber-crime-division-dmp-2009017
[7] http://bdlaws.minlaw.gov.bd/act-705/section-29350.html
[8] http://bdlaws.minlaw.gov.bd/act-957/section-29340.html
[9] https://einfon.com/nationalsymbols/national-languages-ofbangladesh/
[10] https://www.ethnologue.com/language/benl
[11] https://scholar.google.com/

guages. Then they showed the comparative analysis of classifiers algorithms, dataset type (balanced or imbalanced), dataset size, and performance matrices (accuracy) and offered observations during the detection of hateful speech. Here, the authors did not specify the relationship between the five variants of the cyberbullying text. In addition, they did not investigate the pre-processing and feature extraction tasks in depth.

Alsaed and Eleyan [7] inspected ten cyberbullying detection articles in various languages like Indonesian, Bengali, English, Turkish, etc. At first, the authors defined "Cyberbullying" and described the types of cyberbullying. In addition, they provided detailed illustrations of cyberbullying detection-related papers. After that, they presented a comparative study of each detection system, considering dataset size, applied algorithms, and performance matrices. Then, after identifying the research gap, they highlighted open research issues and shortcomings of the existing works. Lastly, they proposed research directions for detecting cyberbullying content in the future. The primary limitation of this study is that the authors analyzed only ten studies and did not discuss the correlation between cyberbullying-related texts. In addition, there was a lack of a detailed description of the datasets (e.g., code-mixing issue, diversity, availability, etc.), pre-processing, and feature selection tasks.

Pamungkas et al. [8] studied articles about detecting abusive text from a multi-domain and multilingual perspective. In the case of multi-domain, they studied the corresponding datasets by concerning the factors: topical focuses (e.g., misogyny, racism, xenophobia, etc.), source (e.g., Facebook, Twitter, YouTube, etc.), size, availability, annotation scheme, and data distribution (training and testing set). Furthermore, they described strategies for feature extraction, classifier models, and domain shifting between training and test data. Similarly, they presented datasets, feather representation strategies, classifier models, and a language-shifting approach from a multi-lingual perspective. Finally, they pointed out key challenges and future research opportunities during detecting abusive text in cross-domain and cross-lingual aspects. However, this review study did not correlate five variants of cyberbullying texts. In addition, it did not provide clear guidelines on preparing a dataset, pre-processing tasks, features, feature extraction techniques, and classifier model selection to build an intelligent detection system.

Fortuna and Nunes [9] offered a comprehensive definition of hate speech and its related terminology. They followed a systematic structure to extract the relevant articles and then provided a detailed comparative analysis of dataset description, general features (e.g., N-grams, TF-IDF, POS, etc.), specific features (e.g., othering language, perpetrator attributes (e.g., gender, geographical location, etc.), focus on stereotypes, etc.), classifier algorithms, performance matrices, and so on. In addition, they investigated other open-source hate speech detection projects. In conclusion, they identified obstacles and opportunities that may aid future efforts to identify hate speech. Although the authors performed excellent work, there is a lack of clarity on dataset preparation, data preprocessing, features, feature extraction methods, and choosing an appropriate classifier.

Balayn et al. [10] reviewed aggressive, abusive, harmful, and offensive language detection papers from psychology and computer science. During the literature search, they utilized their proposed term Online Conflictual Language (OCL), which includes possible related terms such as aggressive, abusive, harmful, and offensive. They defined the OCL from the perspectives of psychology, social science, and computer science. After reconciling the definition, they constructed a taxonomy. During the detection of OCL in the collected papers, the authors discovered contextual and semantic mismatches. After that, they discussed the creation of a dataset, where the factors like data collection sources, data mining techniques, data collection biases, data augmentation techniques, data pre-processing, data splitting criteria, data annotation quality, etc., were analyzed. Then, they showed a comparative study about the features and methodologies for feature selection. They illustrated the features from four categorical views: textual features, user-centric information, user social links, and conversation context. They mentioned various feature selection techniques, including Chi-square, Principle Component Analysis (PCA), and Singular Value Decomposition (SVD), and discussed biases regarding the features from various perspectives. After the feature analysis, the authors focused on several classification algorithms, and they explained multiple biases regarding these algorithms. For the comparative examination of system performance, this study addressed several criteria, including the distribution of the dataset into training and test set, dataset class labels, evaluation metrics, accountability and transparency, and the refinement of the metrics. In conclusion, they identified numerous challenges and issues and attempted to provide solutions.

In summary, almost all the survey studies mainly collected English-specific datasets and showed comparative analysis based on these datasets. In our survey paper, we concentrate on Bengali datasets and present a review framework (see ***Section 3***) with critical comparative analysis (see ***Sections 5*** and ***6***) to provide guidelines (see ***Section 8***) for detecting Bengali cyberbullying text.

3. METHODS AND MATERIALS

We have found minimal research for identifying abusive, hateful, aggressive, bullying, and toxic texts in a low-resource Bengali language. However, researchers give extra attention to detecting cyberbullying-type texts. This section presents a review strategy for answering the research questions mentioned in ***Section 1***. In addition, we illustrate the literature selection process with a basic statistical comparison.

3.1 Review Methods

We divide the review methods into four segments: literature search with keywords, filtering irrelevant documents, comparative analysis among the extracted articles, and critical reflections to address the research issues (see ***Fig. 1***). In the first segment, we looked for relevant documents using domain-specific keywords, as detailed in ***Section 3.2***. The second segment describes various levels of filtering and identifies the desired articles (see ***Section 3.3***). In the third segment, we deploy the understanding of the five variants of Bengali cyberbullying text (see ***Section 5***) and the extensive comparative analysis (see ***Section 6***). In the last segment, we have co-related five variants of Bengali cyberbullying text (see ***Section 5***) with mentioning research challenges (see ***Section 7***). To construct an intelligent detection system, we also provide more precise suggestions for creating a dataset, pre-processing tasks, features, feature extraction methods, and selecting appropriate classifiers (see ***Section 8***).

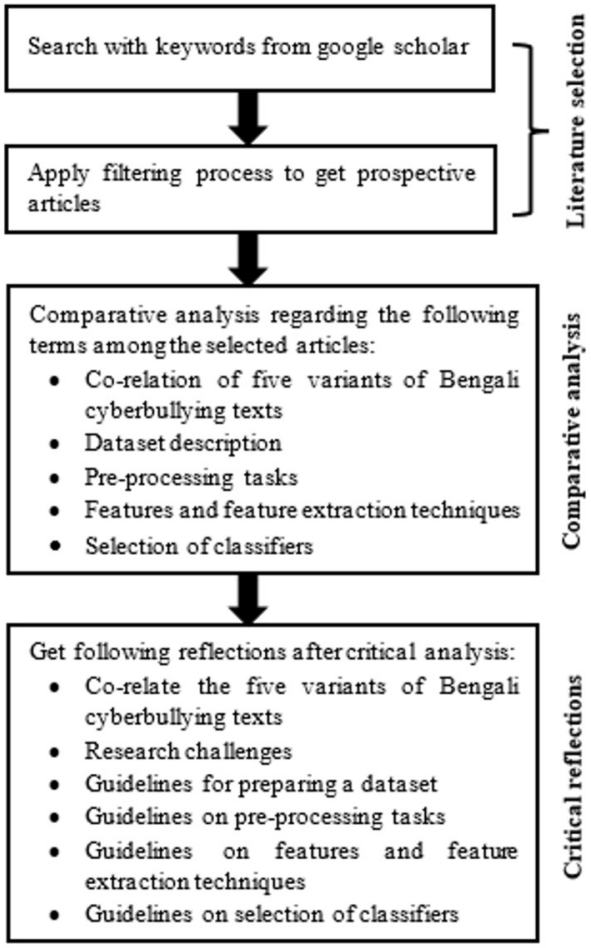

Fig.1: *Review Methods.*

3.2 Literature Search

To find the domain-specific articles, we have individually searched each variant of cyberbullying text through the Google Scholar platform. In the abusive text, we have used the keywords: abusive detect "machine learning" Bengali. For the other four variants, we have just replaced the first word with "hate speech" for hateful text, "aggressive" for aggressive text, "bully" for bully text, and finally, "toxic comment" for toxic text. The search result shows the highest value for aggressive text with 1,531 documents. In contrast, the lowest value found in the toxic comment is 39. In the case of abusive, hateful, and bullying texts, the value is 593, 334, and 248, respectively. Since the search with keywords via Google Scholar is not only based on the title but rather full text, most links are not directly associated with our prospective study.

3.3 Study Selection

Since most of the articles from the search result are not domain-centric, particularly not for the Bengali language, we have applied different levels of filtering to select and extract our desired articles for review. Different stages of the exclusion process have depicted in ***Fig. 2***. At first, we remove all surveys, reviews, reports, book or book chapters, presentations, and non-Bengali content. In the second elimination stage, we scrutinize the abstract and discard those articles mentioned in the first elimination level. Finally, we again filter the less important contents by reading and inspecting the whole text. Furthermore, we also exclude non-technical, less significant, and weakly associated content regarding our domain. Applying these three consecutive processes, we have selected 28 research articles for our final study, listed in ***Table 1***. Among those, eight pieces are on abusive text, six are on hateful speech, eleven articles deal with aggressive text, two are on bullying text, and the remaining two are on toxic comments (see ***Fig. 3***). Here, one paper covers the both hateful and aggressive text. Among the extracted 28 pieces article, there are five journal papers, fourteen conference papers, and the remaining nine have come from workshops or symposiums (see ***Fig. 4***). The enlisted articles were published between 2017 to July 2021 (see ***Fig. 5***) where one got published in 2017, three in 2018, seven in 2019, ten in 2020, and seven in 2021. Detecting the five cyberbullying text variants is a relatively new topic of

study, as no research articles we do not found before 2017.

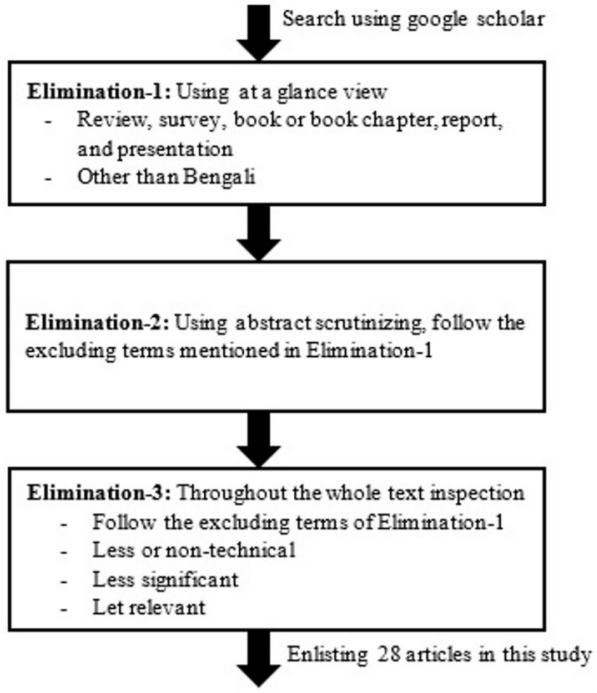

Fig.2: *Literature Selection Process.*

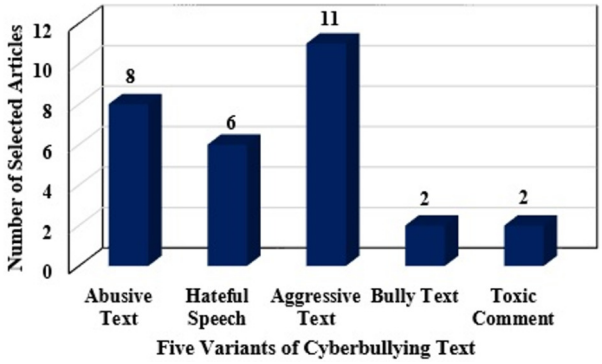

Fig.3: *Number of Literature by Cyberbullying Types Text.*

4. REVIEWED LITERATURE

In this study, we review 28 publications to identify Bengali cyberbullying texts. The following is a brief synopsis of these articles:

Ahmed et al. [11] worked on Bengali text (5,000 comments), Romanized Bengali text (7,000 comments), and the combined Bengali and Romanized Bengali text (12,000 comments). The authors performed fundamental pre-processing activities, such as deleting duplicate data, punctuation, numbers, hyperlinks, etc. They did not consider emoji and emoticons. Furthermore, stop words were not eliminated, nor was stemming or lemmatization performed to

Table 1: *Prospective Literature List.*

Study	Cyberbullying Text Type	Paper Type	Year
[11]	Bully	Conference	2021
[12]	Abusive	Workshop-conference	2021
[13]	Aggressive	Journal	2021
[14]	Hateful speech	Journal	2021
[15]	Hateful speech	Conference	2021
[16]	Abusive	Journal	2021
[17]	Aggressive and hateful speech	Journal	2021
[18]	Hateful speech	Conference	2020
[19]	Aggressive	Workshop-conference	2020
[20]	Aggressive	Workshop-conference	2020
[21]	Aggressive	Workshop-conference	2020
[22]	Aggressive	Workshop-conference	2020
[23]	Aggressive	Workshop-conference	2020
[24]	Aggressive	Workshop-conference	2020
[25]	Aggressive	Workshop-conference	2020
[26]	Aggressive	Workshop-conference	2020
[27]	Aggressive	Journal	2020
[28]	Abusive	Conference	2019
[1]	Abusive	Conference	2019
[29]	Hateful speech	Conference	2019
[30]	Abusive	Conference	2019
[31]	Hateful speech	Conference	2019
[5]	Toxic	Conference	2019
[32]	Toxic	Conference	2019
[33]	Abusive	Conference	2018
[34]	Abusive	Conference	2018
[35]	Bully	Conference	2018
[36]	Abusive	Conference	2017

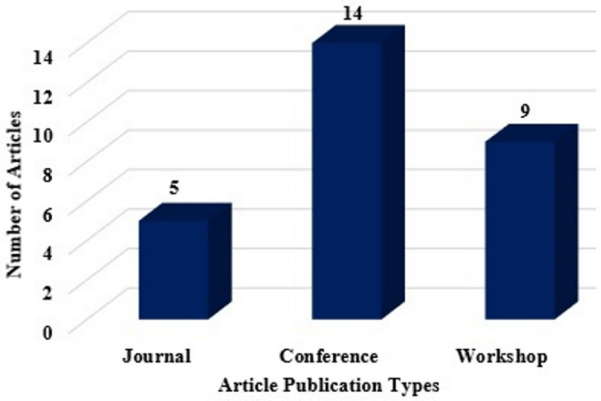

Fig.4: *Number of Literature by Article Publication Type.*

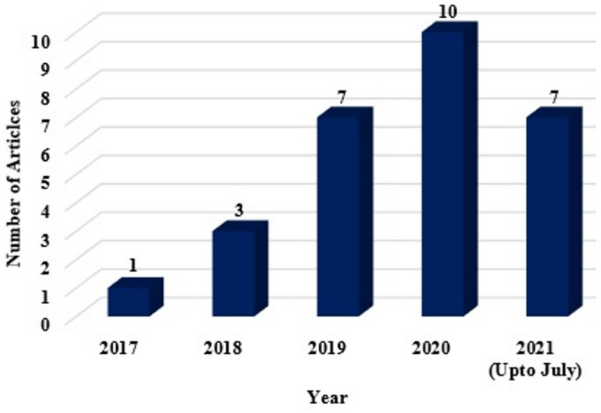

Fig.5: *Year-wise Number of Literature.*

maximize features. In this study, the authors analyzed the performance of multiple machine learning (ML) and deep learning (DL) classifiers, including Multinomial Naive Bayes (MNB), Support Vector Machine (SVM), Logistic Regression (LR), and Extreme Gradient Boosting (XGBoost), and Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), Bidirectional LSTM (BLSTM), and Gated Recurrent Unit (GRU). In the feature extraction phase, ML classifiers employed TF-IDF (Term Frequency - Inverse Document Frequency), while DL classifiers used one hot representation. The CNN model produced the excellent results (84.0% accuracy) on Romanized Bengali text; nevertheless, MNB performed well on the Bengali and combined datasets (84.0% and 80.0% accuracy, respectively). Therefore, there are still opportunities to improve accuracy. Despite the fact that the authors only experimented with a limited sample, they assert that this is the first study using Romanized Bengali text and the combined dataset for recognising cyberbullying material.

Sazzed [12] conducted experiments on 3,000 transliterated Bengali comments. This study considers three classical ML classifiers: LR, Random Forest (RF), and SVM, and one DL classifier: BLSTM. Using SVM with uni-gram, bi-gram, and TF-IDF scoring, the author obtained the best F1-scores (0.827 ± 0.010). However, they did not precisely mention the pre-processing tasks. Moreover, a detailed comparative performance analysis is also missing when detecting abusive text.

Ranasinghe and Zampieri [13] built an offensive text recognition system for seven low-resource languages: Arabic, Bengali, Danish, Greek, Hindi, Spanish, and Turkish. They employed a mix of XLM-RoBERTa (XLM-R) and the Transfer Learning (TL) approach, in which XML-R was applied to the English dataset to train the model. Then, the authors preserved the model's weights and the softmax layer. They finally transferred these stored values to low-resource languages. In the case of Bengali, the authors just transferred the weights and achieved the best weighted F1 score of 0.84. The authors did not mention preprocessing tasks and feature extraction strategies for recognizing Bengali aggressive text acquired from the TRAC-2 shared task [37]. This paper dealt with off-domain (training data comes from Twitter and test data from Facebook) and off-task (training dataset was binary classification and test dataset was three levels classification) Bengali language data.

Das et al. [14] experimented with 7,425 Bengali comments on hate speech. This research addresses binary classification (hateful and non-hateful) and multi-classification (hate speech, aggressive remark, religious hatred, ethnic attack, religious comment, political comment, and suicidal comment) problems. The authors performed numerous pre-processing operations, including removing punctuation, tokenization, stemming, stop-word removal, etc. They incorporated emojis and emoticons by constructing the "Bangla Emot" module. They also employed TF-IDF and word embedding for feature extraction. Lastly, they applied classification models, including attention-based decoder, LSTM, GRU, SVM, Naïve Bayes (NB), and RF. Attention-based decoder achieved the maximum accuracy of 77.0% and the highest F1-score of 0.78 in a multi-classification task, while its accuracy in binary classification was 88.0%. The dataset of this work is minimal, and it does not cover Bengali-English code-mixing remarks.

Romim et al. [15] worked with a relatively extensive dataset of 30,000 comments, even though the dataset was not balanced (only 10,000 samples were hateful). The authors discarded all emojis, punctuation, numeric values, non-Bengali alphabet, and symbols in the pre-processing phase. They extracted features using three word-embedding techniques: word2vec, fastText, and BengFastText. They finally implemented three classifier models: SVM, LSTM, and BLSTM, with SVM achieving the highest result with 87.5% accuracy and 0.911 F1-score. Among the three word-embedding methods, fastText produced the best results. Due to the removal of non-Bengali alphabets, this detection method cannot interpret Bengali-English code-mixing text.

Islam et al. [16] evaluated 12,028 comments to identify abusive text, where 4,880 comments are abusive, but the rest are not. The authors eliminated stickers, emoticons, and non-Bengali texts at the preprocessing level. They also conducted tokenization and stemming operations. Then, they utilized TF-IDF to extract the features. They finally implemented multiple ML models, including MNB, Multilayer Perceptron (MLP), SVM, Decision Tree (DT), RF, Stochastic Gradient Descent (SGD), Ridge, Perceptron, and K-Nearest Neighbors (K-NN) to construct an abusive text detection system. Authors experimented with stemming, not stemming, com-

plete dataset, and balanced dataset (using under-sampling). In all scenarios, SVM produced superior results. In the case of under-sampling, the accuracy of stemming decreased from 88.0% to 85.0%. Since the dataset contains just the Bengali alphabet, this method may not perform well with Bengali-English code-mixing and Romanized Bengali datasets.

Kumar et al. [17] conducted experiments on three distinct datasets: the HASOC shared task, the TRAC-2 shared task, and the combination of the HASOC and TRAC-2 shared tasks. They worked on three different languages: Bengali, English, and Hindi. Here, the authors just removed the links as the pre-processing task of the dataset. They employed a combination of character and word-level n-grams to generate features. They utilized four classifier models: SVM, Bidirectional Encoder Representations from Transformers (BERT), ALBERT, and DistilBERT. SVM with a combination of character tri-gram and word uni-gram functioned well (0.93 F1-score) to detect aggressive text in Bengali. This research showed a comprehensive error analysis of the detection system and demonstrated the correlation between aggressiveness and offensiveness in texts.

Karim et al. [18] conducted research on three distinct datasets: document classification (376,226 documents), categorization of hate speech (35,000 remarks), and sentiment analysis (320,000 statements). Authors contemplating general pre-processing work, such as removal of HTML markup, links, numbers, special characters, etc., Parts-of-Speech (PoS) tagging, stop-word removal, stemming, etc. Several ML classifiers, including SVM, KNN, LR, NB, DT, RF, and GBT, and one DL classifier, MConv-LSTM, were utilized in this study. As feature extraction methods, they employed character n-grams and word uni-grams with TF-IDF weighting for ML classifiers, while they used word2vec, GloVe, and BengFastText for DL classifiers. MConv-LSTM achieved the highest F1 scores of 0.871, 0.882, and 0.87 for document classification, hate speech, and sentiment analysis, respectively. However, the Model Averaging Ensemble (MAE) of the top three models improved the classification scores by 0.01 to 0.02 for each problem. The writers of this article claim to have created the largest Bengali word embedding, BengFastText, based on 250 million Bengali articles. This system was incapable of handling misspelled words. In addition, there is no explanation for the relationship between the five types of cyberbullying text: abusive, hateful speech, aggressive, bullying, and toxic comments or texts.

Authors of the eight articles [19-26] attended the 2nd workshop on Trolling, Aggression, and Cyberbullying (TRAC-2) at Language Resources and Evaluation Conference (LREC 2020). Participants worked on two types of tasks: aggression identification (sub-task A) and gendered aggression identification (sub-task B), in three distinct languages like Bengali, English, and Hindi. They experimented on approximately 5,000 instances and tested over 1,000 samples for each language. In [19], Risch and Krestel fixed each comment length with 200 tokens with retaining emoji. For Bengali datasets, authors utilized multiple fine-tuned ensembling BERT models based on bootstrap aggregating (bagging). They obtained the maximum weighted F1 scores of 0.80 and 0.93 for sub-task A and sub-task B, respectively. Kumari and Singh attempted to apply three word-embedding approaches, Global Vector (GloVe), fastText, and One-hot embeddings, with two DL classifiers, CNN and LSTM, in [20]. For aggressive and gendered identifications, the combination of FastText embedding and LSTM reached the highest F1 scores of 0.7175 and 0.8793 and the highest accuracy of 73.06% and 88.47%, respectively. The authors of this study did not indicate any pre-processing works of the datasets. In [21], Baruah et al. employed BERT, RoBERTa, DistilRoBERTa, XLM-RoBERTa, and SVM as the model classifiers; however, for the Bengali dataset, SVM with TF-IDF features of the word and character n-grams performed better with weighted F1-scores of 0.81 and 0.93 for sub-task A and sub-task B, respectively. This paper presented an error analysis of the classification system. The contextual information, improper grammar, and misspelled words in a comment degraded the overall system's performance. Mishra et al. [22] implemented monolingual and multi-lingual transformer networks such as BERT, m-BART, and XML-R with fine-tuning, where bert-base-multilingual-uncased demonstrated superior weighted F1-scores for sub-tasks A and B, respectively. Gordeev and Lykova [23] did not talk about text pre-processing or augmentation tasks. They employed the byte-pair encoding (BPE) approach for tokenizing the texts. Then, they used a single BERT-based model with two linear layer outputs for all sub-tasks, obtaining a weighted F1 score of 0.7716 and accuracy of 78.11% for sub-task A, and a weighted F1 score of 0.9297 and accuracy of 92.93% for sub-task B in Bengali dataset. This technique is ineffective at detecting covertly aggressive comments. In [24], Koufakou et al. achieved a 0.746 weighted F1-score for sub-task A and 0.927 weighted F1-score for sub-task B in the Bengali language using LSTM with word-aligned pre-trained vectors based on the HurtLex lexicon. Samghabadi et al. [25] utilized the BERT tokenizer to tokenize the posts or comments. They shortened each sample to 200 tokens, and the shorter instances were left-padded with zeros. In this study, the authors constructed an attention mechanism over BERT and got weighted F1-scores of 0.7369 for sub-task A and 0.9206 for sub-task B in Bengali. This system could not handle implicit types of aggressive text adequately (covertly aggressive). In [26], Datta et al. used n-grams with TF-IDF scoring and two boosting algorithms: XGBoost and Gradient

Boosting. Combining bi-gram with TF-IDF scoring and Gradient Boosting yielded a weighted F1-score of 0.4484 for sub-task A in Bengali Language. The authors did not mention any pre-processing tasks in this work. Moreover, the system's performance was inferior to that of the other contestants.

Ranasinghe and Zampieri [27] conducted an experimental study on four distinct languages: Bengali, English, Hindi, and Spanish. TRAC-2 shared task uses Bengali language, SemEval-2019 Task 6 (OffensEval) uses English, SemEval-2019 Task 5 (HatEval) corresponds to Spanish, and HASOC-2019 shared task uses Hindi. In this study, the authors used a combination of XML-R and the Transfer Learning (TL) approach, where XLM-R was used to train on the English dataset. Then, they stored the weight values and the softmax output values. Finally, they used these stored values to the new languages. However, the authors only transferred the weights to the Bengali language and got a macro F1-score of 0.8415 and a weighted F1-score of 0.8423. This system dealt with both off-domain and off-task data in the Bengali language. Here, the authors did not notify any pre-processing tasks. In addition, they did not illustrate the interconnection between five variants of cyberbullying texts.

Chakraborty and Seddiqui [28] analyzed 5,644 threat and abusive text instances. They performed numerous pre-processing operations, such as removal of unnecessary white space, special characters, and punctuation. They also conducted tokenization, stopword removal, stemming, etc. They utilized n-grams with TF-IDF calculation for ML classifiers: MNB and SVM with Linear and Radial Basis Function (RBF) kernels and word embedding for the CNN-LSTM model. Although linear SVM demonstrated the highest accuracy (78.0%), the rate of change in accuracy for the CNN-LSTM model is encouraging for the expanded dataset.

Jahan et al. [1] experimented on 2,000 comments incorporating single-coded Bengali, Bengali-English code-mixing, and transliterated Bengali text. In the pre-processing phase, authors performed link removal, profanity detection, emoji recognition with categories, tokenization, spelling correction, stemming, and sentiment score calculation. During feature extraction, they utilized n-grams with CountVectorizer, emoji, emoji categories, the number of emoji, punctuation, abusive words, curse count, number of likes, and emotion scores. Lastly, they applied three classifier models: SVM, RF, and Adaboost, with RF outperforming the others with a 72.14% correct result.

Ishmam et al. [29] investigated 5,126 comments on hateful speech. The authors divided hate comments into four categories: hate, incitement, communal hatred, and religious hatred, and non-hate comments into two categories: political and religious. Initially, the authors removed incorrect characters, punctuation, etc. They also performed tokenization, stopword removal, and stemming operations. Then, they used n-grams with TF-IDF weighting, word2vec word embedding, text readability scores, hashtags, mentions, and URLs, as well as the number of characters, words, and syllables for selecting hateful and non-hateful features. Finally, they implemented six classifier models: SVC, linear SVC, Adaboost, NB, RF, and GRU. The authors obtained the best results with 70.10% accuracy and a 0.69 F1 score by combining word2vec word embedding and the GRU model. The authors claim that this is the first study to identify hate speech in Bengali.

Emon et al. [30] researched a single-coded Bengali offensive text. They experimented on 4,700 comments that fell into seven categories: slang, religious hatred, personal attack, politically violated, antifeminism, positive, and neutral. In the pre-processing portion, they eliminated punctuation, whitespaces, emoticons, and numbers and used their stemmer. After that, they utilized n-grams with count vectorizer and TF-IDF vectorizer for ML classifiers: Linear SVC, LR, MNB, and RF, and word embedding for DL classifiers: ANN and RNN with LSTM cell. In this case, RNN (with parameter adjustment) surpassed the other algorithms by achieving a remarkable accuracy of 82.20% and an F1 score of 0.82.

Ahammed et al. [31] analyzed 1,339 data regarding hateful speech. They preprocessed the data by removing emojis, correcting misspelled words, and addressing data negation. They extracted the features using TF-IDF vectorizer and count vectorizer. Lastly, they implemented two classifiers: SVM and NB, with NB demonstrating a superior performance of 0.72 F1-score than SVM of 0.70 F1-score.

Banik and Rahman [5] developed a technique for detecting toxic comments by analyzing 10,219 samples. In the pre-processing phase, they just used the removal of punctuations and emoticons and tokenization. They utilized one-hot encoding as the feature extraction method. In this work, they tried five classifier models: NB, SVM, LR, CNN, and LSTM, in which CNN outperformed the other models with 95.30% accuracy.

Jubaer et al. [32] classified toxic comments into six categories: toxic, severe toxic, obscene, threat, insult, and identity hate. They experimented using 300 comments collected from multiple Facebook pages. They deleted punctuation and stop words and conducted tokenization to pre-process the data. Then, they extracted the features by utilizing count vectorizer. Lastly, several classifiers, including GausseanNB, MNB, Classifier Chain with MNB, Label Powerset with MNB, SVM, KNN, and Back Propagation Multi-Label Learning (BPMLL), were applied. Here, the BPMLL neural network outperformed the others with 60.0% accuracy.

Hussain et al. [33] built a system for detecting abusive content using their self-made algorithm. The authors worked on a little amount of dataset of 300 comments. They manually eliminated special characters, punctuation, conjunctions, and Unicode emotions from the comments and used n-grams during the feature extraction phase. In their proposed system, they allocated weight (abusive or non-abusive) for each token of an instance. If the overall abusive weight is greater than the entire non-abusive weight, the authors labeled the comment as abusive; otherwise, it is non-abusive.

Awal et al. [34] developed a technique for detecting abusive text. Initially, they translated an English dataset of 2,665 instances into Bengali using Google Translator. After that, they removed URLs, IP addresses, and other special characters and applied tokenization and stemming. Finally, they employed NB with BoW features and obtained an accuracy of 80.57% and an F1-score of 0.79.

Mamun and Akhter [35] examined 2,400 instances, of which 10% were bullying remarks. This dataset included the users' profile information. The authors separated emojis and special characters from the text before applying tokenization and stemming. Additionally, they isolated user-specific data from the dataset. Then, as the feature extraction technique, they employed the BoW and tri-gram approaches. This study utilized four classifier models: NB, J48, SVM, and KNN. However, the combination of the tri-gram, BoW approach, and SVM model achieved the best accuracy of 95.4% for just the user posts dataset and 97.27% for the user posts and user-centric information dataset. However, the small number of bullying-related posts may mislead the reliability.

Ehsan and Hasan [36] analyzed abusive language in 2,500 comments. This research did not consider any pre-processing tasks. The authors used uni-gram, bi-gram, and tri-gram with count vectorizer and TF-IDF vectorizer to extract features. They deployed several ML models, including RF, MNB, and SVM, with Linear, RBF, Polynomial, and Sigmoidal kernels. They discovered that linear SVM with trigram and TF-IDF vectorizer features produced the most accurate result.

In conclusion, to the best of our knowledge, there has been no analysis of the correlation between the five variants of cyberbullying text in the 28 reviewed articles. We also observe that the vast majority of research studies employ a minimal amount of datasets, which may mislead the actual efficacy of the Bengali cyberbullying text detection system. The effectiveness of the detecting system can be enhanced further. Researchers continue to look for techniques that handle contextual information more accurately.

5. BENGALI CYBERBULLYING TEXT

In this review paper, we focus on five variants of cyberbullying texts: abusive texts, hateful speeches, aggressive texts, bully texts, and toxic comments in the Bengali language. We have seen these texts on digital platforms, especially on different micro-blogging and social media sites. This section gives intuitive knowledge about these five text variants with comparative analysis. First, we can define each type of cyberbullying text as follows:

Abusive Text: A text with demeaning or insulting nature to an individual is considered an abusive text [12,28,34,38-41]. For example:

"আপনি কি বলতে পারবেন যে আপনি কোন লেভেল এর মোমিন গাধা???" (Do you know what a bloody fool you (Momin) are?)

Here, the victim has been insulted and demeaned in a social perspective directed to the mental damage.

Hateful Speech: A text is under the hate speech category that disrespects or dehumanizes a particular group or a member of a group on the ground of sex, gender, religion, race, local or geopolitics, disability, and so on [15,14,18,29,42-44]. Consider the following example:

"অরিজিনাল ফাকিস্তানি মাল!" (It's a genuine fuckistani (Pakistani) stuff!)

Here, the commenter disrespects the victim based on gender and racial perspective.

Aggressive Text: We can characterize aggressive text as it shows violent behavior, threat, or direct or indirect attack to hurt someone or harm the social relations of an individual or a group [38,45-48]. For instance:

"সালিরে আর সালারে আগুনে পুড়ে মারা হোক!" (Burn the boy and girl (shala and shali)!)

Here, the commenter threats the victim through violent behavior.

Bully Text: We can specify a bully text as when someone uses this to threaten or embarrass an individual or a group member [11,35,49-53]. For example:

"এটা একটা বেশ্যার আইডি।" (It's the id of a slut.)

Here, an individual feels embarrassed by this comment.

Toxic Comment: A toxic comment conveys immoral, unpleasant, or unexpected content, directing dishonour or irritation against a person or a group of people [5,32,54-57]. For instance:

"গরুর মুত খা।" (Drink the urine of a cow.)

Here, the commenter dishonors the victim with an unexpected and unpleasant comment.

From the above discussion, we see that all five variants of cyberbullying texts dehumanize or disrespect people. Therefore, it is clear that it is tough to distinguish one type of text message from another when it is transmitted via a digital platform. However, the context of hate speech may slightly differ from the other types that always occur against the ground of any particular social group or community. If we consider a sentence, "কেনো তোর কি চুলকানি উঠছে?" (Do you have a bloody itching?), in which a person is insulted without the basis of any social community or

any disability. Therefore, the hateful group does not include this comment. On the other hand, a text is closer to an aggressive class that approaches attacking or violent attitudes to harm the victim. E.g. "পিঠের চামড়া তুলে নবণ লাগিয়ে দাও খানকির পোলাদের।" (Strip off the skin from the back of these sons of the bitch.). However, this text may be an example of the other four categories, but more fitted for aggressive class. If we focus on toxic comments, there is also a slight difference. If a text expresses an entirely unexpected or unpleasant thing, the percentage of toxicity is very high in that case. "বাল ছাল তর সাউয়া।" (Hair of your vagina.). In summary, we can say that it is nearly impossible to classify a particular unfair text into the five different variants of cyberbullying classes in a mutually exclusive manner.

6. STATE-OF-THE-ART APPROACHES FOR DETECTION

Throughout the study of the 28 selected articles, we divide the overall implementation procedure for identifying cyberbullying types of texts into four subsections: dataset description, pre-processing tasks, feature extraction techniques, and selection of classifiers.

6.1 Dataset Description

Unlike the English language, there are very limited datasets available for the five variants of cyberbullying texts in Bengali, listed in ***Table 2***. Here, we have discussed over corpus's size, language variation, diversity, dataset balancing, target class labeling, and public availability. In the case of corpus's size, we have seen that most of the researchers have worked on small entries (less than 10,000) [1,12–14,17,19–26,28–36,45] while Karim et al. [18] and Romim et al. [15] worked on 35,000 hatred statements and 30,000 comments, respectively. In language variation, only four different datasets [1,11,12,37] deal with transliterated Bengali or Bengali-English code-mixed text. In contrast, other researchers concentrate on only single-coded Bengali text [5,13–16,27–36]. We are unable to locate any research literature that addresses the topic of code-switching in cyberbullying. Almost every corpus is built to fulfil the subject of diversity. The data comes from multiple platforms, including Facebook, YouTube, micro-blogging sites, online news, etc. In addition, this data also comes from various classes like sports, fashion, entertainment, celebrity, religion, crime, politics, and so on. However, in each case, there is also the possibility for enhancing the diversity of the data. Suppose we focus on balancing dataset, where only two pieces of article [16,28] covered this issue, while the remaining did not explicitly mention this property in their research tasks. The final thing is that in the classification of the target variable, in which twelve different datasets focused on only binary classification [1,5,11,12,15,16,28,31,33–36], five deal with only multi-classification problem [13,17–27,29,30,32], and only one dataset [14] covered both types of classification, wherein all cases the number of classes and class titles have differed from one dataset to other. One major problem related to the datasets is that these are not publicly available in most cases.

6.2 Pre-processing Tasks

After collecting data from various online platforms, this needs to be pre-processed. There are many reasons for this pre-processing. These are:

- Data are generally unstructured and do not follow specific standards [5].
- Removing noisy data [5,11].
- Reducing annotation efforts [18].
- Fit for Unicode encoding [14,29].
- Facilitate feature extraction [16,35].
- Feeding it into the classifiers [16,31,34].
- Improving the efficiency or performance of the system [16,31].

Because of the above issues, we have taken the following pre-processing tasks from 28 selected articles:

- Remove extra white space (e.g., multiple spaces, tab, etc.) [18,28,30].
- Discard duplicate data [11].
- Replace consecutive exclamation (e.g. !!!) and question marks (e.g. ???) with the term "exn" and "qsn", respectively [28].
- Remove all special characters and punctuation marks like '@', '♯', '$', ',', etc. [5,11,14,15,18,28–30,32–35]
- Eliminate all types of digits [11,15,18,30].
- Filter various links and URLs [1,11,17,18,34].
- Discard user tags and mentions within the post or comment [11,17,18].
- Single Exclamation (!) and question (?) marks are considered valid input [28].
- Apply Parts-of-Speech (PoS) tagging to the entire dataset [18].
- Apply tokenization where each sentence or comment has been split into small pieces of elements such as single words or terms [1,5,14,16,21,23,25,28,29,32,34,35].
- Remove the tokens of a low frequency (less than 5) [18].
- Use a spell checker to correct the lexical error or mistake in the entire data [1,31].
- Calculate sentiment polarity or scores for each comment [1].
- Handle the data negation [31].

Furthermore, we have seen other pre-processing tasks, albeit with some conflicts. These are:

- *Emoji and Emoticons:* Some authors suggest removing all the emoji and emoticons [5,11,15,16,30,31,33,35], while others are in-

Table 2: *Dataset Statistics.*

Dataset	**Source**	**Size**	**Language**	**Balancing**	**Labeling of Target Variable**	**Publicly Available**
Dataset-1 [11]	YouTube	12,000	Single-coded Bengali and Romanized Bengali	Not mentioned	Binary classification	Not available
Dataset-2 [12]	YouTube	3,000	Transliterated Bengali	Imbalanced	Binary classification	Available
Dataset-3 [13,17,19–27]	YouTube	5,971	Combination of single-coded Bengali and transliterated Bengali	Imbalanced	Multi-classification (3 labels)	Available
Dataset-4 [14]	Facebook	7,425	Single-coded Bengali	Not mentioned	Binary and multi-classification (7 labels)	Not available
Dataset-5 [15]	Facebook and YouTube	30,000	Single-coded Bengali	Imbalanced	Binary classification	Available
Dataset-6 [16]	Facebook and YouTube	12,028	Single-coded Bengali	Imbalanced	Binary classification	Not available
Dataset-7 [18]	Bengali Wikipedia dump, Bengali news articles, news dumps of TV channels, sports portals, books, blogs, and social media	35,000	Single-coded Bengali	Imbalanced	Multi-classification (5 labels)	Not available
Dataset-8 [28]	Facebook	5,644	Single-coded Bengali	Balanced	Binary classification	Not available
Dataset-9 [1]	Facebook	2,000	Combination of single-coded Bengali, Bengali-English code mixed, and transliterated Bengali	Not mentioned	Binary classification	Not available
Dataset-10 [29]	Facebook	5,126	Single-coded Bengali	Balanced	Multi-classification (6 labels)	Not available
Dataset-11 [30]	YouTube, Prothom Alo online, and Facebook	4,700	Single-coded Bengali	Not mentioned	Multi-classification (7 labels)	Not available
Dataset-12 [31]	Facebook	1,339	Single-coded Bengali	Balanced	Binary classification	Not available
Dataset-13 [5]	Social media	10,219	Single coded Bengali	Likely balanced	Binary classification	Available
Dataset-14 [32]	Facebook	Not mentioned	Single-coded Bengali	Not mentioned	Multi-classification (6 labels)	Not available
Dataset-15 [33]	Facebook, YouTube, and Prothom Alo news	3,000	Single-coded Bengali	Not mentioned	Binary classification	Not available
Dataset-16 [34]	YouTube	2,665	Single-coded Bengali	Balanced	Binary classification	Not available
Dataset-17 [35]	Facebook and Twitter	2,400	Single-coded Bengali	Imbalanced	Binary classification	Not available
Dataset-18 [36]	Facebook	2,500	Single-coded Bengali	Balanced	Binary classification	Not available

terested in retaining these for further processing [1,14,19,28] (see ***Table 3***).

- *Stop-words:* Stop-words are those words that are less useful and carry very little significant information about a sentence or a document. E.g.,"কিন্তু" (kintu), "এবং" (ebong), etc., are considered as the stop-words. These are only used to keep the sentence structure. Most research filters these types of words from the dataset [14,18,28,29,32]. In contrast, since the corpus consists of many regional languages, authors [11] do not remove stop-words to get maximum feature variants (see ***Table 4***).
- *Stemming or Lemmatization:* Since Bengali is a highly inflectional language, a dictionary or root word has many variants. For example, the root word "বাংলাদেশ" (bangladesh) has many alternative forms like "বাংলাদেশের" (bangladesher), "বাংলাদেশকে" (bangladeshke), "বাংলাদেশি" (bangladeshi), etc. Lemmatization or stemming is the process that converts inflectional words into their base word or close to the base word. In most of the literature, this pre-processing task has been implemented to get the atomic word [1,14,16,18,28–30,34,35], while some authors advise not performing this operation for retaining the variation of feature words [5,11,17] (see ***Table 5***).
- *Normalization:* In [18], authors suggested normalizing the hashtag, while in [17], authors were not interested in performing any normalization (see ***Table 6***).

6.3 Feature Extraction Techniques

The feature extraction methods aim to shrink the corpus's dimension by discarding inappropriate or less significant features for classification [16]. Throughout the learning of our prospective literature, we have got various feature extraction techniques that may divide into two groups - one is for machine learning classifiers, and the other is for deep learning classifiers:

6.3.1 Feature Extraction for Machine Learning Classifiers

To fit into the machine learning (ML) classifiers, four types of methods are repeatedly used in the existing research (see ***Table 7***). These are word-level N-grams, character-level N-grams, TF-IDF vectorization[12], and count vectorization[13]. Sometimes these techniques are used separately [1,11,14,16,31,33,35] or sometimes applied with the combination of two types of N-grams with two kinds

[12]https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[13]https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

Table 3: *Consideration of Emoji and Emoticons.*

Study	Emoji and Emoticons	Cyberbullying Text Type	Evaluation Metrics
Chakraborty and Seddiqui [28]	Considered	Abusive	Accuracy (0.78)
Jahan et al. [1]	Considered	Abusive	Accuracy (0.7214)
Emon et al. [30]	Not Considered	Abusive	Accuracy (0.822) and F1-score (0.82)
Islam et al. [16]	Not Considered	Abusive	Accuracy (0.88) and F1-score (0.88)
Hussain et al. [33]	Not Considered	Abusive	Not Mentioned
Das et al. [14]	Considered	Hateful Speech	Accuracy (0.77) and F1-score (0.78)
Romim et al. [15]	Not Considered	Hateful Speech	Accuracy (0.875) and F1-score (0.911)
Ahammed et al. [31]	Not Considered	Hateful Speech	Accuracy (0.72)
Risch et al. [19]	Considered	Aggressive	Weighted F1-score (0.82)
Ahmed et al. [11]	Not Considered	Bully	Accuracy (0.84)
Mamun and Akhter [35]	Not Considered	Bully	Accuracy (0.9727)
Banik and Rahman [5]	Not Considered	Toxic	Accuracy (0.953)

Table 4: *Consideration of Stop-words.*

Study	Stop-words	Cyberbullying Text Type	Evaluation Metrics
Chakraborty and Seddiqui [28]	Not Considered	Abusive	Accuracy (0.78)
Ishmam and Sharmin [29]	Not Considered	Hateful Speech	Accuracy (0.953)
Das et al. [14]	Not Considered	Hateful Speech	Accuracy (0.77) and F1-score (0.78)
Karim et al. [18]	Not Considered	Hateful Speech	F1-score (0.891)
Jubaer et al. [32]	Not Considered	Toxic	Accuracy (0.60)
Ahmed et al. [11]	Considered	Bully	Accuracy (0.84)

of vectorizations [12,18,20,28,30,34,36,54] while [17] experimented with the four methods both separately and combined. Besides applying N-grams with TF-IDF vectorization for minimizing the feature dimension, logistic regression with L1 regularization and principal component analysis has also been considered [29].

Table 5: *Consideration of Stemming and Lemmatization.*

Study	Stemming or Lemmatization	Cyberbullying Text Type	Evaluation Metrics
Chakraborty and Seddiqui [28]	Not Considered	Abusive	Accuracy (0.78)
Jahan et al. [1]	Considered	Abusive	Accuracy (0.7214)
Emon et al. [30]	Considered	Abusive	Accuracy (0.822) and F1-score (0.82)
Awal et al. [34]	Considered	Abusive	Accuracy (0.8057) and F1-score (0.79)
Islam et al. [16]	Considered	Abusive	Accuracy (0.88) and F1-score (0.88)
Das et al. [14]	Considered	Hateful Speech	Accuracy (0.77) and F1-score (0.78)
Karim et al. [18]	Considered	Hateful Speech	F1-score (0.891)
Kumar et al. [17]	Not Considered	Aggressive	F1-score (0.93)
Mamun and Akhter [35]	Considered	Bully	Accuracy (0.9727)
Ahmed et al. [11]	Not Considered	Bully	Accuracy (0.84)
Banik and Rahman [5]	Not Considered	Toxic	Accuracy (0.953)

6.3.2 Feature Extraction for Deep Learning Classifiers

Feature extraction techniques differ from the other extraction approaches for the neural network and the various deep learning (DL) methods (see ***Table 8***). Here, different word embedding techniques have been used that follow the dense (mostly ones or non-zeros) representation instead of sparse (mostly zeros) for a vector or a matrix. Throughout the inspection of chosen literature, several word-embedding methods like simple word embedding [58,59], word2vec [60-65], GloVe [66], fastText[14] [67,68], BengFastText, one-hot-encoding, etc. have been noted [5,11,12,14,15,18,20,28,30]. In the case of Bidirectional Encoder Representations from Transformers (BERT) [69], a transformer-based machine learning technique, it uses its embedding system [37].

[14]https://fasttext.cc/

Table 6: *Consideration of Normalization.*

Study	Normalization	Cyberbullying Text Type	Evaluation Metrics
Karim et al. [18]	Hashtag Normalization	Hateful Speech	F1-score (0.891)
Kumar et al. [17]	Not any Normalization	Aggressive	F1-score (0.93)

Other Features: With the help of the techniques mentioned above, feature words and feature vectors are extracted from the corpus. Besides, some additional features have been taken from the existing research. For each comment, these features are - emoji, number of emoji, emoji categories, number of punctuations, abusive word list, number of curse words, number of likes, sentiment polarity or scores [1], text readability scores, number of characters, words, and syllables [29], and so on.

6.4 Selection of Classifiers

Several classifiers have been taken into account from the existing literature while detecting five variants of cyberbullying texts. We have discussed each of them below:

6.4.1 Classifiers for Abusive Text

According to the prospective articles, ***Table 9*** articulates which classifiers perform better when detecting the Bengali abusive text. In [12,16,28,36], researchers got the best outcome through the Support Vector Machine (SVM) [70-73], whereas Awal et al. [34] suggested Naïve Bayes (NB) [74-77] classifier in binary classification problem. However, Hussain et al. [33] detected the abusive text with their self-proposed approach. In the multi-classification issue, Jahan et al. [1] achieved better output by using Random Forest (RF) [78-80], and Emon et al. [30] applied Recurrent Neural Network (RNN) along with a Long Short Term Memory (LSTM) [81][82] cell to get a better result.

6.4.2 Classifiers for Hateful Speech

To identify the hateful speech and then further classify them into various hatred classes, different researchers proposed different classifiers (see ***Table 10***) where Ishmam and Sharmin [29] used Gated Recurrent Unit (GRU) [83], Romim et al. [15,17] applied SVM, Das et al. [14] worked on attention-based RNN, and Karim et al. [18] experimented on Multichannel Convolutional-LSTM (MConv-LSTM). On the other hand, Ahammed et al. [31] applied NB for better accuracy in only binary classification problems.

Table 7: *Feature Extraction Techniques for ML Classifiers.*

Feature Extraction Techniques	Study	Evaluation Metrics
Word n-grams with count vectorization	[36]	Accuracy (0.89)
	[30]	Accuracy (0.822) and F1-score (0.82)
	[34]	Accuracy (0.8057) and F1-score (0.79)
Word n-grams with TF-IDF vectorization	[36]	Accuracy (0.89)
	[28]	Accuracy (0.78)
	[29]	Accuracy (0.953)
	[30]	Accuracy (0.822) and F1-score (0.82)
	[12]	F1-score (0.827±0.010)
	[18]	F1-score (0.891)
	[20]	Weighted F1-score (0.72)
	[26]	Weighted F1-score (0.4484)
Character n-grams with TF-IDF	[18]	F1-score (0.891)
	[20]	Weighted F1-score (0.72)
Count vectorization	[31]	Accuracy (0.72)
	[35]	Accuracy (0.9727)
TF-IDF	[11]	Accuracy (0.84)
	[31]	Accuracy (0.72)
	[35]	Accuracy (0.9727)
	[14]	Accuracy (0.77) and F1-score (0.78)
	[16]	Accuracy (0.88) and F1-score (0.88)
N-grams	[1]	Accuracy (0.7214)
	[33]	Not mentioned
	[17]	F1-score (0.93)
Character n-grams	[17]	F1-score (0.93)
Combination of the word and character n-grams	[17]	F1-score (0.93)

6.4.3 Classifiers for Aggressive Text

While detecting and classifying the aggressive text, all research worked on the same dataset of TRAC-2 shared task [37] in which combination of XLM-R and Transfer Learning (TL) [13,17], different types of BERT-based systems with fine-tuning [19,22,23,25], LSTM [20,24], SVM [17,21], and Gradient Boosting (GB) [54] (see ***Table 11***) had been applied to label each text into three classes, i.e., overtly aggressive, covertly aggressive, and non-aggressive.

6.4.4 Classifiers for Bully Text

There have only two articles found on this type of text. In both cases, researchers only concentrated on detecting the bully text as a bully or non-bully in what Ahmed et al. [11] gained the best result by incorporating Convolutional Neural Network (CNN) [84-86] and Multinomial Naive Bayes (MNB) classifiers. In contrast, Mamun and Akhter [35] used SVM (see ***Table 12***).

Table 8: *Feature Extraction Techniques for DL Classifiers.*

Feature Extraction Techniques	Study	Evaluation Metrics
General word embedding	[28]	Accuracy (0.78)
	[30]	Accuracy (0.822) and F1-score (0.82)
	[12]	F1-score (0.827±0.010)
	[14]	Accuracy (0.77) and F1-score (0.78)
Word2vec	[15]	Accuracy (0.875) and F1-score (0.911)
	[18]	F1-score (0.891)
FastText	[15]	Accuracy (0.875) and F1-score (0.911)
	[20]	Weighted F1- score (0.72)
BengFastText	[15]	Accuracy (0.875) and F1-score (0.911)
	[18]	F1-score (0.891)
GloVe	[18]	F1-score (0.891)
	[20]	Weighted F1-score (0.72)
One-hot-encoding	[11]	Accuracy (0.84)
	[5]	Accuracy (0.953)
	[20]	Weighted F1-score (0.72)
BERT pre-trained embedding	[25]	Weighted F1-score (0.9206)

Table 9: *Preferred Classifiers from Existing Articles for Abusive Text.*

Classifiers	Study	Evaluation Metrics
Support Vector Machine	[36]	Accuracy (about 0.89)
	[28]	Accuracy (0.78)
	[12]	F1-score (0.827±0.010)
	[16]	Accuracy (0.88) and
		F1-score (0.88)
Random Forest	[1]	Accuracy (0.7214)
Self-developed algorithm	[33]	Not mentioned
Naïve Bayes	[34]	Accuracy (0.8057) and F1-score (0.79)
Recurrent Neural Network	[30]	Accuracy (0.822) and F1-score (0.82)

6.4.5 Classifiers for Toxic Comment

Like bully text, only two pieces of the article have been traced for the toxic comment. In [5], the CNN classifier performed well in binary classification, while Jubaer et al. [32] dealt with the multi-classification problem through the Back Propagation Multi-Label Learning (BPMLL) [87] approach (see ***Table 13***).

7. CHALLENGES TO DETECT BENGALI CYBERBULLYING TEXT

Research on detecting cyberbullying text with its five variants, bully text, abusive text, hateful speech, aggressive text, and toxic comments, is a recent study for the Bengali language. As the low resource lan-

Table 10: *Preferred Classifiers from Existing Articles for Hateful Speech.*

Classifiers	Study	Evaluation Metrics
Gated Recurrent Unit	[29]	Accuracy (0.953)
SVM	[15]	Accuracy (0.875) and F1-score (0.911)
	[17]	F1-score (0.93)
NB	[31]	Accuracy (0.72)
Attention-based RNN	[14]	Accuracy (0.77) and F1-score (0.78)
Multichannel Convolutional-LSTM	[18]	F1-score (0.891)

Table 11: *Preferred classifiers from existing articles for aggressive text.*

Classifiers	Study	Evaluation Metrics
Combination of XLM- R and TL	[13]	Macro F1-score (0.84) and Weighted F1-score (0.84)
	[27]	Macro F1-score(0.8415) and Weighted F1-score (0.8423)
Variation of BERT models with fine-tuned	[19]	Weighted F1-score (0.82)
	[22]	Weighted F1-score (0.78)
	[23]	Weighted F1-score (0.7716)
	[25]	Weighted F1-score (0.9206)
	[17]	F1-score(0.93)
LSTM	[20]	Weighted F1-score (0.72)
	[24]	Weighted F1-score (0.746)
SVM	[17]	F1-score(0.93)
	[21]	Weighted F1-score (0.81)
GB	[26]	Weighted F1-score (0.4484)

Table 12: *Preferred Classifiers from Existing Articles for Bully Text.*

Classifiers	Study	Evaluation Metrics
CNN	[11]	Accuracy (0.84)
MNB	[11]	Accuracy (0.84)
SVM	[35]	Accuracy (0.9727)

Table 13: *Preferred Classifiers from Existing Articles for Toxic Text.*

Classifiers	Study	Evaluation Metrics
CNN	[5]	Accuracy (0.953)
BPMLL	[32]	Accuracy (0.60)

guage, researchers face various difficulties in this context. We have noted the following points to mention the research gaps in this domain:

- Dataset Related Issues: We discovered inconsistencies in the corpus size, language variation, dataset balancing, target class labeling, and dataset accessibility. The vast majority of studies employ a small number of instances, a single-coded Bengali language, an imbalanced dataset, and binary classification (See ***Table 2***). Furthermore, there are extremely few publicly accessible datasets, which may restrict efforts to enhance existing works.
- Dealing with Complex Sentences: The sentences that are too long and convey a complex nature are challenging to label correctly.
- Handling Misspelling Text: The text with misspellings is sometimes difficult to understand or indicates a different meaning, resulting in the wrong classification [18,21,30].
- Finding the Commenter's Past Behavior: Besides detecting unfair comments, it is also necessary to find out the personality trait of the commenters throughout the analysis of their prior attitudes or earlier comments [34].
- Dealing with Social Media Photo Comments: Photo comments sometimes unveil the personal expression of the commenters against the victim [14]. Therefore, consideration of the photo comments may be another challenge for the researchers.
- Lack of Interaction among the Various Cyberbullying Text: Only one paper tries to do a comparative analysis of aggressive, hate, and offensive comments [17]. So, there has still a lack of understanding of the correlation between abusive, hateful, aggressive, bullying, and toxic texts or comments.
- Lack of Real-time Detection Software: There is minimal real-time application software automatically detecting the Bengali cyberbullying text. One of the cyberbullying apps, named the "Cyber Teens"[15], works on the small aspect, and supports to protect from cyberbullying. However, the automated detection system is not embedded here.

8. DISCUSSION

Detecting five variants of Bengali cyberbullying text is one of the major challenging issues for recent research. According to the comparative study of ***Section 6*** and ***7***, it is clear that if we want to develop a robust detection system along with showing outstanding performance, we need to focus on four points, i.e., preparing dataset, pre-processing tasks, feature selection or extraction, and selection of appropriate classifiers, at the same time. We have discussed each point by providing guidelines as follows:

8.1 Guideline on Preparing Dataset

The corpus is crucial for the prediction-based system. The following factors should be thought about when creating a dataset:

[15]https://cyber-teens.com/

- Comments or posts should be collected from several platforms like Facebook, YouTube, Twitter, online news, blogs, etc. Besides, the collected data should be taken from different categories such as entertainment, fashion, sports, celebrity, politics, geopolitics, religion, culture, women, crime, and so on to cover data diversity.
- Since the performance increases with the dataset's expansion, the corpus size should be as large as possible [12,28,33].
- The dataset should be balanced, i.e., the number of cyberbullying and non-cyberbullying instances should be equal because the imbalanced dataset can cause adverse effects on the classification performance while applying machine learning algorithms. In an unbalanced dataset, classification accuracy tends to favor the majority class [88].
- Code-mixing and transliterated or Romanized Bengali are typical in modern-day communication [89,90]. Therefore, besides Bengali Unicode, we can consider code-mixing, code-switching, and transliterating Bengali to the newly built dataset.
- After detecting the cyberbullying content, it may classify into further sub-classes for future investigation.
- Finally, suppose there is a facility from the developer's side for open access to datasets from the web. In that case, many researchers will be highly interested in contributing to this research domain.

8.2 Guideline on Pre-processing Tasks

After taking data from different sources, many researchers pre-process the data because of the noisy text and unstructured arrangement of data. Throughout the analysis of pre-processing tasks, given in ***Section 6.2***, we have suggested the following points in this regard:

- Unwanted strings like links, URLs, and IP addresses can be removed from the text since these data may not convey the significant meaning of a particular post or comment [34,91].
- The uninformative data, such as special characters, punctuation marks, numbers, HTML tags, and extra white space, can be eliminated from the corpus [92].
- Tokenization is the process of text segmentation that splits a text into a piece of words, phrases, or other significant terms [93]. It is the prerequisite of other pre-processing tasks like stop-words removal, stemming, etc. [93,94].
- In the Bengali-English code-mixing dataset, a lower casing of English text is needed [17,19,21]. This task may help for the subsequent pre-processing tasks, such as removing duplicate data, stop-words removal, stemming, and so on.
- Duplicate data (e.g., characters, words, or sentences) can be removed [11]. This type of data may increase the computational time during other preprocessing tasks like tokenization, stop-word removal, stemming, and so on.
- Misspelled words lead to enlarging the vocabulary list [18]. At the same time, these types of terms degrade the performance of the detection system [21,30]. Therefore, an automatic spell checker must be embedded in the detection system to handle the misspelled words.
- Emojis (Unicode characters) and emoticons (ASCII characters) convey the expression or opinion of the user from which we can calculate a polarity score of a text [95,96]. Therefore, we can consider these types of characters that may be significant in this detection system.
- Stop words make a document larger and do not convey the semantic meaning of the document [92-94]. Removal of stop words reduces the dimension of feature space [94]. Thus, removing the stop-words from the text is an essential pre-processing task.
- By reducing the number of unique words, lemmatization or stemming saves memory space and processing time [94,97,98]. Hence, lemmatization or stemming is another significant pre-processing task.

8.3 Guideline on Feature Selection or Extraction

Feature extraction techniques are different for machine learning (ML) and deep learning (DL) classifiers, as discussed in ***Section 6.3***. In this section, we have discussed other extraction techniques for ML and DL classifiers below:

- *For ML classifiers:* To extract the term features, we can consider N-grams for both character and word levels where the value of N is varied up to 5 [17]. Apart from applying several variations of N-grams, the TF-IDF vectorizer is also suggested instead of the count vectorizer [36]. The reason is that count vectorizer calculates just term frequency for assigning weight on each term, whereas TF-IDF measures term value according to the importance of that term in the text. Moreover, we can also use principal component analysis and logistic regression with L1 regularization for optimizing feature dimensions [29]. Thus, computational complexity is minimized.
- *For DL or Neural Networks:* Three embedding techniques, i.e., word2vec [60-65], GloVe

[66] and fastText for the Bengali language [18,68], have been suggested in this regard. At first, focus on word2vec that, works on local context windows by incorporating focus words and neighbor words from the corpus and deals with word-similarity and word-analogy syntactically and semantically. For a large corpus size (about one trillion words), it shows better output. It takes less execution time than the other neural network-based models like the feedforward Neural Net Language Model (NNLM) [58], Recurrent Neural Net Language Model (RNNLM) [99], etc., because it avoids the expensive hidden layer and applies a log bi-linear regression model [60,61]. However, it gives a slightly poor performance on global co-occurrence statistics of the corpus or when the negative sample size is increased [66]. In the case of GloVe, that is formed by taking the positive sides of count-based methods like Latent Semantic Analysis (LSA) [100] and prediction-based methods like word2vec. It deals with global corpus statistics using a weighted least squares technique and captures meaningful sub-structure using a bilinear logarithmic method. This word representation technique also performs better than the word2vec for the small vector sizes and the small corpus. However, it struggles when the word or phrase vectors or the dataset are too large [66]. Finally, concentrate on fastText, an extension of word2vec. This fastText model represents the vectors from the character N-gram instead of direct word vectors. Like the word2vec model, fastText has two model architectures, skip-gram and CBOW. In the skip-gram model, each word is split into character N-grams format, and then each piece (sub-word) is converted into vector form. At the same time, a vector representation of a word is obtained by summing vectors of character n-grams associated with that word. A similar concept of sub-word information is implemented in the CBOW model. In this model, besides applying character n-grams of each context word, the positional weight [101] of each neighboring-word vector is accounted for as well [68,102,103]. Unlike the word2vec and GloVe, the fastText model can handle the rare words or words not present in the model dictionary through the subdivision of word vectors [18]. Another embedding technique, named BERT embedding, integrated with the BERT classifier, may also be considered [25].

- *Other Features:* In addition to the techniques mentioned above, some features such as abuse or curse word list, number of curse words, sentiment score [104,105], etc., for each comment or text may be considered [1].

8.4 Guideline on Selecting the Classifiers

Scrutinizing all the 28 related articles, we have seen that SVM is the preferable ML classifier for the researchers. It has many reasons, these are:

- The main reason is the working principle of the SVM, where a hyper-plane is created by maximizing the distance between two margins of two different classes (say positive and negative) of support vectors. This marginal distance makes a more generalized model. Furthermore, SVM uses a regularization technique with the soft margin hyperplane over misclassified data points to handle the errors [70-73,106].
- Kernels empower the SVM classifier. This classifier has different kernels like polynomial, radial bias function (RDF), sigmoid, and so on that transform the data point from low to high dimensional space for separating the two types of vectors by creating a hyper-plane [71].
- SVM can perform better for a large dataset using minimum enclosing ball (MEB) clustering [107].
- Unlike neural networks, SVM can generate a unique solution rather than multiple solutions [106,108].

Besides SVM, different pre-trained models like CoVe (context vectors) [109], ELMo (Embeddings from Language Models) [110], ULMFiT (Universal Language Model Fine-tuning) [111], OpenAI GPT (Generative Pretraining) [112], BERT [69], etc., have brought new dimensions in the NLP field. BERT has become a state-of-the-art model for a various context-specific tasks like language inference and questions answering among these pre-trained models [69]. We have discussed some key points about this model below:

- BERT is a purely deep bi-directional language model [69]. Here, only the encoder part is extracted using the self-attention mechanism-based transformer neural network [113] instead of the LSTM technique [110,112].
- In the pre-train framework of this model, two tasks, Mask Language Model (MLM) and Next Sentence Prediction (NSP), deal with both sides (left and right) of the context of a text in all self-attention layers to understand the language [69].
- The whole sequence of words as a token is inputted into the encoder layer and processed simultaneously to reduce the training time. Here, three embedding vectors - token, segment, and position embeddings are used for a particular word, and by adding these three vectors, a final embedding of that word is calculated [69].

- BERT's fine-tuning framework facilitates different custom hyper-parameter tuning. It outperforms verities of NLP-related downstream tasks like named entity recognition, sentence prediction, sentiment classification, etc., by slightly modifying task-specific architecture [69].
- One variation of this model, such as DistilBERT [114], uses the technique of knowledge distillation and also performs well in the aggressive text classification [17].

Moreover, another transformer-based model, XLM-RoBERTa (XLM-R) [115], may apply in the cross-lingual benchmarks for classifying five variants of cyberbullying texts [13,27].

Finally, we may also use the MConv-LSTM model to recognize the five variants of cyberbullying texts. Here, the MConv with fixed filter sizes handles the local dependency of a sentence's contiguous words. On the other hand, the LSTM layer can convey the overall reliance of an entire sentence by preserving the long-term dependence on the features. Combining these two layers may focus on local information and the overall relationship of a whole sentence [18].

Evaluation metrics: Researchers use various evaluation metrics like accuracy, F1-score, AUC, etc., to measure the performance of their system quantitatively. We have found 18 datasets from our enlisted articles. These datasets are uneven in size, language code, data-balancing, target class labeling, etc. (see ***Table 2***). Consequently, the systems' performance has differed from one dataset to another. Furthermore, several pre-processing and feature extraction tasks and using classifiers and their proper parameter tuning have impacted the system's performance. Therefore, the higher value or score of a system's evaluation metrics does not ensure that one detection system performs well than the others. This review paper does not focus on a comparison of performance metrics for these reasons.

In summary, it is possible to develop a robust detection system over the five variants of cyberbullying texts if we focus on the above four segments, i.e., dataset, pre-processing, feature extraction, and classifier algorithms, with equal importance.

9. CONCLUSIONS

In this review paper, we have studied the detection of five variants of cyberbullying Bengali text, i.e., abusive text, hateful speech, aggressive text, bully text, and toxic comments, throughout the analysis of the existing articles. Here, we have illustrated the correlation among these five text variants and noticed the research gaps in this domain. Furthermore, we have provided the directions to fulfill these gaps for future studies along with the critical analysis regarding dataset preparation, pre-processing tasks, feature selection tasks, and sorting out the classifier algorithms. We hope this study will significantly expand the research scope in many aspects of this domain paradigm that may minimize malicious activities over the world wide web.

In future research, we will create a comprehensive cyberbullying detection system for Bengali text based on the recommendations of this review study. In addition, we will establish a multi-lingual (English and low-resource languages) cyberbullying detection system that incorporates image, audio, and video information.

ACKNOWLEDGMENT

We appreciate the support of the ICT Division of the Government of the People's Republic of Bangladesh. In addition, we want to give special gratitude to the three resource persons, Mr. Quazi Mosiur Rahaman, Mr. Md. Rakibul Islam, and Mr. Habibur Rahman, designated by the assistant professor in the Department of English in Bangabandhu Sheikh Mujibur Rahman Science and Technology University. They share the inherent understanding and implicit knowledge of five variants of cyberbullying texts.

References

[1] M. Jahan, I. Ahamed, M. R. Bishwas and S. Shatabda, "Abusive comments detection in bangla-english code-mixed and transliterated text," *2019 2nd International Conference on Innovation in Engineering and Technology (ICIET)*, Dhaka, Bangladesh, pp. 1–6, 2019.

[2] C. L. Nixon, "Current perspectives: the impact of cyberbullying on adolescent health," *Adolescent health, medicine and therapeutics*, vol. 5, p. 143, 2014.

[3] S. Hinduja and J. W. Patchin, "Bullying, cyberbullying, and suicide," *Archives of suicide research*, vol. 14, no. 3, pp. 206– 221, 2010.

[4] J. Culpeper, *Impoliteness: Using language to cause offence*. Cambridge University Press, 2011, vol. 28.

[5] N. Banik and M. H. H. Rahman, "Toxicity detection on Bengali social media comments using supervised models," *2019 2nd International Conference on Innovation in Engineering and Technology (ICIET)*, Dhaka, Bangladesh, pp. 1– 5, 2019.

[6] L. Dhanya and K. Balakrishnan, "Hate speech detection in asian languages: A survey," *2021 International Conference on Communication, Control and Information Sciences (ICCISc)*, Idukki, India, pp. 1–5, 2021.

[7] P. Tulkarm, "Approaches to cyberbullying detection on social networks: A survey," *Journal of Theoretical and Applied Information Technology*, vol. 99, no. 13, 2021.

[8] E. W. Pamungkas, V. Basile and V. Patti, "Towards multidomain and multilingual abusive

language detection: a survey," *Personal and Ubiquitous Computing*, pp. 1–27, 2021.

[9] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," *ACM Comput. Surv.*, vol. 51, no. 4, jul 2018. [Online]. Available: https://doi.org/10.1145/3232676

[10] A. Balayn, J. Yang, Z. Szlavik and A. Bozzon, "Automatic identification of harmful, aggressive, abusive, and offensive language on the web: A survey of technical biases informed by psychology literature," *Trans. Soc. Comput.*, vol. 4, no. 3, oct 2021. [Online]. Available: https://doi.org/10.1145/3479158

[11] M. T. Ahmed, M. Rahman, S. Nur, A. Islam and D. Das, "Deployment of machine learning and deep learning algorithms in detecting cyberbullying in bangla and romanized bangla text: A comparative study," *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Bhilai, India, pp. 1–10, 2021.

[12] S. Sazzed, "Abusive content detection in transliterated bengali-english social media corpus," in *Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching*, pp. 125–130, 2021.

[13] T. Ranasinghe and M. Zampieri, "Multilingual offensive language identification for low-resource languages," *arXiv preprint arXiv:2105.05996*, 2021.

[14] A. K. Das, A. Al Asif, A. Paul and M. N. Hossain, "Bangla hate speech detection on social media using attention-based recurrent neural network," *Journal of Intelligent Systems*, vol. 30, no. 1, pp. 578–591, 2021.

[15] N. Romim, M. Ahmed, H. Talukder and M. S. Islam, "Hate speech detection in the bengali language: A dataset and its baseline evaluation," in *Proceedings of International Joint Conference on Advances in Computational Intelligence*. Springer, pp. 457–468, 2021.

[16] T. Islam, N. Ahmed and S. Latif, "An evolutionary approach to comparative analysis of detecting bangla abusive text," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2163–2169, 2021.

[17] R. Kumar, B. Lahiri and A. K. Ojha, "Aggressive and offensive language identification in hindi, bangla, and english: A comparative study," *SN Computer Science*, vol. 2, no. 1, pp. 1–20, 2021.

[18] M. R. Karim, B. R. Chakravarthi, J. P. McCrae and M. Cochez, "Classification benchmarks for under-resourced bengali language based on multichannel convolutional-lstm network," *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 390–399, 2020.

[19] J. Risch and R. Krestel, "Bagging bert models for robust aggression identification," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 55– 61, 2020.

[20] K. Kumari and J. P. Singh, "Ai_ml_nit_patna@ trac-2: Deep learning approach for multilingual aggression identification," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 113–119, 2020.

[21] A. Baruah, K. Das, F. Barbhuiya and K. Dey, "Aggression identification in english, hindi and bangla text using bert, roberta and svm," in *Proceedings of the second workshop on trolling, aggression and cyberbullying*, pp. 76–82, 2020.

[22] S. Mishra, S. Prasad and S. Mishra, "Multilingual joint finetuning of transformer models for identifying trolling, aggression and cyberbullying at trac 2020," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 120–125, 2020.

[23] D. Gordeev and O. Lykova, "Bert of all trades, master of some," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 93–98, 2020.

[24] A. Koufakou, V. Basile and V. Patti, "Florunito@ trac-2: Retrofitting word embeddings on an abusive lexicon for aggressive language detection," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 106–112, 2020.

[25] N. S. Samghabadi, P. Patwa P. Srinivas, P. Mukherjee, A. Das, and T. Solorio, "Aggression and misogyny detection using bert: A multitask approach," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 126–131, 2020.

[26] A. Datta, S. Si, U. Chakraborty and S. K. Naskar, "Spyder: Aggression detection on multilingual tweets," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 87–92, 2020.

[27] T. Ranasinghe and M. Zampieri, "Multilingual offensive language identification with cross-lingual embeddings," *arXiv preprint arXiv:2010.05324*, 2020.

[28] P. Chakraborty and M. H. Seddiqui, "Threat and abusive language detection on social media in bengali language," *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, Dhaka, Bangladesh, pp. 1–6, 2019.

[29] A. M. Ishmam and S. Sharmin, "Hateful speech detection in public facebook pages for the bengali language," *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, Boca Raton, FL, USA,, pp. 555–560, 2019.

[30] E. A. Emon, S. Rahman, J. Banarjee, A. K. Das and T. Mittra, "A deep learning approach to detect abusive bengali text," *2019 7th International Conference on Smart Computing & Communications (ICSCC)*, Sarawak, Malaysia, pp. 1–5, 2019.

[31] S. Ahammed, M. Rahman, M. H. Niloy and S. M. H. Chowdhury, "Implementation of machine learning to detect hate speech in bangla language," *2019 8th International Conference System Modeling and Advancement in Research Trends (SMART)*, Moradabad, India, pp. 317–320, 2019.

[32] A. N. M. Jubaer, A. Sayem and M. A. Rahman, "Bangla toxic comment classification (machine learning and deep learning approach)," *2019 8th International Conference System Modeling and Advancement in Research Trends (SMART)*, Moradabad, India, pp. 62–66, 2019.

[33] M. G. Hussain, T. Al Mahmud and W. Akthar, "An approach to detect abusive bangla text," *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, Dhaka, Bangladesh, pp. 1–5, 2018.

[34] M. A. Awal, M. S. Rahman and J. Rabbi, "Detecting abusive comments in discussion threads using naïve bayes," *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pp. 163–167, 2018.

[35] Abdhullah-Al-Mamun and S. Akhter, "Social media bullying detection using machine learning on bangla text," *2018 10th International Conference on Electrical and Computer Engineering (ICECE)*, Dhaka, Bangladesh, pp. 385–388, 2018.

[36] S. C. Eshan and M. S. Hasan, "An application of machine learning to detect abusive bengali text," *2017 20th International Conference of Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, pp. 1–6, 2017.

[37] S. Bhattacharya, S. Singh, R. Kumar, A. Bansal, A. Bhagat, Y. Dawer, B. Lahiri and A. K. Ojha, "Developing a multilingual annotated corpus of misogyny and aggression," *arXiv preprint arXiv:2003.07428*, 2020.

[38] R. Kumar, A. N. Reganti, A. Bhatia and T. Maheshwari, "Aggression-annotated corpus of hindi-english code-mixed data," *arXiv preprint arXiv:1803.09402*, 2018.

[39] N. Ashraf, A. Zubiaga and A. Gelbukh, "Abusive language detection in youtube comments leveraging replies as conversational context," *PeerJ Computer Science*, vol. 7, p. e742, 2021.

[40] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. AbdelMajeed, and T. Zia, "Abusive language detection from social media comments using conventional machine learning and deep learning approaches," *Multimedia Systems*, pp. 1–16, 2021.

[41] Z. Waseem, T. Davidson, D. Warmsley and I. Weber, "Understanding abuse: A typology of abusive language detection subtasks," *arXiv preprint arXiv:1705.09899*, 2017.

[42] T. Mandl, S. Modha, P. Majumder, D. Patel, M. Dave, C. Mandlia and A. Patel, "Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages," in *Proceedings of the 11th forum for information retrieval evaluation*, pp. 14–17, 2019.

[43] R. Nayak and R. Joshi, "Contextual hate speech detection in code mixed text using transformer based approaches," *arXiv preprint arXiv:2110.09338*, 2021.

[44] A. Rodriguez, C. Argueta and Y.-L. Chen, "Automatic detection of hate speech on facebook using sentiment and emotion analysis," *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Okinawa, Japan, pp. 169–174, 2019.

[45] R. Kumar, A. K. Ojha, S. Malmasi and M. Zampieri, "Evaluating aggression identification in social media," in *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pp. 1–5, 2020.

[46] K. Lorenz, M. Latzke, and E. Salzen, *On aggression*. Routledge, 2021.

[47] K. Kumari, J. P. Singh, Y. K. Dwivedi and N. P. Rana, "Bilingual cyber-aggression detection on social media using lstm autoencoder," *Soft Computing*, pp. 1–14, 2021.

[48] A. Shrivastava, R. Pupale and P. Singh, "Enhancing aggression detection using gpt-2 based data balancing technique," *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, pp. 1345–1350, 2021.

[49] F. S. Ansari, M. Barhamgi, A. Khelifi and D. Benslimane, "An approach to detect cyberbullying on social media," in *International Conference on Model and Data Engineering*, Springer, pp. 53–66, 2021.

[50] S. Bharti, A. K. Yadav, M. Kumar and D. Yadav, "Cyberbullying detection from tweets using deep learning," *Kybernetes*, 2021.

[51] S. Thanigaivel, S. Harshan, M. Syed Shahul Hameed and K. Umadevi, "Detection and prevention of cyberbullying using ensemble classifier," in *International Virtual Conference on Industry 4.0*, Springer, pp. 323–333, 2021.

[52] A. Bozyiğit, S. Utku and E. Nasibov, "Cyberbullying detection: Utilizing social media features," *Expert Systems with Applications*, vol. 179, p. 115001, 2021.

[53] N. Lu, G. Wu, Z. Zhang, Y. Zheng, Y. Ren

and K.-K. R. Choo, "Cyberbullying detection in social media text based on character-level convolutional neural network with shortcuts," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 23, p. e5627, 2020.

[54] P. Fortuna, J. Soler and L. Wanner, "Toxic, hateful, offensive or abusive? what are we really classifying? an empirical analysis of hate speech datasets," in *Proceedings of the 12th language resources and evaluation conference*, pp. 6786–6794, 2020.

[55] R. Beniwal and A. Maurya, "Toxic comment classification using hybrid deep learning model," in *Sustainable Communication Networks and Application*, Springer, pp. 461– 473, 2021.

[56] A. G. d'Sa, I. Illina and D. Fohr, "Bert and fasttext embeddings for automatic detection of toxic speech," *2020 International Multi-Conference on: "Organization of Knowledge and Advanced Technologies" (OCTA)*, Tunis, Tunisia, pp. 1–5, 2020.

[57] B. Van Aken, J. Risch, R. Krestel and A. Löser, "Challenges for toxic comment classification: An in-depth error analysis," *arXiv preprint arXiv:1809.07572*, 2018.

[58] Y. Bengio, R. Ducharme, P. Vincent and C. Jauvin, "A neural probabilistic language model," *Journal of machine learning research*, vol. 3, no. Feb, pp. 1137–1155, 2003.

[59] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, pp. 160–167, 2008.

[60] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[61] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[62] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies, pp. 746–751, 2013.

[63] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[64] K. W. Church, "Word2vec," *Natural Language Engineering*, vol. 23, no. 1, pp. 155–162, 2017.

[65] X. Rong, "word2vec parameter learning explained," *arXiv preprint arXiv:1411.2738*, 2014.

[66] J. Pennington, R. Socher and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[67] A. Joulin, E. Grave, P. Bojanowski and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*, 2016.

[68] E. Grave, P. Bojanowski, P. Gupta, A. Joulin and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[69] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[70] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[71] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[72] S.-i. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions," *Neural Networks*, vol. 12, no. 6, pp. 783–789, 1999.

[73] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of machine learning research*, vol. 2, no. Nov, pp. 45–66, 2001.

[74] D. D. Lewis, "Naive (bayes) at forty: The independence assumption in information retrieval," in *European conference on machine learning*, Springer, pp. 4–15, 1998.

[75] A. McCallum, K. Nigam et al., "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, vol. 752, no. 1, Citeseer, pp. 41–48, 1998.

[76] I. Rish et al., "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, pp. 41–46, 2001.

[77] K. P. Murphy et al., "Naive bayes classifiers," *University of British Columbia*, vol. 18, no. 60, pp. 1–8, 2006.

[78] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[79] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan and B. P. Feuston, "Random forest: a classification and regression tool for compound classification and qsar modeling," *Journal of chemical information and computer sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[80] A. Cutler, D. R. Cutler and J. R. Stevens,

"Random forests," in *Ensemble machine learning*, Springer, pp. 157–175, 2012.

[81] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[82] A. Graves, "Long short-term memory," in *Supervised sequence labelling with recurrent neural networks* Springer, pp. 37–45, 2012.

[83] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[84] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," *2017 International Conference on Engineering and Technology (ICET)*, Antalya, Turkey, pp. 1–6, 2017.

[85] P. Kim, "Convolutional neural network," in *MATLAB deep learning*, Springer, pp. 121–147, 2017.

[86] N. Kalchbrenner, E. Grefenstette and P. Blunsom, "A convolutional neural network for modelling sentences," *arXiv preprint arXiv:1404.2188*, 2014.

[87] Y. Huang, W. Wang, L. Wang, and T. Tan, "Multi-task deep neural network for multi-label learning," *2013 IEEE International Conference on Image Processing*, Melbourne, VIC, Australia, pp. 2897–2900, 2013.

[88] V. Ganganwar, "An overview of classification algorithms for imbalanced datasets," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 4, pp. 42–47, 2012.

[89] A. Chanda, D. Das and C. Mazumdar, "Unraveling the english-bengali code-mixing phenomenon," in *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pp. 80–89, 2016.

[90] U. Barman, A. Das, J. Wagner and J. Foster, "Code mixing: A challenge for language identification in the language of social media," in *Proceedings of the first workshop on computational approaches to code switching*, pp. 13–23, 2014.

[91] K. Dinakar, R. Reichart and H. Lieberman, "Modeling the detection of textual cyberbullying," in *fifth international AAAI conference on weblogs and social media*, 2011.

[92] M. J. Denny and A. Spirling, "Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it," *Political Analysis*, vol. 26, no. 2, pp. 168–189, 2018.

[93] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Information processing & management*, vol. 50, no. 1, pp. 104–112, 2014.

[94] S. Vijayarani, M. J. Ilamathi, M. Nithya et al., "Preprocessing techniques for text mining-an overview," *International Journal of Computer Science & Communication Networks*, vol. 5, no. 1, pp. 7–16, 2015.

[95] G. Guibon, M. Ochs and P. Bellot, "From emojis to sentiment analysis," in *WACAI 2016*, 2016.

[96] M. Shiha and S. Ayvaz, "The effects of emoji in sentiment analysis," *Int. J. Comput. Electr. Eng.(IJCEE.)*, vol. 9, no. 1, pp. 360–369, 2017.

[97] R. Sadia, M. A. Rahman and M. H. Seddiqui, "N-gram statistical stemmer for bangla corpus," *CoRR*, vol. abs/1912.11612, 2019. [Online]. Available: `http://arxiv.org/abs/1912.11612`

[98] M. H. Seddiqui, A. A. M. Maruf and A. N. Chy, "Recursive suffix stripping to augment bangla stemmer," in *International Conference Advanced Information and Communication Technology (ICAICT)*, 2016.

[99] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ and S. Khudanpur, "Recurrent neural network based language model," in *Interspeech*, vol. 2, no. 3, Makuhari, pp. 1045–1048, 2010.

[100] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[101] A. Mnih and K. Kavukcuoglu, "Learning word embeddings efficiently with noise-contrastive estimation," in *Advances in neural information processing systems*, 2013, pp. 2265–2273.

[102] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[103] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch and A. Joulin, "Advances in pre-training distributed word representations," *arXiv preprint arXiv:1712.09405*, 2017.

[104] M. Rahman, M. Seddiqui et al., "Comparison of classical machine learning approaches on bangla textual emotion analysis," *arXiv preprint arXiv:1907.07826*, 2019.

[105] N. Banik, M. H. H. Rahman, S. Chakraborty, H. Seddiqui and M. A. Azim, "Survey on text-based sentiment analysis of bengali language," *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, Dhaka, Bangladesh, pp. 1–6, 2019.

[106] S. Karamizadeh, S. M. Abdullah, M. Halimi, J. Shayan and M. javad Rajabi, "Advantage

and drawback of support vector machine functionality," *IEEE 2014 International Conference on Computer, Communication, and Control Technology (I4CT 2014)*, Langkawi, Kedah, Malaysia, pp. 63–65, 2014.

[107] J. Cervantes, X. Li, W. Yu and K. Li, "Support vector machine classification for large data sets via minimum enclosing ball clustering," *Neurocomputing*, vol. 71, no. 4-6, pp. 611–619, 2008.

[108] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST)*, vol. 2, no. 3, pp. 1–27, 2011.

[109] B. McCann, J. Bradbury, C. Xiong and R. Socher, "Learned in translation: Contextualized word vectors," *Advances in neural information processing systems*, vol. 30, 2017.

[110] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[111] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," *arXiv preprint arXiv:1801.06146*, 2018.

[112] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding with unsupervised learning," 2018.

[113] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[114] V. Sanh, L. Debut, J. Chaumond and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[115] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2019.

Md. Nesarul Hoque is an assistant professor at the Bangabandhu Sheikh Mujibur Rahman Science and Technology University in Bangladesh. He works in the department of Computer Science and Engineering. He received Bachelor of Science and Master of Science in Engineering degrees from the University of Chittagong, Bangladesh. At present, he is pursuing a Doctor of Engineering degree from the University of Chittagong, Bangladesh. He has some articles in renowned international conferences. His research interests cover the areas of Machine Learning, Deep Learning, Natural Language Processing, and Information Retrieval. He can be contacted at email: mnhshisir@gmail.com.

Puja Chakraborty is working as a lecturer at the Department of Computer Science and Engineering at Premier University, Bangladesh. She obtained her Bachelor of Science in Engineering from University of Chittagong, Bangladesh. She has several research publications, and her research interest includes Natural Language Processing, Machine Learning and Neural Networks. She can be contacted at email: puja.cse@std.cu.ac.bd.

Md. Hanif Seddiqui has completed his Master of Engineering and Doctor of Engineering degrees from Toyohashi University of Technology, Japan with a number of research awards. He has also completed his post-doctoral fellowship under the European Erasmus Mundus cLINK Programme in the DISP Laboratory, University Lumiere Lyon 2, France, and a short fellowship from KDE Lab, Toyohashi University of Technology, Japan. Currently, he is taking responsibility as a professor in the department of Computer Science and Engineering, University of Chittagong, Bangladesh. He has a few more remarkable contributions on instance matching, Semantic NLP, healthcare and geospatial domain using semantic technology as well. Dr. Seddiqui is a regular member of the conference program committees in the areas of machine learning, data mining, and semantic web as well as a reviewer of some journals. He can be contacted at email: hanif@cu.ac.bd.