Grenze International Journal of Engineering and Technology, June Issue

Contextual Cyberbullying Detection using DeBERTa and LSTM for Sequential Text Analysis

Golve Anusha[1], Atluri Aasritha[2], Pulivarthi Sandhya[3], Pannala Renuka[4] and D. Rajeswara Rao[5]
[1-4]Department of Computer Science and Engineering, Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, India
Email: anushagolve@gmail.com, aasrithaatluri@gmail.com, sandhyapulivarthi6@gmail.com, pannalarenuka13@gmail.com
[5]Professor & Head, Department of Computer Science and Engineering, Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, India
Email: hodcse@vrsec.ac.in

***Abstract*— Cyberbullying has been on the rise over the years due to the growing popularity of social media platforms and communication tools, which pose serious emotional and psychological hazards to users, especially the younger generation. Although various machine learning and deep learning approaches have been proposed for the detection of such toxic content, traditional methods like SVMs or basic neural networks are surface-level approaches and cannot possibly tap into the implicit, context-driven, and subtle nature of abusive language. Even transformer-based approaches like BERT, which are inherently context-driven, may not be able to grasp the subtle nuances of abuse. To overcome these limitations, we propose a hybrid approach to cyberbullying detection that combines the context understanding capabilities of DeBERTa with the ability of LSTM to process sequences. In our approach, DeBERTa is used to generate deep, semantic representations of the input text, which are then further processed by LSTM to identify any sequential patterns and relationships that may exist and potentially indicate cyberbullying. The approach ensures end-to-end detection of cyberbullying and provides a balanced, interpretable, and accurate solution that harnesses the power of deep learning and advanced language modeling.**

***Index Terms*— Cyberbullying Detection, DeBERTa, LSTM, Hybrid Model, Natural Language Processing, Hate Speech.**

I. INTRODUCTION

The internet has simplified communication like never before. However, it has also posed significant difficulties, and one of the most concerning issues is cyberbullying. As the population continues to grow, popularity of social networking sites, messaging applications and internet discussion boards, instances of online harassment, especially among adolescents and young adults, have surged significantly, creating a permanent effect on the emotional and mental health of the victims. According to [1], a survey conducted by Cyberbullying Research Center in the United States reports that, 34% of the ages between 12 and 17 have experienced cyberbullying.
Cyberbullying is hard to recognize due to its ability to stay understated. It isn't always straightforward, featuring clear messages of harassment and intimidation, but instead via irony, inflection, setting, or perhaps suggested messages. Traditional techniques for recognizing cyberbullying, like keyword identification or basic neural networks, fail to detect these fine patterns.

Grenze ID: 01.GIJET.12.2.244_26

Recently, the study [2] developed a cyberbullying detection system by improving the features and selecting the best combination of models. To understand the contextual understanding of the words, [3] has used models such as DeBERTa and BERT, which increase the performance of the models. Although these models may contextually understand the words, sequentially analyzing the whole text is even more necessary to understand the complete tone of the sentence, even if the content of the sentence is flagged as toxic.

To address this issue, our project suggests a combined strategy that merges DeBERTa with LSTM. DeBERTa is utilized to derive profound contextual and semantic insights from text, and LSTM is used to examine sequential relationships and structures in phrases. This method assists the system in recognizing both overt and subtle occurrences of cyberbullying. Additionally, the system features an intuitive interface, allowing users to input text or upload conversation records for immediate analysis, thereby aiding in the establishment of a secure digital space.

II. Related Work

TABLE I. COMPARISON BETWEEN RECENT CYBERBULLYING APPROACHES AND PROPOSED MODEL

Title	Model	Datasets	Advantage	Disadvantage
Siddhartha et al.,2022	SMSDA	Social media	Improved text representation	Limited contextual and sequential understanding
Islam et al., 2020	SVM+TF-IDF	Twitter/Facebook	Used NLP feature extraction with ML classifiers to detect bullying text	Depends on manual feature extraction
Kumar et al., 2024	Transformers (BERT, DeBERTa)	Social media	Improved contextual understanding of abusive content	Does not model sequential patterns effectively
Aldhyani et al., 2022	Deep Neural Network	Online Text	Better performance compared to traditional ML models	Limited contextual semantic understanding
Ogunleye et al., 2023	RoBERTa	Twitter+Form-spring	Achieved strong performance	Focuses only on transformer-based models.
Aggarwal et al., 2024	BERT + SVM	Social media	Improves cyberbullying detection using contextual understanding and classification	Limited sequential understanding
Alikhashashneh et al., 2025	Transformers (BERT, GPT-2, T5)	Cyberbullying Dataset	High performance with contextual understanding of text	No sequential learning
Proposed model	DeBERTa + LSTM	Cyberbullying Dataset	Improves cyberbullying detection model by performing both contextual and sequential analysis of text	More computational complexity

Manowarul et al. [2] proposed a study using feature extraction methods such as Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) were used to represent text messages. Classifiers like Naive Bayes, Support Vector Machine (SVM), Decision Tree, and Random Forest are trained using these features. SVM with TF-IDF showed better accuracy in detecting bullying content. However, this approach mainly focused on manually designed text features and cannot fully capture contextual meaning of text.

Kumar et al. [3] proposed a method for cyberbullying detection using DeBERTa and BERT. These models were used to improve contextual understanding of social media text and increase performance of the model in detecting abusive content. This study shows that, using these transformer architectures, it is easier to capture the contextual understanding between words. Despite these advantages, this method may not effectively capture the sequential patterns that show the harmful language present in the sentence.

Siddhartha et al. [4] proposed a method for detecting cyberbullying on social media using a Semantic-Enhanced Marginalized Denoising Auto-Encoder (SMSDA) which is used to improve the representation of text messages so that bullying can be detected easily. Special techniques were used to learn useful features from social media. Although the model is effective, it lacks contextual understanding and sequential relationships within sentences.

Aldhyani et al. [5] introduced a cyberbullying system which used neural network architectures such as CNN and BiLSTM which are used to learn patterns of harmful content from textual data. Compared to traditional approaches, this approach showed improved performance in detecting abusive text. However, this approach still faced challenges in detecting deeper contextual semantics in bullying content.

Ogunleye et al. [6] explored the advantages of using large language models for cyberbullying detection on social media platforms. A new dataset was built by combining the data from online sources, which is used to evaluate

transformer-based models. In this study, several models were compared and the results showed that RoBERTa achieved better performance compared to other models. However, this method mainly focuses on a transformer-based approach, but did not use a hybrid approach that combines contextual and sequential learning.

Aggarwal et al. [7] suggested a cyberbullying detection system using BERT and Support Vector Machine. This novel approach combines BERT and SVM as an ensemble model which is useful for multiclass classification of harmful social media content. BERT is used to generate contextual representations which are used by SVM to increase the performance. Explainable technique like SHAP is used to justify the model predictions. However, this approach mainly discusses cyberbullying detection through feature extraction and may not be able to provide sequential analysis effectively.

Alikhashashneh et al. [8] proposed a study using transformers like BERT, GPT-2 and T5 for automated cyberbullying detection. In this transformer framework, the models first undergo through a common preprocessing pipeline and then are evaluated using same dataset for fair comparison. In the final analysis, text-to-text transfer transformer (T5) achieved best performance when compared to BERT AND GPT-2. However, longer range text analysis is not possible through this framework as it entirely depends upon transformer models. Table I shows the comparison of recent cyberbullying approaches with proposed model.

III. PROPOSED METHODOLOGY

The main objective of this proposed system is to increase the effectiveness in detecting bullying or abusive content. The process flow diagram presented in Figure 1 graphically demonstrates the sequential steps within the operation of the system, ensuring clear comprehension of the overall process. The suggested framework for detecting cyberbullying features multiple essential elements working together to ensure precise categorization. The process starts with data preprocessing, which includes cleaning the text data by removing noise, converting to lowercase, tokenizing, and normalizing sentence lengths through padding or truncation. The model subsequently utilizes DeBERTa, an architecture based on transformers, to generate contextual embeddings that convey deep semantic meanings of the embeddings derived from context words. These embeddings serve as inputs for an LSTM network, which identifies sequential and hidden features, allowing the model to understand the order and deeper significance of sentences.

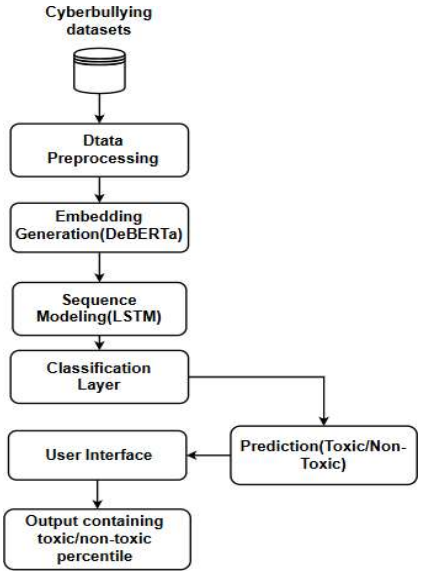

Figure 1. Process flow of proposed cyberbullying detection model

A. Dataset Collection

The dataset for the proposed system is obtained from Kaggle, which contains texts from various websites, platforms, and social media sites. The dataset pertains to the identification of damaging, antagonistic, prejudiced, sexist and additional damaging material in user-created texts. It encompasses comments, writings, and discussions collected. This dataset consists of various data from different sources like Twitter, YouTube, which has both toxic and non-toxic types of data.

B. Dataset Description

The dataset for this proposed model is obtained from Kaggle, which is divided into various categories. It contains text data collected from multiple online platforms like Twitter, YouTube, Wikipedia discussion pages, and various social media channels. The dataset includes comments created by users that are marked to show if they include cyberbullying or abusive content. It includes eight CSV files gathered from different online sources, including Twitter, YouTube and Wikipedia discussion threads. For model training and evaluation, a subset of 19,531 samples was used. This allows the model to train and accurately differentiate between bullying or non-bullying text.

C. Data Preprocessing

- Text Cleaning: Removing unwanted elements like special characters, hyperlinks, and extra whitespace.
- Lowercasing: Converting text to lowercase to ensure uniformity.
- Tokenization: Breaking sentences into tokens for processing.
- Padding and Truncation: Making all sequences of a fixed length, which is necessary for uniform model input

D. Text Encoding with Transformer Tokenizer

Transformers require input data to be in numerical form. Unlike label encoding, which is applied to categorical variables, text data needs to be tokenized and converted into dense numerical vectors that reflect semantic information. This is accomplished using pretrained tokenizers such as those found in the DeBERTa model. These tokenizers split text into subword tokens, transform them into matching numerical token IDs, and produce additional data such as attention masks for padding as shown in Algorithm 1.

Algorithm 1: Text Encoding with Transformer Tokenizer

1: Output: Encoded dataset E
2: for each text sample ti ∈ T do
3: tokensi ← Tokenizer(ti)
4: input_dsi ← Tokenizer.encode(ti)
5: attention_maski ← padding
6: Ei ← {input_idsi, attention_maski}
7: End for
8: return E = {E1, E2, . . . , En}

E. Embedding Generation (DeBERTa)

The modified text data is then fed into the DeBERTa transformer model to produce contextual semantic representations. In contrast to conventional embedding techniques, transformer-based embeddings grasp profound contextual connections among words by taking into account bidirectional context. DeBERTa generates high-dimensional contextual embeddings that capture semantic significance beyond mere surface-level text patterns. These embeddings improve the representational quality of the input data and increase the model's ability to detect toxic or abusive language efficiently.

The DeBERTa tokenizer processes the input text by segmenting it into tokens and incorporating special tokens like [CLS] to indicate the full sequence. The tokenized input is then provided to the DeBERTa model, generating hidden states for each token. The hidden state for the [CLS] token is generated from this, serving as a compact, high-dimensional representation that captures the overall significance and context of the input text. The CLS embedding is subsequently input into an LSTM classifier that predicts from the embedding, offering insights into the content's toxicity and generating a confidence score for that toxicity as in Algorithm 2. The model is capable of grasping nuanced contextual signals and sequential patterns, improving its ability to detect offensive or harmful language.

Algorithm 2: Extract CLS Token Embedding from DeBERTa

1: Input: Text sample text
2: Output: CLS embedding vector
3: Tokenize text using DeBERTa tokenizer with:
4: truncation = True
5: padding = max_length
6: max_length = 64
7: Move tokenized tensors to the appropriate device
8: Disable gradient computation using torch.no_grad()

9: Pass input_ids and attention mask to the DeBERTa model
10: Extract first token of the last hidden state
11: cls_embedding ← outputs.last_hidden_state[:,0,:]
12: return cls_embedding

F. Feature Extraction (LSTM) and Training

LSTM is used for the sequential analysis of the tokens, which are generated after the embedding generation process using DeBERTa. The embeddings generated by the DeBERTa model are input into a Long Short-Term Memory (LSTM) network for consecutive feature extraction and categorization.
The LSTM network carries out these functions:

- Manages Sequential Embeddings: Recognizes the arrangement of words and the context's progression in the text.
- Captures Long-Term Dependencies: Effectively models semantic connections among far-apart tokens.
- Extracts Hidden Features: Acquires meaningful representations that aid in precise classification.

This combination guarantees that both profound contextual comprehension and sequential relationships are accurately captured. After this process, the system not only understands the contextual meaning of the words but also the hidden tone in the sentence, which helps in detecting the content as toxic or non-toxic with respect to the sequential analysis of token embeddings. Algorithm 3 outlines the feature extraction and classification process using LSTM.
Algorithm 3: Classify with LSTM Using DeBERTa Embedding
1: function classify_with_lstm(cls_embedding):
2: Reshape cls embedding to add sequence dimension if needed
3: cls_embedding ← cls_embedding.unsqueeze(1)
4: Feed embedding to LSTM layer
5: (hn, _) ← lstm(cls_embedding)
6: Get last hidden state hidden ← hn[−1]
7: Forward through fully connected layer
8: logits ← fc(hidden)
9: return logits (raw class scores)
10: end function

G. User Interface Deployment using Gradio

The User Interface is deployed using Gradio to allow users to access the system. This interface helps users to:

- Upload text data.
- Check whether the input is toxic or non-toxic.
- View the probability scores for toxic and non-toxic classification.
- Determine whether the text is flagged as cyberbullying.

H. Experimental Setup

The proposed methodology is executed using PyTorch framework. DeBERTa-base is used to create contextual embeddings from the input text which were then analyzed by LSTM classifier for cyberbullying detection. 8 epochs are used to train the model, using Adam optimizer with batch size 16. Its performance is evaluated using accuracy, precision, recall, F1-score and confusion matrix.

IV. Result and Analysis

A. Performance Analysis

- Confusion Matrix: Figure 2 displays the confusion matrix which illustrates the classification performance of the model. From 13,021 actual "Non-Toxic" cases, 12,370 were correctly recognized as "Non-Toxic", while among 6,510 actual "Toxic" cases, 6,185 were correctly classified as "Toxic", leading to 325 incorrectly labelled as "Non-Toxic". The errors in the classification can also be reasoned with spelling variations, special expressions found in social media content, contextual dependencies, ambiguous language and sarcasm. Despite these challenges, the model achieved the total accuracy of 95%, pointing it effectiveness in distinguishing toxic content from non-toxic content.
- Tabular Representation of Metrics: TABLE II provides an overall summary of the model's performance across two classes "Toxic" and "Non-Toxic". The "Non-Toxic" class achieved a precision of 0.97, recall of

0.95 and F1-score of 0.96. The "Toxic" class achieved a precision of 0.90, recall of 0.95 and F1-score of 0.92, indicating the effective detection of harmful content. The high recall indicates the model's ability to identify most of the toxic instances, which is important in cyberbullying detection.

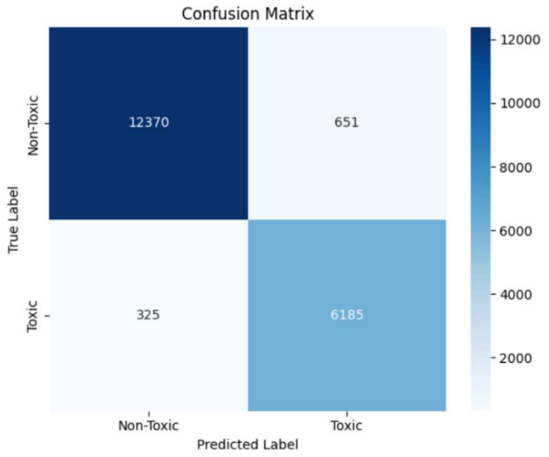

Figure 2. Confusion Matrix

As a well-balanced model, the overall validation accuracy is 95%. The macro average precision, recall and F1-score were 0.94, 0.95 and 0.94 respectively and the weighted averages were 0.95, 0.95 and 0.95. The results show the balanced performance of "Toxic" and "Non-Toxic" classes achieving strong results for precision, recall and F1-scores, ensuring reliable classification while minimizing misclassification errors. Higher precision scores of toxic and non-toxic classes indicates its less error rates in distinguishing them. The high recall scores for both classes shows the models ability to detect most of the instances in toxic and non-toxic classes. The higher results show the capability of combining DeBERTa's contextual understanding and LSTM's sequential learning abilities which are useful in improving cyberbullying detection performance.

TABLE II. CLASSIFICATION METRICS FOR CYBERBULLYING DETECTION

Class	Precision	Recall	F1-score	Support
Non-Toxic	0.97	0.95	0.96	13021
Toxic	0.90	0.95	0.92	6510
Accuracy	0.95 (on 19531 samples)			
Macro Avg	0.94	0.95	0.94	19531
Weighted Avg	0.95	0.95	0.95	19531

B. User Interface

Our suggested approach seeks to address cyberbullying more intelligently and promptly. By combining DeBERTa's contextual understanding with LSTM's capability to track conversational dynamics, our system can identify not only blatant insults but also nuanced bullying that may be concealed in sarcasm or coded language. To evaluate the model performance in real-time, a simple User Interface has been built using Gradio, where users can submit any text and immediately determine if it is toxic, accompanied by a percentage indicating the system's confidence level. This facilitates the early detection of harmful messages—regardless of whether they appear in chats, comments, or social media platforms. Figure 3 shows the User Interface where the model classifies a very common text 'hi, how are you?' as non-toxic content, indicating no cyberbullying with 98.46% of non-toxic score. Figure 4 shows the User Interface where the model classifies an abusive text as the Toxic content, indicating cyberbullying with 98.91% of toxic score.

Figure 3. Prediction by the model as Non-Toxic in Gradio

Although various existing models use different ways to improve the cyberbullying detection model, our suggested system overcomes their limitations by contextual understanding followed by sequential analysis of the text, which proves its ability to understand hidden meanings present in the text.

Figure 4. Prediction of the model as Toxic in Gradio

C. Discussion

The proposed model achieved an overall accuracy of 95%. The high accuracy value indicates the model's ability to accurately distinguish between toxic and non-toxic classes. The high precision and recall values show model's capability in making correct predictions. This performance of the model is due to the combined strengths of DeBERTa and LSTM. DeBERTa understands the context of the sentence and converts them into meaningful vectors, LSTM then captures the relationships among these representations. DeBERTa and LSTM together enables the model to better understand if the given content is harmful or not and classify them as toxic or non-toxic. These results suggests that the proposed model can be used in real world cyberbullying detection applications.

V. CONCLUSION

This paper proposes a hybrid approach for cyberbullying detection, which combines contextual understanding and sequential analysis using DeBERTa and LSTM. The model achieved a validation accuracy of 95%, with precision, recall and F1-score of 0.90, 0.95 and 0.92 for Toxic instances, which demonstrates the model's effectiveness in detecting harmful content accurately. To enable real-time classification for users, the proposed system also provided access to the User Interface. Overall, this proposed method achieved a good balance between accuracy, precision, and recall, making it useful for real-world applications. The main advantage of this project is that, using the combination of DeBERTa and LSTM, understanding the deeper tone and meaning of a sentence can be effectively achieved, regardless of whether harmful words are present. Compared to traditional models and methodologies, this solution is useful for accurately detecting whether the given text is cyberbullying or not. In the future, this system can also be improved by including emojis and images to enhance cyberbullying detection.

REFERENCES

[1] "Cyberbullying," Encyclopedia Britannica, Mar. 28, 2026. https://www.britannica.com/topic/cyberbullying

[2] M. M. Islam, M. A. Uddin, L. Islam, A. Akter, S. Sharmin, and U. K. Acharjee, "Cyberbullying Detection on Social Networks Using Machine Learning Approaches," in Proc. IEEE Asia-Pacific Conf. Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 2020, pp. 1–6, doi:10.1109/CSDE50874.2020.9411601

[3] Y. Kumar et al., "Bias and Cyberbullying Detection and Data Generation Using Transformer Artificial Intelligence Models and Top Large Language Models," Electronics, vol. 13, no. 17, Art. no. 3431, 2024, doi:10.3390/electronics13173431

[4] K. Siddhartha, K. R. Kumar, K. J. Varma, M. Amogh, and M. Samson, "Cyber Bullying Detection Using Machine Learning," in Proc. 2nd Asian Conf. Innovation in Technology (ASIANCON), Ravet, India, 2022, pp. 1–4, doi:10.1109/ASIANCON55314.2022.9909201

[5] T. H. H. Aldhyani, M. H. Al-Adhaileh, and S. N. Alsubari, "Cyberbullying Identification System Based Deep Learning Algorithms," Electronics, vol. 11, no. 20, Art. no. 3273, 2022, doi:10.3390/electronics11203273

[6] Ogunleye and B. Dharmaraj, "The Use of a Large Language Model for Cyberbullying Detection," Analytics, vol. 2, no. 3, pp. 694–707, 2023, doi:10.3390/analytics2030038

[7] P. Aggarwal and R. Mahajan, "Shielding Social Media: BERT and SVM Unite for Cyberbullying Detection and Classification," Journal of Information Systems and Informatics, vol. 6, no. 2, pp. 607–623, Jun. 2024, doi: 10.51519/journalisi.v6i2.692

[8] E. Alikhashashneh, H. Alsawan, K. M. O. Nahar, N. Shatnawi, A. Almomani, M. Alauthman, S. Bansal, and V. S.-H. Pan, "Unified Transformer Framework for Automated Cyberbullying Detection," International Journal of Cloud Applications and Computing, vol. 15, no. 1, Apr. 2025, doi: 10.4018/IJCAC.386166