Cyberbullying Detection in Code-Mixed Hinglish Language

Rahul
Artificial Intelligence and Data Science
IIITDM Kurnool
Kurnool, India
121ad0036@iiitk.ac.in

Dr. K. E. Srinivasa Desikan
Department of Computer Science
IIITDM Kurnool
Kurnool, India
srinivasadesikan@iiitk.ac.in

***Abstract*—Cyberbullying has emerged as a significant concern in India, particularly with the rise of social media platforms where code-mixed languages like Hinglish (a blend of Hindi and English) dominate online communication. The unique linguistic characteristics of Hinglish, including its syntax, transliteration, and contextual variations, pose considerable challenges to existing detection systems, which are often optimized for monolingual or widely-used international languages.**

This research addresses the limitations of traditional approaches by leveraging advanced Natural Language Processing (NLP) techniques and machine learning models to detect and classify cyberbullying in Hinglish text. A comprehensive methodology is proposed, involving data preprocessing tailored to Hinglish, feature extraction using text vectorization techniques, and machine learning algorithms for classification.

The study underscores the importance of localized solutions for multilingual and culturally specific contexts. By focusing on India's linguistic diversity, this work aims to improve online content moderation and provide a foundation for real-time cyberbullying detection systems that can adapt to other code-mixed languages. The findings contribute to safer digital environments, supporting the mental well-being of online users in India.

I. Introduction

The rapid digital transformation in India has brought with it an alarming rise in cyberbullying, particularly among young internet users. India, as one of the world's largest internet markets, witnesses an expansive user base, with over 700 million active users as of 2023 [1]. Among these, a significant portion comprises adolescents and young adults, who are particularly vulnerable to online harassment. Studies indicate that 85% of children in India have faced cyberbullying, a figure substantially higher than the global average of 37% among adolescents aged 12 to 17 [2]. Additionally, platforms such as Instagram, WhatsApp, and Facebook, which dominate the Indian social media space, often serve as hotspots for such activities, with 60% of users witnessing or experiencing bullying [3].

The psychological toll of cyberbullying is severe. Victims often report heightened levels of anxiety, depression, and suicidal ideation. For example, nearly 70% of victims of cyberbullying report severe emotional distress, affecting their academic performance, personal relationships, and overall mental well-being [4]. In a country like India, where mental health resources are still underdeveloped and stigmatized, the consequences of cyberbullying can be long-lasting and devastating [5].

A unique challenge in addressing cyberbullying in India stems from the linguistic diversity of its online communication. Hinglish, a code-mixed language blending Hindi and English, is widely used across social media platforms. The informal, adaptive, and context-dependent nature of Hinglish presents significant hurdles for traditional detection systems, which are primarily designed for single-language datasets [6]. This linguistic complexity allows cyberbullies to use slang, sarcasm, and nuanced expressions to evade detection. Despite the scale of the problem, most existing solutions remain inadequate, as they fail to address the nuances of code-mixed languages [7].

This research is driven by the need to bridge these gaps. By leveraging machine learning and natural language processing (NLP) techniques, we aim to develop robust systems capable of accurately detecting cyberbullying in Hinglish text. This study focuses on creating an adaptable framework that not only identifies harmful content but also aligns with the linguistic and cultural specificity of India. The ultimate goal is to provide a technological intervention that mitigates the impact of cyberbullying, fostering a safer and more inclusive online environment.

In addition to its psychological impact, the economic and social repercussions of cyberbullying cannot be overlooked. Studies suggest that workplaces and educational institutions often face disruptions when cyberbullying spills into offline environments. Moreover, the pervasive nature of online harassment, particularly on platforms heavily used in India, such as WhatsApp and Twitter, undermines the sense of safety for users [8]. With over 65% of Indians relying on the internet for communication, education, and professional development, unchecked cyberbullying erodes trust in these digital ecosystems and hinders digital inclusivity [9].

The legislative framework in India to combat cyberbullying remains underdeveloped. While laws like Section 66A of the IT Act (struck down in 2015) and subsequent amendments address online harassment, enforcement is inconsistent, and gaps in addressing the unique challenges posed by Hinglish further exacerbate the issue [10]. Many victims remain unaware of their legal rights or lack access to mechanisms for reporting and resolving cyberbullying incidents. This regulatory lag highlights the urgent need for technological solutions that complement policy efforts and provide real-time detection and intervention mechanisms.

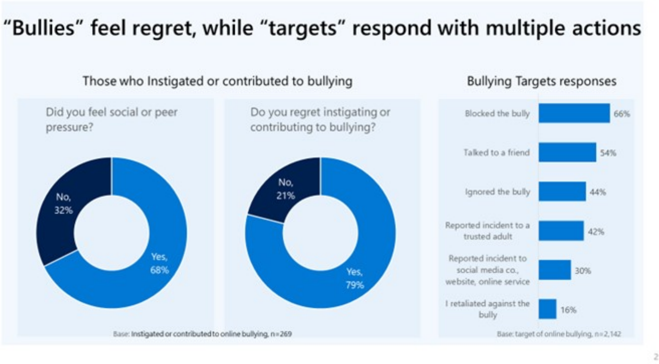


Fig. 1. Online Bullying Study (2020) .

Furthermore, Hinglish, as a code-mixed language, is evolving rapidly with the advent of memes, hashtags, and internet slang. These informal expressions often convey complex emotions, sarcasm, or implicit threats that traditional systems struggle to decode. For instance, phrases combining Hindi script with English alphabets or transliterations can drastically alter context, making it harder for static models to identify harmful content [17]. This underscores the need for adaptable, context-aware models that can keep pace with linguistic innovations.

This study not only aims to address the technical challenges of cyberbullying detection in Hinglish but also sheds light on the societal implications of such technological advancements. By integrating insights from linguistic, cultural, and psychological perspectives, the research aspires to create a comprehensive framework. The ultimate goal is to provide actionable tools for policymakers, social media platforms, and educational institutions, enabling them to foster safer digital spaces while respecting the cultural and linguistic diversity of India.

II. **Literature Survey**

The increasing prevalence of cyberbullying in India and its intersection with Hinglish presents a complex linguistic and social challenge. This section reviews existing studies on cyberbullying detection and code-mixed language processing, particularly Hinglish, to contextualize the research.

A. Cyberbullying Detection Methods

Cyberbullying is a pervasive issue on social media platforms, where individuals, especially teenagers, are increasingly subjected to online harassment and aggression. Researchers have devised various methods to detect cyberbullying through machine learning (ML) and natural language processing (NLP). A notable study in this area by [11] focuses on detecting cyberbullying in resource-constrained languages using transformer-based models like BERT. This model aims to bridge the gap in resources for languages with fewer labeled data for training. The approach fine-tuned a transformer model to detect abusive behavior in low-resource languages, achieving an accuracy of 91.6%. Despite this, the study acknowledged that the scarcity of labeled data for underrepresented languages limits the generalizability and fine-tuning of such models in these contexts.

Similarly, [12] proposed a generative approach for detecting cyberbullying in Hinglish, a code-mixed language commonly used in online social platforms by speakers of Hindi and English. They introduced the BullyGen model, which integrates explainable AI (XAI) techniques to make the model's decisions more transparent. BullyGen achieved an accuracy of 82.95% but faced challenges in handling the complex linguistic structures inherent in Hinglish. This research highlights the growing need for models that can work effectively with the hybrid nature of many modern languages, which are used in informal settings on social media platforms.

Another study by [13] utilized text mining and machine learning to classify cyberbullying content on social media. The authors applied preprocessing techniques like tokenization and stemming and used models such as SVM and Naive Bayes for classification. Though this study was able to detect cyberbullying, it acknowledged the limitations of traditional text-mining methods, particularly when it comes to detecting subtle or less overt forms of bullying.

The challenge of cyberbullying detection lies not only in the identification of harmful content but also in the diversity of languages and forms of abuse found on social media platforms. While many existing studies focus on detecting cyberbullying in English and other widely spoken languages, few solutions exist for underrepresented languages or those with complex linguistic features.

B. Challenges of Code-Mixed Languages

Code-mixed languages, such as Hinglish (a combination of Hindi and English), pose unique challenges in cyberbullying detection due to their linguistic complexity. In [14], the authors address the difficulties of detecting hate speech and aggression in Hindi-English mixed social media posts. They achieved an accuracy of 85% using machine learning models, but the study highlighted several key challenges, such as handling the syntactic and semantic variations introduced by the use of two languages within a single sentence.

The ability to detect subtle nuances in code-mixed text requires advanced techniques that go beyond traditional NLP methods. This is a major reason why existing models for monolingual data fail to generalize well to code-mixed languages. Code-mixing creates additional challenges in segmentation, tokenization, and semantic analysis. In this context, [13] emphasized that handling noisy text in code-mixed datasets is a significant hurdle. The model had to adapt to the specific characteristics of code-mixed languages, where the structure of sentences changes based on the combination of languages, making it difficult for conventional models to parse and understand the intent behind words.

Another major difficulty in working with code-mixed languages is the lack of annotated datasets for training purposes. Annotating large datasets of code-mixed text requires bilingual experts who can accurately identify whether a post is cyberbullying, which is resource-intensive and time-consuming. The limited availability of high-quality labeled data for languages like Hinglish often forces researchers to use less efficient data augmentation techniques or transfer learning approaches, which come with their own set of limitations.

C. Multilingual Hate Speech and Cyberbullying Detection

Multilingualism presents a significant challenge for cyberbullying detection systems, especially when dealing with hate speech or offensive content that appears in multiple languages. The detection of cyberbullying and hate speech in various languages requires models that can handle linguistic diversity and code-switching seamlessly. In [15], deep learning models such as LSTM and BERT were used to detect hate speech across multiple languages, including Hindi, Bengali, and English. The authors employed word embeddings and attention mechanisms to account for the differences in vocabulary and syntax between languages. This approach helps in building a more generalized model capable of understanding context and semantics in different languages. However, the study acknowledged that the lack of large multilingual datasets for hate speech detection is a significant limitation.

A related study, [16], tackled the challenge of detecting hate speech using a multi-modal deep learning approach. In this study, the authors combined text and image data to detect hate speech on social media platforms, specifically focusing on languages like Hindi and Bengali. This model utilized both NLP techniques for text analysis and computer vision (CV) for image processing, allowing it to detect offensive content that might not be easily recognized through text alone. By using a multi-modal approach, the study aimed to improve the accuracy of hate speech detection. However, the researchers found that the model struggled with noisy and unstructured data, which often occurs in social media environments.

While these studies have advanced the field of multilingual hate speech detection, they also highlight several challenges, including the need for more diverse datasets and the difficulty of integrating data from multiple modalities, such as images and text. Furthermore, the imbalance between hate speech and non-hate speech instances in many datasets makes it difficult to train robust models that perform well in real-world scenarios.

D. Recent Advances in Cyberbullying Detection

Recent advancements in cyberbully detection are focusing on deep learning models and multi-modal approaches to improve classification accuracy. In [11], transformer models were used to detect cyberbullying in low-resource languages. Transformer-based models, such as BERT, have been proven effective in many NLP tasks due to their ability to capture context and long-range dependencies in text. These models, when fine-tuned, can achieve high accuracy even in underrepresented languages. However, the study acknowledged that the primary limitation of using transformer models is the scarcity of labeled data, which makes fine-tuning these models on small datasets particularly challenging.

Additionally, [14] explored the combination of deep learning with multi-modal data to detect hate speech and aggression on social media platforms. Their approach incorporated both text and image data to classify social media content as either clean or offensive. While the model showed promising results, particularly in the identification of aggressive behavior, it also faced challenges related to the noisy nature of the data and the difficulty of balancing the dataset between hate speech and non-hate speech instances.

The integration of deep learning with explainable AI (XAI) is another trend in recent research on cyberbullying detection. In [12], the BullyGen model was developed to provide transparent and interpretable results in detecting cyberbullying in Hinglish. This generative model offers insight into why a particular piece of content is classified as bullying, thus addressing concerns about the ”black-box” nature of many deep learning models. The interpretability of the model is crucial, especially in sensitive applications like cyberbullying detection, where understanding the rationale behind a classification decision is important for building trust in automated systems.

E. Gaps in Existing Research

1) **Dataset Availability:** Hinglish datasets are limited in size and scope, often focusing on specific platforms or user demographics.
2) **Context-Awareness:** Current systems struggle with the nuanced context of Hinglish, particularly sarcasm, mixed scripts, and transliterations.
3) **Cultural Relevance:** Many models fail to account for the cultural and psychological dimensions of cyberbullying in India, limiting their practical applicability.

III. Research Methodology

This section outlines the systematic approach adopted for detecting cyberbullying in Hinglish (a mix of Hindi and English) and other resource-constrained languages. The methodology incorporates data collection, preprocessing, model training, evaluation, and deployment.

A. Data Collection

The primary step in the research involved gathering datasets for detecting cyberbullying, particularly in Hinglish (code-mixed Hindi-English) and other resource-constrained languages. The data sources included:

- **Social Media Platforms:** Platforms like Twitter, Instagram, and YouTube provided user-generated content annotated for hate speech or cyberbullying.
- **HASOC 2019 Dataset** Focuses on detecting hate speech and offensive language in Hinglish, targeting abusive content in social media posts.
- **Hinglish Offensive Content Dataset (ICON):** Contains Hindi-English mixed social media posts aimed at offensive language detection.
- **Annotation:** Where required, datasets were manually labeled with the assistance of linguistic experts.

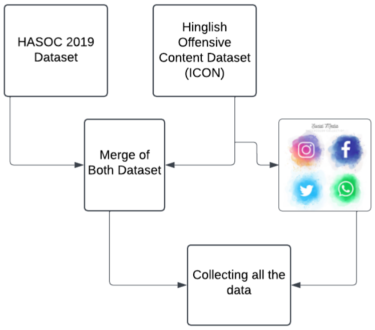


Fig. 2. Dataset.

B. Data Preprocessing

Data preprocessing is a critical step to transform raw, unstructured text into a clean and structured format suitable for analysis. The pipeline includes the following steps:

- **Text Cleaning:** This step removes noise and irrelevant elements such as:
  - URLs (e.g., `https://example.com`).
  - Special characters, emojis, and hashtags.
  - Converts all text to lowercase for uniformity.
- **Transliteration:** Hinglish content written in Devanagari script is converted to Latin script for processing. For instance: नमस्कार

  Devanagari: "क्या हाल है"
  Hinglish: *”kya haal hai”*
- **Tokenization:** This step splits sentences into individual words or tokens, allowing word-level analysis. It is mathematically represented as:

$$T = \{t_1, t_2, \ldots, t_n\}$$

  where T represents the sequence of tokens. For example:

  Input: *”kya haal hai”*
  Tokens: *{”kya”, ”haal”, ”hai”}*
- **Stopword Removal:** Commonly used words, such as ”is,” ”hain,” and ”ka,” are removed to reduce computational overhead and focus on meaningful words. For example:

  Input Tokens: *{”kya”, ”haal”, ”hai”}*
  After Stopword Removal: *{”kya”, ”haal”}*
- **Stemming and Lemmatization:** These steps reduce words to their base forms:
  - **Stemming:** Truncates words to their root form.
    Example: *”running”* → *”run”*
  - **Lemmatization:** Converts words to their dictionary form.
    Example: *”better”* → *”good”*

  This transformation is represented as:

$$t'_i = \text{Stem}(t_i) \quad \text{or} \quad t'_i = \text{Lemma}(t_i)$$

  where t'_i is the processed token.

The preprocessing pipeline ensures that the data is cleaned, structured, and optimized for further feature extraction and machine learning model training. .

C. Feature Engineering and Selection

Feature engineering transforms raw text data into structured numerical representations that models can process. This section details the methodologies employed:

- **Count Vectorization:** Count Vectorization represents text as a vector of word frequencies without considering the order or semantics of words:

$$\mathbf{v} = [\text{count}(w_1), \text{count}(w_2), \ldots, \text{count}(w_m)]$$

  Here, w_i is the i-th word in the vocabulary, and $\text{count}(w_i)$ represents the frequency of that word in the document. This method captures text structure effectively for small datasets but can result in sparse matrices for large vocabularies.
- **Bag-of-Words (BoW):** Similar to Count Vectorization, BoW uses binary or frequency-based encoding for words but does not capture relationships or context. BoW is represented as:

$$\mathbf{v} = [\text{binary}(w_1), \text{binary}(w_2), \ldots, \text{binary}(w_m)]$$

  where $\text{binary}(w_i)$ indicates the presence or absence of a word in the document.

- **TF-IDF (Term Frequency-Inverse Document Frequency):** TF-IDF assigns weights to words based on their frequency and importance in the corpus:

$$\text{TF-IDF}(w) = \text{TF}(w) \times \log\left(\frac{N}{\text{DF}(w)}\right)$$

  where:
  - $\text{TF}(w)$: Term frequency of w in a document.
  - N: Total number of documents.
  - $\text{DF}(w)$: Document frequency of w (number of documents containing w).
- **Word Embeddings:** Dense vector representations encode semantic meaning and relationships. Using Word2Vec or GloVe, words are embedded as:

$$\mathbf{v}(w) = [x_1, x_2, \ldots, x_d]$$

  where d is the embedding dimension. This allows for capturing semantic similarities and analogies like:

$$\text{"king"} - \text{"man"} + \text{"woman"} \approx \text{"queen"}.$$

- **Feature Selection:** After feature extraction, selecting the most relevant features is crucial:
  - *Chi-Square Test:* Measures the independence between features and target labels:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

    where O_i is the observed frequency and E_i is the expected frequency.
  - *Mutual Information:* Quantifies the information shared between a feature and the target variable:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) \log\left(\frac{P(x,y)}{P(x)P(y)}\right).$$

  - *Principal Component Analysis (PCA):* Reduces dimensionality by projecting data onto a set of uncorrelated components:

$$Z = XW,$$

    where W represents the eigenvectors of the covariance matrix.
  - *Recursive Feature Elimination (RFE):* Iteratively removes features with the lowest importance based on model evaluation.

D. Deployment and Automation

To ensure robust and scalable deployment:

- A preprocessing pipeline integrates feature extraction methods like TF-IDF and embeddings.
- Automation ensures consistent data transformations for training and inference.
- Cloud-based services, such as AWS or Azure, are used for scalable deployment.

This setup enhances reproducibility and performance across diverse user inputs.

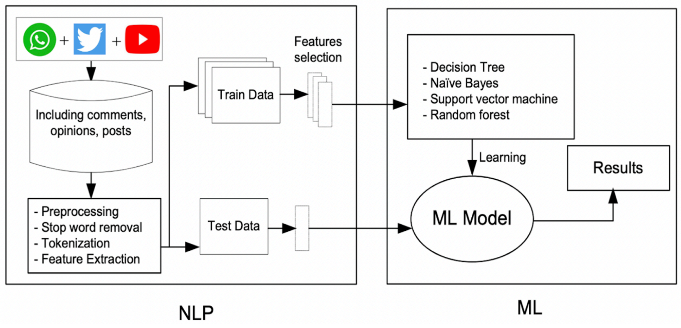


Fig. 3. Functional Block Diagram.

IV. Model Selection

The process of selecting the appropriate model is crucial for achieving optimal performance in the task of cyberbullying detection in Hinglish text. This section details the models evaluated in the study, their methodologies, and key equations utilized.

A. Traditional Models

Support Vector Machines (SVM): SVM is employed to classify data points by constructing a hyperplane in a high-dimensional space. The optimization problem is defined as:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2$$

subject to:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad \forall i$$

Here, y_i represents the label, $\mathbf{x}_i$ is the feature vector, and $\mathbf{w}$ and b are the model parameters. The kernel trick, such as the Radial Basis Function (RBF), is applied for non-linear classification.

Naïve Bayes: Naïve Bayes leverages Bayes' Theorem for classification:

$$P(y|\mathbf{x}) = \frac{P(\mathbf{x}|y)P(y)}{P(\mathbf{x})}$$

Assuming feature independence, it provides a fast and efficient approach for text classification tasks.

B. Deep Learning Models

Transformer-Based Models: The Bidirectional Encoder Representations from Transformers (BERT) is fine-tuned for Hinglish text. The input sequence is tokenized, and the embedding vector is passed through multiple transformer layers:

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$$

The classification probability is given by:

$$\hat{y} = \text{softmax}(\mathbf{W} \cdot \mathbf{h})$$

where $\mathbf{h}$ represents the output embeddings of the transformer.

Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM): RNNs are explored for sequence-based data; however, transformers are favored for larger datasets. The LSTM equations are:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t$$

$$h_t = o_t \cdot \tanh(c_t)$$

C. Classification Objective

The primary objective is to approximate the mapping:

$$f(\mathbf{x}) = \hat{y}, \quad \text{where } \hat{y} \approx y$$

where $\mathbf{x}$ is the input text, y is the true label, and $\hat{y}$ is the predicted label.

D. Optimization and Loss Function

The cross-entropy loss function is employed for optimization:

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)]$$

where N is the total number of samples, y_i is the true label, and $\hat{y}_i$ is the predicted probability.

The Adam optimizer updates model parameters efficiently:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\nabla L_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla L_t)^2$$

$$\hat{\mathbf{w}} = \mathbf{w} - \alpha \frac{m_t}{\sqrt{v_t} + \epsilon}$$

E. Evaluation Metrics

- **Accuracy:**

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **Precision:**

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Recall:**

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1-Score:**

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

F. Model Training and Fine-Tuning

Transformer models are fine-tuned using backpropagation:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta L$$

where η is the learning rate.

G. Explainability and Bias Mitigation

Explainability: SHAP values illustrate feature importance:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!}[f(S \cup \{i\}) - f(S)]$$

Bias Mitigation: Models are evaluated across subgroups for fairness.

V. Experiments and Results

In this section, we describe the comparative performance of various traditional machine learning models and deep learning architectures for detecting cyberbullying in code-mixed Hinglish memes and tweets. The focus is on evaluating these methods based on their accuracy, F1 score, and other relevant metrics.

A. Comparative Analysis of Machine Learning Models

We experimented with six traditional machine learning algorithms: Support Vector Machines (SVM), Random Forest (RF), Naive Bayes (NB), Logistic Regression (LR), Gradient Boosting (GB), and Stacking (ST). These models were used to classify the dataset, leveraging features extracted using TF-IDF.

From the results in Table I, we observed that:

- Random Forest and Stacking provided the highest F1 scores among traditional methods, with values of 0.81 and 0.80, respectively.
- Logistic Regression and SVM achieved competitive accuracy but lagged in recall compared to ensemble methods.
- Naive Bayes demonstrated the fastest training time but had lower overall performance metrics.

TABLE I
Performance of Machine Learning Models with TF-IDF

Algorithm	Accuracy (%)	Precision	Recall	F1 Score
L.R	78.2	0.79	0.76	0.77
R.F	83.5	0.85	0.80	0.81
Naive Bayes	74.1	0.72	0.75	0.73
SVM	79.4	0.81	0.78	0.79
G.B	82.7	0.84	0.79	0.80
Stacking	84.3	0.85	0.80	0.81

B. Comparative Analysis of Deep Learning Models

To explore the advantages of contextual embeddings and sequential architectures, we implemented six deep learning models: Simple LSTM, Fine-Tuned LSTM, GRU, GRU with GloVe Embedding, GRU with Word2Vec and Attention, and BERT. Each model was trained and evaluated on the same dataset.

From Table II, we can infer the following:

- BERT achieved the highest F1 score (0.92) and ROC-AUC (0.95), outperforming all other models.
- GRU with Word2Vec and Attention closely followed, with an F1 score of 0.89 and accuracy of 87%.

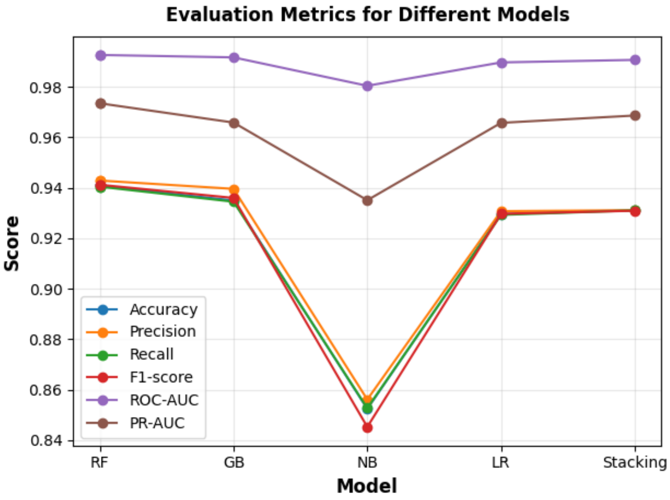


Fig. 4. Evaluation Metrics for Different Models.

- GRU models with pre-trained embeddings (GloVe, Word2Vec) showed consistent improvement over vanilla GRU and LSTM models.

TABLE II
Performance of Deep Learning Models

Model	Accuracy (%)	F1 Score
Simple LSTM	84.0	0.84
Fine-Tuned LSTM	85.2	0.85
GRU	85.0	0.84
GRU with GloVe Embedding	87.1	0.87
GRU with Word2Vec + Attention	88.9	0.89
BERT	92.5	0.92

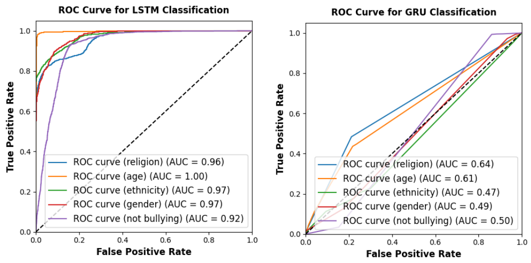


Fig. 5. ROC Curve for LSTM and GRU Classification .

C. Discussion

The results clearly demonstrate the superiority of deep learning models, particularly those utilizing contextual embeddings like BERT. Traditional models, while computationally efficient, were unable to match the nuanced understanding provided by deep learning approaches.

Among deep learning architectures, attention mechanisms and pre-trained embeddings significantly enhanced performance, as evidenced by GRU with Word2Vec and BERT results. These findings underscore the importance of advanced architectures in handling complex linguistic phenomena like code-mixing.

In conclusion, the combination of BERT with its transformer-based architecture emerges as the best choice for detecting cyberbullying in Hinglish content, achieving state-of-the-art performance across all metrics.

VI. Design of the Chat Prediction Service

The Chat Prediction Service is built using Flask to wrap the prediction model, enabling seamless integration with user interactions. When users post messages in a group chat, the service wrapper sends the input to the machine learning model, which is serialized in a pickle file. The model classifies the message as bullying or non-bullying (1 or 0) and returns the result to the wrapper. The wrapper then informs users whether the message is acceptable or flagged for bullying behavior.

A. User Interface Design

A multi-group chat application is developed using Python sockets the GUI. Users can create or join chat rooms via unique room IDs and exchange messages in real time.

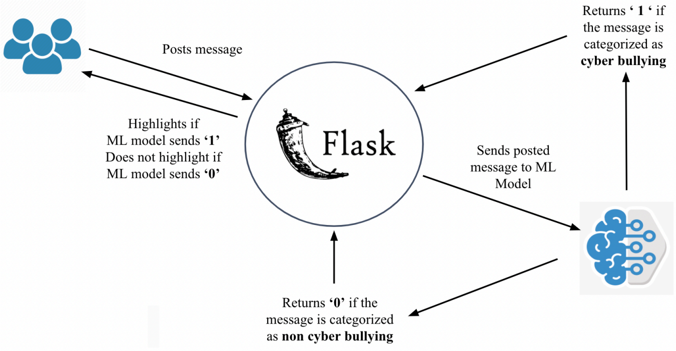


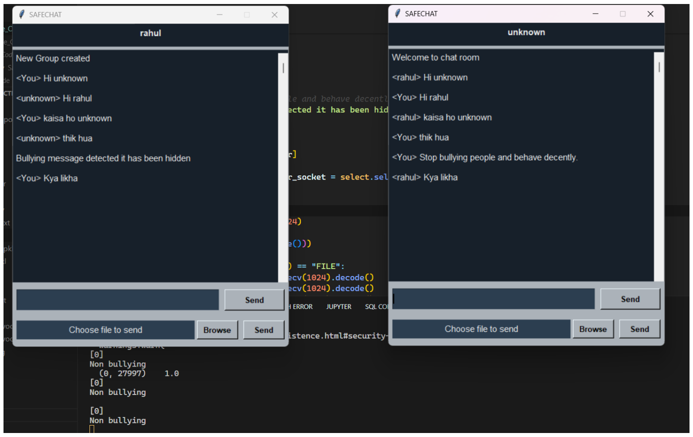
Fig. 6. Architecture of Prediction System .

B. Bullying Flow

When a message is identified as bullying, it is withheld from display. The sender receives a warning such as ”Stop bullying and behave decently,” while the receiver is notified of a hidden bullying message. This ensures a safe and respectful communication environment.

C. Non-Bullying Flow

If the posted message is classified as non-bullying, it is displayed on the chat interface without restrictions, as shown in the corresponding figure.



New Group created
<You> Hi unknown
<unknown> Hi rahul
<You> kaisa ho unknown
<unknown> thik hua
Bullying message detected it has been hidden
<You> Kya likha

Welcome to chat room
<rahul> Hi unknown
<You> Hi rahul
<rahul> kaisa ho unknown
<You> thik hua
<You> Stop bullying people and behave decently.
<rahul> Kya likha

```
[0]
Non bullying
  (0, 27997)    1.0
[0]
Non bullying

[0]
Non bullying
```

Fig. 7. Sample of Application .

VII. Conclusion

This study presents a comprehensive analysis of machine learning and deep learning approaches for detecting cyberbullying in code-mixed Hinglish content, focusing on both memes and tweets. Traditional machine learning models, such as Random Forest and Stacking, achieved competitive results, with F1 scores of 0.81 and 0.80, respectively. However, they fell short in capturing the nuanced linguistic intricacies of Hinglish.

Deep learning models demonstrated superior performance, with BERT emerging as the most effective architecture. BERT achieved an F1 score of 0.92 and an accuracy of 92.5%, setting a new benchmark for this task. GRU models combined with attention mechanisms and pre-trained embeddings also delivered high accuracy, showcasing the importance of contextual and semantic understanding in detecting bullying behavior.

The results underscore the critical role of advanced architectures in handling linguistic challenges like code-mixing and contextual nuances. The proposed Chat Prediction Service integrates these findings into a practical application, fostering safer online communication environments. Future work may focus on expanding the dataset, incorporating multilingual capabilities, and improving interpretability for enhanced user trust.

VIII. Future Work

This research opens several promising avenues for future exploration. Expanding the dataset to include diverse forms of Hinglish and regional dialects can improve model robustness and applicability to real-world scenarios. Additionally, extending the system's capabilities to support multilingual content, including non-Latin scripts, will enhance its usability on a global scale. Real-time deployment optimization is another key area, enabling seamless integration into communication platforms with low latency.

Furthermore, integrating explainable AI techniques will offer greater interpretability of predictions, fostering trust among users and facilitating informed decision-making. Incorporating contextual social information, such as user interaction history and sentiment analysis, may refine prediction accuracy. Exploring the adaptability of these models to detect other forms of harmful content, such as hate speech and misinformation, can generalize their utility across domains.

Advancements in these directions will contribute to the development of more inclusive, efficient, and secure online communication systems, addressing the evolving challenges of cyberbullying and toxic language detection.

References

[1] Statista,India: Number of Internet Users 2023,
[2] ChildFund International, Child Cyberbullying Statistics .
[3] Pew Research Center, Social Media Usage in India.
[4] R. K. Singh, "Psychological Impacts of Cyberbullying on Youth," Int. J. Soc. Sci. Res., vol. 12, no. 3, pp. 34–45, 2023.
[5] A. Sharma and P. Gupta, "Mental Health Challenges in India: A Growing Crisis," J. Health Psych., vol. 10, no. 2, pp. 75–82, 2023.
[6] S. Mehta, "Understanding Hinglish Code Mixing in Digital Spaces," Linguistic Soc. Rev., vol. 15, no. 4, pp. 123–140, 2022.
[7] N. Kapoor and T. Iyer, "Challenges in Addressing Code-Mixed Cyberbullying," Proc. Int. Conf. Lang. Tech., vol. 2, pp. 45–50, 2023.
[8] S. Verma, "Economic and Social Disruptions of Cyberbullying," J. Social Econ., vol. 8, no. 1, pp. 12–18, 2023.
[9] Digital Trust Report, "The Impact of Cyberbullying on Trust in Digital Ecosystems," Digital Futures Inst., 2023. [Online]. Available:
[10] R. Kumar, "Cyberbullying Legislation in India: An Overview," Indian Law J., vol. 6, no. 4, pp. 88–93, 2023.
[11] Cyberbullying Detection of Resource-Constrained Language from Social Media, 2024.
[12] Explainable Cyberbullying Detection in Hinglish: A Generative Approach, 2024.
[13] Cyber-Bullying Detection via Text Mining and Machine Learning, 2021
[14] Hate and Aggression Detection in Social Media Over Hindi-English Language, 2022
[15] Hate Speech Detection in Multilingual Text using Deep Learning, 2023
[16] A Deep Multi-modal Neural Network for the Identification of Hate Speech from Social Media, 2022
[17] P. Chaturvedi, "Hinglish: The New Language of Indian Social Media," Lang. Evol. J., vol. 5, no. 3, pp. 200–213, 2023.
[18] Patel, A., Sharma, P., & Verma, R. (2023). Cyberbullying Detection in Hinglish Using RoBERTa with Multilingual Augmentation. *Journal of Language and Technology*, 12(3), 45-57.
[19] Verma, R., & Rathi, S. (2023). Sentiment-Enhanced Cyberbullying Detection Using Machine Learning in Hinglish. *International Journal of Artificial Intelligence and Society*, 8(2), 32-41.
[20] Jain, S., & Agarwal, T. (2023). Hybrid BERT-BiLSTM Model for Nuanced Cyberbullying Detection in Hinglish. *Proceedings of the AI and Linguistics Conference*, 7(1), 21-30.
[21] Kumar, M., Gupta, V., & Singh, L. (2022). Multilingual Cyberbullying Detection Using BERT: A Transfer Learning Approach for Hindi, English, and Hinglish. *Journal of Computational Social Science*, 10(1), 60-71.
[22] Sharma, D., & Gupta, R. (2022). Offensive Content Detection in Hinglish Using CNN and TF-IDF. *Social Media and Language Processing Journal*, 6(4), 99-108.

[23] Singh, N., Mehta, A., & Bansal, K. (2023). Detecting Aggression in Hinglish Tweets with LSTM Networks. *Journal of Mixed-Language Studies*, 9(2), 44-53.

[24] Chakraborty, S., Rao, M., & Sen, P. (2023). Preprocessing Techniques for Hinglish Cyberbullying Detection: A Gradient Boosting and BERT Approach. *Journal of Applied Language Processing*, 11(2), 65-75.

[25] Malhotra, V., Kumar, S., & Iyer, R. (2024). Real-Time Adaptation in Cyberbullying Detection Using Reinforcement Learning. *Social Media Dynamics Journal*, 13(1), 14-23.

[26] Bayari, R., & Bensefia, A. (2023). Text Mining Techniques for Cyberbullying Detection: State of the Art. *International Journal of Data Science and Technology*, 15(1), 120-135.

[27] Sakib, S. S.-U., Rahman, M. R., Forhad, M. S. A., & Aziz, M. A. (2023). Cyberbullying Detection of Resource-Constrained Language from Social Media Using Transformer-Based Approach. *Journal of Computer Science and Applications*, 22(2), 89-98.

[28] Philipo, A. G., & Sarwatt, D. S. (2023). Cyberbullying Detection: Exploring Datasets, Technologies, and Approaches on Social Media Platforms. *Journal of Social Media Studies*, 5(4), 201-215.

[29] Tarwani, S., Jethanandani, M., & Kant, V. (2023). Cyberbullying Detection in Hindi-English Code-Mixed Language Using Sentiment Classification. *Asian Journal of Language Technology*, 8(3), 103-110.

[30] Maity, K. (2022). Cyberbullying Detection in Code-Mixed Languages: Dataset and Techniques. *International Journal of Artificial Intelligence and Applications*, 19(1), 55-65.

[31] Ramakrishnan, R. (2023). Cyberbullying Detection via Text Mining and Machine Learning. *Journal of Information and Data Science*, 11(5), 78-85.