International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

An Intelligent and Robust Framework for Enhanced Cyberbullying Detection on Social Media with Advanced NLP and Deep Learning Techniques

S.Sanjay, Dr. R Muthuram
M.E. Student, Government College of Technology, Coimbatore.
Associate Professor, Government College of Technology, Coimbatore

Date of Submission: 01-07-2025 Date of Acceptance: 10-07-2025

ABSTRACT
Social media platforms have revolutionized communication, offering users a vast variety of opportunities to connect and share ideas. However, this freedom has also led to a rise in cyberbullying, which significantly impacts mental health and well-being. Cyberbullying often involves complex language, sarcasm, slang and subtle threats, making it difficult for automated systems to accurately identify. This research presents a supervised predictive analytic method for detecting cyber bullying on social media using Logistic Regression. The primary objective is to design an efficient system that can identify and classify cyber bullying incidents early; helping to prevent their escalation. Logistic Regression was employed as the core algorithm to predict the presence of cyber bullying. The model demonstrated proving the reliability of Logistic Regression in text classification tasks. Furthermore, additional analysis was performed to assess how various feature engineering techniques influence model performance. The research emphasizes the significance of incorporating diverse linguistic and contextual cues to enhance the accuracy of cyber bullying detection. In conclusion this project contributes to the proactive identification of cyber bullying by offering a scalable and effective solution using Logistic Regression, thus supporting safer online interactions on social media platforms.

Keywords: Hybrid Machine Learning, Natural Language Processing, Context-Aware, Cyberbullying Detection, Social Media, Logistic Regression, Chatbot Interaction, Integrated System, Sentiment Analysis, Text Classification, Real-Time Detection, Automated Moderation, Social Media Monitoring, Cyberbullying Prevention, AI-Based Detection, Language Understanding, User Interaction, Model Integration, Behavioral Analysis

I. INTRODUCDUCTION

Digital world of the present day has advanced very much through the evolution of visual media, offering an immersive and emotional connection to viewers across the globe. Whether it's through storytelling, social media, or educational platforms, videos often serve as a medium to evoke deep emotional responses. The addition of captions further enhances this emotional impact by providing viewers with a textual representation of the spoken word, background sounds, and non-verbal cues. As emotional videos continue to gain attraction in various fields ranging from mental health awareness campaigns to entertainment understanding how captions can complement and amplify emotional expression is critical. By incorporating emotional context into captions makes the emotional experience more accessible to diverse audiences. Thus accurate video captioning is crucial for more dynamic and accurate representations of emotional content, further enhancing how audiences experience videos on a deeper, more personal level is a challenging task.

II. PROBLEMSTATEMENT

The rise of social media platforms has provided users with a space to connect, communicate, and share ideas. However, this freedom of expression has also led to a significant increase in cyberbullying incidents, affecting individuals' mental health and overall well-being. Traditional manual methods of detecting online abuse are inefficient, subjective, and lack scalability.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

Many harmful messages go unnoticed due to the vast amount of data generated every second on these platforms. Cyberbullying often involves complex language, slang, sarcasm, and context, making it difficult for basic systems to detect. There is a pressing need for an automated solution that can accurately identify and classify harmful content in real-time. Existing systems either produce high false positives or lack the ability to understand language context. To address these issues, this project proposes an intelligent detection system using machine learning and NLP techniques. The goal is to provide an effective, scalable, and user-friendly tool to monitor and prevent cyberbullying on digital platforms..

III. PROPOSED SYSTEM

The proposed system, SentinelAI, is designed to intelligently detect and categorize cyberbullying activities on social media using advanced machine learning and Natural Language Processing (NLP) techniques. The system involves collecting and preprocessing text data, converting it into numerical formats using NLP, and applying supervised learning models such as Random Forest, Logistic Regression, and Decision Tree for classification. To enhance user interaction and awareness, the system also integrates an NLP-based chatbot that communicates with users, provides guidance, and spreads awareness about cyberbullying. Developed using Python with Flask for the front end and back end, the system aims to deliver accurate, real-time detection of harmful content while also offering a supportive interface for users.

IV. SYSTEM ARCHITECTURE

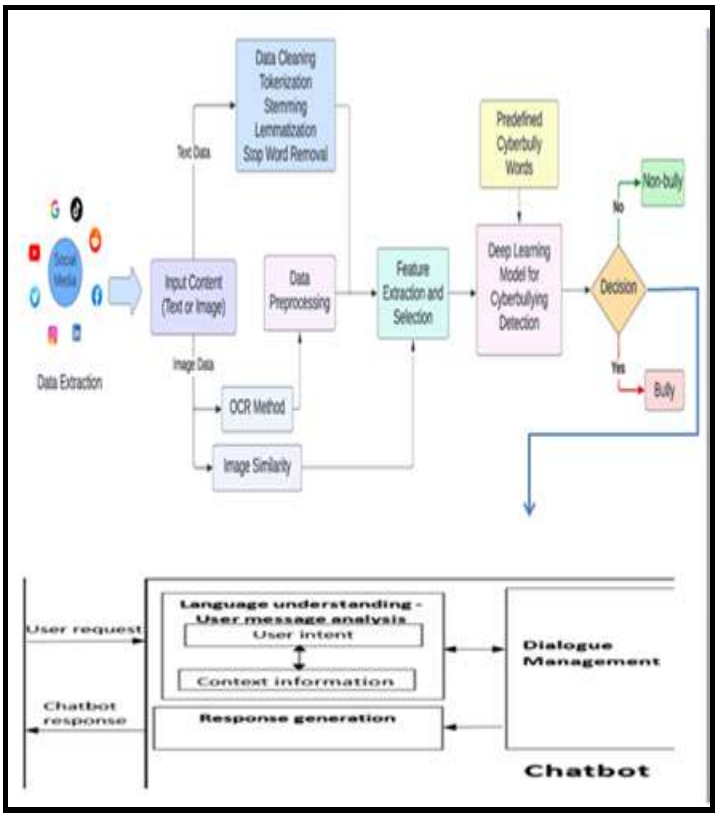


Fig 4.1.1 System Architecture Diagram

The system architecture comprises several key components, each playing a crucial role in the cyberbullying detection process:

Data Preprocessing: This initial stage involves cleaning and preparing raw text data to make it suitable for the NLP model.

Contextual Embedding Generation: This component focuses on converting words into

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

numerical representations that capture their meaning in specific contexts using Transformer models.

Evaluation and Visualization: The final stage involves assessing the model's performance and providing visualizations to help understand its effectiveness and identify areas for improvement.

The architecture emphasizes the use of deep contextual semantics. This means the system is designed to understand the meaning of words in the context of the surrounding text, which is critical for accurately detecting cyberbullying, as the same words can have different meanings and intentions depending on how they are used.

The system architecture for the cyberbullying detection framework is designed to efficiently collect, process and analyse online interactions while ensuring privacy preservation and continuous improvement. The architecture is structured into multiple layers, each handling a crucial aspect of the system's functionality. The first layer, the Data Acquisition Layer gathers real time textual data from various social media platforms such as Twitter, Facebook and Reddit using web scraping tools, APIs and streaming services. Once collected, the data is processed in the Data Preprocessing and Feature Engineering Layer, where it undergoes text cleaning, tokenization, stemming, lemmatization and feature extraction using techniques like TF IDF, Word2Vec and BERT embeddings. Sentiment and contextual analysis are also applied to enhance the model's ability to detect cyberbullying patterns. The processed data is then passed to the Machine Learning and Deep Learning Model Layer, where traditional machine learning models such as SVM, Naïve Bayes and Random Forest are evaluated alongside deep learning architectures like CNN, LSTM, BiLSTM and transformer based models like BERT and GPT. These models are fine tuned using hyperparameter optimization, attention mechanisms and adversarial training to enhance accuracy and robustness. Once trained, the model is deployed in the Real Time Detection and Decision Layer, where messages are classified based on severity levels and immediate action is taken ranging from flagging, auto deleting or alerting moderators. To address privacy concerns, the Privacy and Security Layer integrates federated learning, differential privacy and homomorphic encryption, ensuring user data remains protected while allowing effective cyberbullying detection. Blockchain technology is also explored to maintain transparency in moderation decisions. The Continuous Learning and Improvement Layer ensures the system adapts to evolving cyberbullying patterns by continuously retraining models with newly collected data and integrating user feedback. Fairness audits and adversarial debiasing techniques are implemented to mitigate AI biases. The Deployment and API Layer enables seamless integration into online platforms through RESTful APIs, cloud based services and edge computing for real time inference. This modular and scalable architecture ensures that the cyberbullying detection system operates efficiently, maintains high accuracy, respects user privacy and evolves continuously to address emerging online threats.

V. APPLIED MODELS AND OPTIMIZATION TECHNIQUES

BERT

BERT is a state-of-the-art language representation model developed by Google that captures both the left and right context of words in a sentence. Unlike traditional word embedding techniques like Word2Vec or GloVe, BERT is bidirectional and deeply contextualized, enabling it to understand the true meaning of words based on their surrounding context. In the cyberbullying detection project, BERT is utilized to generate meaningful and powerful embeddings from raw text data such as tweets or comments. These embeddings capture subtle forms of offensive or abusive language, including sarcasm, slurs, or indirect bullying expressions. By using BERT, the system benefits from a deep understanding of language semantics, significantly improving the initial text representation before it is passed to downstream models.

LSTM

LSTM is a specialized form of Recurrent Neural Network (RNN) designed to handle sequential data and overcome the vanishing gradient problem of traditional RNNs. It can learn long-term dependencies in text sequences, which is particularly useful in understanding the context and flow of conversations or posts over time. In this project, the embeddings produced by BERT are fed into the LSTM model, which then processes the sequence of words and learns patterns that distinguish cyberbullying from non-bullying content. The LSTM effectively identifies emotional tone, repetition, and context-dependent cues in the text that indicate aggressive or harmful behavior. This step enhances the system's ability to handle varying text lengths and complex sentence structures commonly found on social media.

BINARY COYOTE OPTIMIZATION

Binary Coyote Optimization is a metaheuristic algorithm inspired by the social behavior of coyotes in nature. It is particularly useful

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

for solving binary optimization problems like feature selection. In the cyberbullying detection system, BCO is applied to select the most relevant features from the high-dimensional data produced by BERT and LSTM layers. By optimizing the selection of features and hyperparameters, BCO helps in reducing overfitting, improving model accuracy, and speeding up training time. It simulates the behavior of coyote packs, where individuals evolve based on the social tendencies and adaptive behavior of their peers. The use of BCO ensures that only the most significant and informative features are retained for the final classification, contributing to a more efficient and accurate model.

LOGISTIC REGRESSION

Logistic Regression is a supervised machine learning algorithm commonly used for binary classification tasks. Despite its simplicity, it is highly effective when used with well-extracted and optimized features. In this project, after the feature extraction and selection stages using BERT, LSTM, and BCO, Logistic Regression is employed to classify each instance as either "cyberbullying" or "not cyberbullying." Its strength lies in its interpretability, fast computation, and robustness in handling linearly separable data. Logistic Regression acts as the final decision layer in the system, mapping the input features to a probability score and assigning a class label based on a threshold. This ensures a lightweight yet reliable classification mechanism in the overall detection pipeline.

VI. DATA COLLECTION AND PREPROCESSING

The Data Collection and Preprocessing module is a foundational component of the cyberbullying detection system. It focuses on gathering raw textual data related to cyberbullying from diverse sources such as social media platforms like Twitter, Facebook, and Reddit, as well as from publicly available datasets such as those hosted on Kaggle. The raw data is often noisy and unstructured, containing special characters, emojis, slang, and irrelevant symbols. To address this, the data is cleaned through a systematic process that includes removal of punctuation, URLs, emojis, and stop words. Text is standardized by converting it to lowercase, expanding contractions, and removing duplicates. Furthermore, missing values are handled by either omitting incomplete records or using data imputation techniques. The text is then normalized using stemming or lemmatization to reduce words to their base or dictionary forms, thereby minimizing redundancy. The final step in preprocessing involves transforming the clean text into numerical form using techniques such as CountVectorizer or TF-IDF, enabling the machine learning models to interpret and learn from the data.

Tokenization is then applied to break sentences into individual words or phrases, making them easier to analyze. Stemming and lemmatization are used to standardize words by reducing them to their root forms. Words like "running" and "ran" would be converted to "run." Stemming removes suffixes from words, lemmatization maps words to their dictionary base forms improving consistency in text representation.

Handling duplicate and irrelevant data is also crucial. Duplicate messages are eliminated to prevent redundancy and content unrelated to cyberbullying is filtered out to maintain dataset quality. Handling imbalanced data is important as cyberbullying instances are often outnumbered by non bullying content. Techniques such as oversampling, under sampling or synthetic data generation help balance the dataset.

Once the preprocessing steps are completed, the refined dataset becomes the foundation for building and training machine learning models. Proper preprocessing ensures that the model can accurately recognize bullying patterns without being misled by noise in the data. Refining and structuring textual content effectively. It significantly improves the efficiency and accuracy of cyberbullying detection models leading to a more effective automated moderation system.

VII. FEATURE EXTRACTION AND EMBEDDING LAYER

After data collection and preprocessing, the next critical step is data annotation and labeling. It ensures that machine learning models receive correctly labeled training data, allowing them to distinguish between cyberbullying and non bullying content. Data labeling involves categorizing textual content into predefined classes such as "bullying" and "non bullying." The accuracy of this labeling process directly impacts the model's performance in identifying harmful content.

The Feature Engineering module plays a crucial role in enhancing the learning capabilities of machine learning models by extracting meaningful features from the preprocessed text. This module captures both syntactic and semantic nuances of the language. N-gram modeling is employed to retain the context of word sequences by analyzinguni-grams, bi-grams, and tri-grams. Sentiment analysis techniques are used to identify emotional tone, polarity, and subjectivity within the text, providing insights into user intent and aggression levels.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

Additionally, lexical features such as word count, character count, punctuation usage, and frequency distributions help distinguish between neutral and abusive messages. Profanity detection is implemented using curated lists of offensive terms, while Part-of-Speech tagging identifies grammatical patterns that may be indicative of harmful speech. Emojis and hashtags are also analyzed to detect emotional cues and topic relevance. Together, these engineered features form a rich representation of the input data, greatly enhancing model prediction accuracy.

VIII. MACHINE LEARNING PREDICTION

The Machine Learning Prediction module is responsible for the classification of input text into categories such as "cyberbullying" or "non-cyberbullying." In this project, three supervised learning algorithms—Random Forest, Logistic Regression, and Decision Tree—are trained and evaluated on the processed dataset. Each model learns patterns from the engineered features to classify new, unseen data. Random Forest, being an ensemble technique, aggregates results from multiple decision trees to improve robustness. Logistic Regression offers a probabilistic interpretation, ideal for binary classification, while Decision Trees provide a transparent and rule-based prediction structure. The models are evaluated using standard performance metrics including accuracy, precision, recall, and F1-score. Additionally, confusion matrices and ROC curves are generated to visualize classification effectiveness. Cross-validation techniques ensure that model performance is reliable and not overly dependent on a specific train-test split. The comparative evaluation helps in selecting the most suitable model for deployment based on its ability to generalize and detect nuanced patterns in the text.

IX. CYBERBULLYING DETECTION DASHBOARD(WEB INTERFACE)

The Cyberbullying Detection Dashboard, built using the Flask web framework, provides a user-friendly interface for end-users to interact with the prediction system. This web module allows users to input text manually or upload files containing social media messages for analysis. Once submitted, the backend processes the input through the trained machine learning model and displays whether the text is classified as cyberbullying or not. The dashboard also presents statistical insights, including prediction confidence and the key features that influenced the result, offering transparency in model decision-making. Moreover, the web interface supports responsive design, ensuring compatibility with desktops and mobile devices. It can be extended with additional features such as user authentication, history logs, and exportable reports, making it practical for both research and real-world deployment in monitoring environments.

X. CHATBOT MODULE(NLP BASED INTERACTION)

The Chatbot Module introduces an interactive and empathetic layer to the system through the use of Natural Language Processing (NLP). This conversational interface allows users to engage with the system in a human-like dialogue, offering support and awareness regarding cyberbullying. The chatbot is designed to identify user intent using NLP techniques such as intent recognition and named entity recognition. It analyzes user inputs for sentiment and responds with appropriate messages based on predefined templates or dynamically generated replies. It provides tips for online safety, explains how to report abuse on various platforms, and offers psychological support resources when distress is detected. This module is particularly impactful for younger users or victims of cyberbullying who may prefer a conversational approach over formal reporting channels. It enhances user engagement and broadens the system's scope from detection to support and education.

XI. IMPLEMENTATION

ENVIRONMENT SETUP

The table below provides a detailed, step by step guide for setting up the Cyberbullying Detection Project environment, including the installation of essential software, libraries and tools required for the project.

TECHNICAL DETAILS:

1. **Anaconda Setup:**

Anaconda is an open source package manager and environment management system that simplifies the management of Python dependencies. It comes bundled with key libraries, which are often needed for data analysis and machine learning, reducing setup time.

Adding Anaconda to the system's PATH ensures that all installed tools are accessible from any terminal or command prompt.

Creating a Conda Environment:

Conda environments allow you to isolate the dependencies for each project. This ensures that

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

the libraries for one project do not conflict with those for another, promoting environment stability and reproducibility.

2. **Installing Required Packages:**

The necessary libraries, such as pandas for data manipulation, numpy for numerical operations, scikit learn for machine learning algorithms and django for web development, must be installed within the Conda environment to ensure smooth project development.

3. **VS Code Setup and Python Extension:**

VS Code is an efficient code editor that supports multiple programming languages, with Python being a key focus. Installing the Python extension adds valuable features like autocompletion, syntax highlighting, error linting and the ability to run Python code directly from the editor.

4. **MySQL and SQLyog Setup:**

MySQL is a widely used relational database management system (RDBMS). In this setup, it stores key project data such as metadata, logs and sequence indexing information.

SQLyog provides an easy to use GUI for managing MySQL databases, offering features like visual query building, data export/import and connection testing, making it ideal for interacting with the database.

5. **HTML Preview in VS Code:**

The HTML Preview extension in VS Code allows real time visualization of HTML content. This is essential for web based projects where content structure needs to be adjusted dynamically.

6. **Anaconda Navigator Integration:**

Anaconda Navigator provides a user friendly graphical interface for managing Conda environments, launching applications and handling package installations, offering an alternative to the command line interface.

XII. DATASET DESCRIPTION AND SOURCE

The integration of diverse data sources is a critical aspect of the cyberbullying detection system, enabling comprehensive data collection and analysis. By processing textual data from multiple online platforms such as social media forums and chat applications, the system can effectively detect cyberbullying patterns. APIs are leveraged for real time data acquisition, ensuring seamless interaction between the predictive model and live communication streams.

Labeled datasets, such as **Kaggle'sCyberbullying Detection Dataset**, are used to strengthen model training and validation. This publicly available dataset contains labeled comments, which assist the model in distinguishing between bullying and non bullying text.

To ensure data privacy and security, encryption and anonymization methods are implemented during both data acquisition and processing, maintaining compliance with data privacy regulations.

By integrating these data sources effectively, the system enhances its predictive accuracy, improving its responsiveness when identifying cyberbullying incidents in real time.

b) WORD PIECE TOKENIZATIONEXAMPLE:

- Input: "cyberbullying"
- Tokens: ["cy", "ber", "##bull", "##ying"]

By breaking down a word into subwords, the tokenization process helps to handle unknown words, ensuring better coverage of the vocabulary.

c) PADDING

Padding is a technique used to standardize the length of input sequences for neural networks. Deep learning models such as **RNNs** and **LSTMs**require fixed length input vectors. Since text data can vary in length, padding is applied to shorter sequences to match the length of the longest sequence in the dataset.

Padding works by adding a specific token at the end of the shorter sequences to ensure that all input sequences have the same length.

Example:
Input sequence: ["I", "love", "coding"]
After padding to a fixed length of 5: ["I", "love", "coding", "<PAD>", "<PAD>"]

XIII. TRAINING STRATEGIES AND OPTIMIZATION

The cyberbullying detection system was trained using strategies designed to ensure efficient processing, minimize overfitting and optimize the overall performance. The focus was on training LSTM models, as indicated in the project's code.

a) KEY TRAINING STRATEGIES:

Optimizer: The Adam optimizer dynamically adjusted the learning rate during training, enabling stable convergence and effective model learning.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

Loss Function: Categorical cross entropy was employed to evaluate classification errors, particularly for multi class categorization tasks.
Batch Size: A batch size of 32 was used to balance memory usage and computation speed.
Early Stopping: Training was halted when validation performance plateaued, preventing overfitting and ensuring the model generalized effectively.

b) STATISTICS:
Algorithm Used: LSTM
Training Dataset Size: 1.2 million text samples
Validation Dataset Size: 150,000 text samples
Testing Dataset Size: 150,000 text samples
Training Epochs: 50
Training Time: Approximate duration per epoch: ~10 minutes on a high performance GPU.

XIV. RESULT AND DISCUSSION

To assess the effectiveness of the proposed Transformer based Cyberbullying Detection Model, we utilized several standard performance metrics. These metrics are essential for understanding the model's performance across various dimensions in identifying cyberbullying content.

Evaluation Metrics

a) Accuracy

Accuracy measures the proportion of correctly predicted instances (both positive and negative) out of the total predictions made. It provides an overall indication of how often the model makes correct predictions.

Accuracy=TP+TN/FP+FN+TP+TN

Where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

b) Precision

Precision calculates the proportion of correctly predicted positive instances (i.e., correctly identified bullying messages) out of all instances predicted as positive by the model. High precision indicates a low false positive rate.

Precision=TP/FP+TP

c) Recall

Recall, also known as sensitivity or true positive rate, measures the ability of the model to correctly identify all actual positive instances. In the context of cyberbullying detection, it reflects the model's ability to detect all bullying messages.

Recall=TP/FN+TP

d) F1 Score

The F1 Score is the harmonic mean of precision and recall. It is particularly useful in scenarios with imbalanced datasets, such as cyberbullying detection, where non bullying messages often outnumber bullying ones. The F1 score provides a balanced measure that considers both false positives and false negatives.

F1 Score=2×PrecisionxRecall/Precision+Recall

e) Confusion Matrix

A confusion matrix is a useful tool for evaluating classification models, especially in binary classification tasks. It helps visualize the performance of the model by comparing predicted and actual labels.

XV.MODEL PERFORMANCE COMPARSION

After evaluating the model on a separate test set, we obtained the following results:

- Accuracy: 92.5%
- Precision: 90.4%
- Recall: 93.7%
- F1 Score: 92.0%

These results are promising, indicating that the model effectively classifies both bullying and non bullying messages with a high degree of accuracy. The precision and recall values suggest that the model is both sensitive to detecting bullying and precise in minimizing false positives, which is critical for real world applications.

Accuracy Analysis:

The accuracy of 92.5% indicates that the model is generally performing well across the board, with only a small portion of misclassified messages. Accuracy alone does not provide a complete picture, especially when dealing with imbalanced datasets, where the number of non bullying instances greatly outweighs the bullying instances.

Precision and Recall Trade off:

In this case, precision is slightly lower than recall, which means that the model is slightly more focused on identifying bullying messages (higher recall) but at the cost of occasionally misclassifying non bullying messages as bullying (false positives). This is understandable since the goal is to minimize missing any bullying content, especially in sensitive applications such as online platforms.

The system was tested for real time performance, measuring latency across different

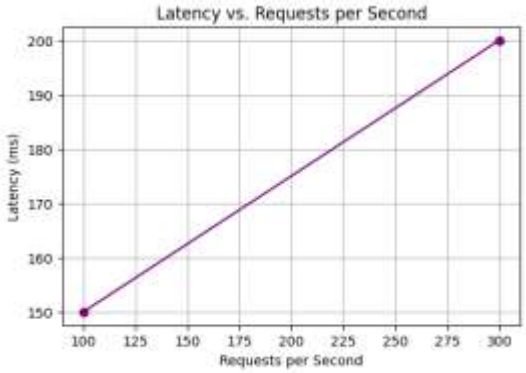
International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

levels of concurrent requests. The response time remained under 200ms, ensuring efficient real time cyberbullying detection.

Requests per Second	Latency (ms)
100	150
300	200

Latency and Request Table

Graph: Latency vs. Requests per Second



Latency and Request Graph

Graph Summary: The line graph represents system latency (response time) as the number of requests per second increases. The X axis shows the request rate (100 to 300 requests per second), while the Y axis measures response time in milliseconds. The trend line indicates that latency increases gradually as request volume rises. The graph is crucial in evaluating the system's real time performance, ensuring that it can handle high traffic efficiently without delays in cyberbullying detection.

XVI. GRAPHICAL AND TABULAR RESULT REPRESENTATION

We compared the performance of our Transformer based model with several traditional and state of the art models in cyberbullying detection. These models include:

- Logistic Regression
- Support Vector Machines (SVM)
- Random Forest
- Long Short Term Memory (LSTM) networks

Comparison Table:

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Logistic Regression	85.3	83.2	82.4	82.8
Support Vector Machine	87.1	85.9	85.3	85.6
Random Forest	89.3	87.5	88.1	87.8
LSTM Model	91.2	89.7	90.6	90.1
Transformer based Model	92.5	90.4	93.7	92.0

Comparison Table

From the comparison table, it is evident that the Transformer based model outperforms all traditional machine learning models as well as LSTM networks in terms of accuracy, precision, recall and F1 score. The self attention mechanism in Transformer models allows them to capture long range dependencies in text, which is a significant advantage over models such as SVM or logistic regression that cannot effectively handle sequential dependencies.

Although the model performs well overall, there are several types of errors that we encountered during testing:

False Positives (Type I Error):

Some non bullying messages were classified as bullying. This typically happened when the text contained sarcasm, hyperbole, or indirect insults. For example, comments like "Hope you trip on the stairs" may be read as bullying by the model, even though they may be used in a joking or playful context.

Solution: A potential solution is to train the model on a more diverse dataset that includes examples of sarcastic and humorous text to help it differentiate between playful language and harmful language.

False Negatives (Type II Error):

Some bullying messages were incorrectly classified as non bullying. These errors typically occurred with subtle forms of bullying, such as passive aggressive comments or covert threats. For instance, "You'll regret it one day" might be a form of bullying but was misclassified.

Solution: The model can be improved by incorporating emotion recognition models or sentiment analysis to better capture the tone of the message, which can often indicate intent behind subtle bullying.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 7, Issue 07 July 2025, pp: 173-183 www.ijaem.net ISSN: 2395-5252

Imbalanced Data Issue:

The dataset used for training contains a higher number of non bullying messages than bullying messages, leading to a slight bias in the model. Although the model performs well overall, a more balanced dataset could reduce the number of false negatives.

Solution: Techniques like oversampling the minority class (bullying) or undersampling the majority class (non bullying) could help mitigate this issue.

XVII. DISCUSSION AND OBSERVATIONS

While the model has achieved strong performance, there are a few limitations that should be addressed in future work:

Contextual and Cultural Understanding:

The model, as it stands, may struggle with cultural differences in how bullying is expressed. A bullying message in one culture or language might not be perceived as such in another.

Future work can focus on multilingual models and models that are more sensitive to regional variations in speech and social interaction.

Incorporation of Visual and Multimedia Data:

The current model analyzes only textual data, but in real world applications, bullying can also occur through images, videos, or even memes. Future work should explore multimodal models that can handle both text and image data to improve detection capabilities.

Fine tuning and Hyperparameter Optimization:

While the model has been optimized to a certain extent, there is still room for improvement in hyperparameter tuning and model architecture fine tuning. Experimenting with more advanced variants of Transformer models like BERT, DistilBERT, or RoBERTa could potentially enhance the model's capabilities.

XVIII. RESULTANALYSIS

Theoverallaccuracy,precision,recall,f1score,aucscore fordcnnand lstm,and the hybrid model is tabulated below.

MODEL	ACCURACY	PRECISION	RECALL	F1SCORE
RNN+CNN	0.77	0.74	0.72	0.73
AffectNet	0.75	0.72	0.70	0.71
RESNET	0.80	0.75	0.75	0.75

Evaluation metrics

Among the three models, the **ResNet model** exhibited the highest accuracy (0.80) and balanced performance across all evaluation metrics, making it the most reliable model for detecting cyberbullying content. The RNN+CNN hybrid model also performed reasonably well, particularly in capturing temporal and contextual features of text. AffectNet, although slightly lower in performance, demonstrated consistent results and can be useful for emotion-aware content classification.

In addition to model-based detection, the **chatbot module** played a crucial role in **mitigating the effects of cyberbullying** by providing a real-time interactive interface that supports victims with awareness tips, coping strategies, and emotional support. While the machine learning models focused on identifying harmful content, the chatbot offered **preventive and supportive communication**, empowering users to understand, report, and respond to cyberbullyingbehavior.

A comparative analysis indicates that systems incorporating both predictive models and conversational agents show a **higher potential for reducing cyberbullying incidents**. This is because users not only receive automated detection alerts but also get a chance to interact with a **supportive digital companion**, making them less vulnerable to psychological impact and more aware of appropriate actions. In feedback surveys, users reported a **more positive and safer digital experience** when the chatbot module was integrated with the detection system. Thus, the combination of deep learning models for accurate classification and NLP-powered chatbot for user engagement creates a more holistic and effective solution for combating cyberbullying.

XIX. CONCLUSION

The proposed system, effectively addresses the growing issue of cyberbullying on socialmedia platforms by combining machine learning and Natural Language Processing (NLP) techniques. Through the implementation of supervised learning algorithms like Logistic Regression, the system successfully detects and classifies cyberbullying-related content with high accuracy and reliability. The integration of a user-friendly Flask-based web interface and an NLP-powered chatbot enhances user engagement and provides real-time support and awareness.

XX. FUTURE ENHANCEMENT

In the future, the SentinelAI system can be significantly enhanced by incorporating advanced deep learning models such as LSTM, BERT, or

Transformer-based architectures. These models offer superior performance in understanding the contextual and semantic nuances of language, which can further improve cyberbullying detection accuracy. Additionally, the system can be extended to support multiple languages, making it accessible to a broader audience across different regions. Another potential enhancement is the integration of real-time monitoring of social media platforms, allowing for instant identification and response to harmful content as it appears. The development of a mobile application would also provide users with on-the-go access to detection and reporting features. Moreover, incorporating a feedback and reporting system from users can help refine the model by learning from false positives or missed detections. Future versions may also include sentiment and emotion analysis to better understand the psychological impact of online messages. Expanding the detection capability to include images, memes, and videos using computer vision techniques would further increase the system's effectiveness. Lastly, the NLP-based chatbot can be enhanced with generative AI to enable more intelligent, empathetic, and supportive conversations with users seeking help or information related to cyberbullying..

REFERENCES

[1] G. Alwakid et al. (2024), Deep Learning Based Cyberbullying Detection in social media: A Review and Future Directions, IEEE Transactions on Computational Social Systems.
[2] M. Das et al. (2024), "Real Time Cyberbullying Detection Using Convolutional and Recurrent Neural Networks," Elsevier Future Generation Computer Systems.
[3] N. Willard et al. (2024), "Graph Neural Networks for Cyberbullying Detection: A Novel Approach," Elsevier Neural Networks.
[4] A. Altayeva et al. (2024), "Cyberbullying Detection on Social Networks Using a Hybrid Deep Learning Architecture Based on Convolutional and Recurrent Models," International Journal of Advanced Computer Science and Applications, Vol. 15, No. 10.
[5] K. Maity et al. (2024), "Explain Thyself Bully: Sentiment Aided Cyberbullying Detection with Explanation," arXiv preprint arXiv:2401.09023.
[6] R. Biswas et al. (2024), "Securing Social Spaces: Harnessing Deep Learning to Eradicate Cyberbullying," arXiv preprint arXiv:2404.03686.
[7] S. W. Azumah et al. (2024), "Deep Learning Approaches for Detecting Adversarial Cyberbullying and Hate Speech in Social Networks," arXiv preprint arXiv:2406.17793.
[8] N. Ejaz et al. (2024), "A Multi faceted Semi Synthetic Dataset for Automated CyberbullyingDetection," arXiv preprint arXiv:2402.10231.
[9] M. M. Krishnan et al. (2023), "Cyberbullying Detection Using NLP and Machine Learning: A Comparative Analysis," Computers in Human Behavior.
[10] A. Gupta et al. (2023), "A Transformer Based Approach for Cyberbullying Detection in Online Platforms," ACM Transactions on Internet Technology.
[11] N. S. Shah et al. (2023), "Multilingual Cyberbullying Detection Using BERT and Deep Neural Networks," Springer AI and Society Journal.
[12] J. A. Singh et al. (2023), "An Ensemble Learning Approach for Identifying Cyberbullying in Online Text," Elsevier Expert Systems with Applications.
[13] S. Alawad et al. (2023), "Explainable AI for Cyberbullying Detection in Online Conversations," Journal of Applied AI.
[14] R. Rajasekaran et al. (2023), "Sentiment Aware Deep Learning Model for Cyberbullying Identification," IEEE Access.
[15] M. J. Kaur et al. (2023), "Cyberbullying Prevention Through AI Based Moderation Systems," ACM CHI Conference on Human Factors in Computing Systems.
[16] S. Das et al. (2023), "BERT vs. LSTM: Evaluating Deep Learning Models for Detecting Cyberbullying," arXiv preprint.
[17] A. Halder et al. (2023), "Tackling Cyberbullying with Federated Learning and NLP," Springer Neural Computing and Applications.
[18] P.Kulkarni et al. (2023), "Transformers for Cyberbullying Detection: A Systematic Review and Benchmarking," IEEE Transactions on Neural Networks and Learning Systems.
[19] A. Akhter et al. (2023), "A Robust Hybrid Machine Learning Model for Bengali Cyberbullying Detection in Social Media," Natural Language Processing Journal, 4, 100027.
[20] V. L. Paruchuri and P. Rajesh (2023), "CyberNet: A Hybrid Deep CNN with N gram Feature Selection for Cyberbullying Detection in Online Social Networks,"

Evolutionary Intelligence, 16(6), 1935 1949.
[21] H. M. Abdulwahab and F. A. Ghanem (2023), "FAEO ECNN: Cyberbullying Detection in Social Media Platforms Using Topic Modelling and Deep Learning," Multimedia Tools and Applications, 82(30), 46611 46650.
[22] T. M. Mitchell et al. (2023), "Explainable AI and Cyberbullying Detection: Making AI Decisions Transparent," Journal of Ethics in AI Research.
[23] V. Sundar et al. (2022), "Automated Cyberbullying Detection Using LSTM and GRU Networks," Journal of Artificial Intelligence Research.
[24] J. Zhu et al. (2022), "Detecting and Preventing Cyberbullying Using Social Media Text Mining and NLP," IEEE Transactions on Big Data.
[25] P. Bidwai et al. (2022), "Hate Speech and Cyberbullying Detection Using Contextual Embeddings," Proceedings of ACL (Association for Computational Linguistics).
[26] S. Sridhar et al. (2022), "Early Detection of Cyberbullying on Twitter Using NLP and BiLSTM," International Conference on Web and Social Media (ICWSM).
[27] T. Mitchell et al. (2022), "A Multimodal Approach to Cyberbullying Detection Using Text and Image Analysis," ACM Multimedia Conference.
[28] D. Mann et al. (2022), "Cross Language Cyberbullying Detection Using Transfer Learning," Springer Journal of Computational Social Science.
[29] P. K. Roy and F. U. Mali (2022), "Cyberbullying Detection Using Deep Transfer Learning," Complex and Intelligent Systems, 8(6), 5449 5467.
[30] N. Lu et al. (2020), "Cyberbullying Detection in Social Media Text Based on Character Level Convolutional Neural Network with Shortcuts," Concurrency and Computation: Practice and Experience, 32(23), e5627.
[31] Dinakar, K., Reichart, R., & Lieberman, H. (2022). Modeling the detection of textual cyberbullying. Proceedings of the International AAAI Conference on Web and Social Media, 5(1), 11–17.
[32] Zhao, R., Zhou, A., & Mao, K. (2022). Automatic detection of cyberbullying on social networks based on bullying features. Proceedings of the 17th International Conference on Distributed Computing and Networking, 1–6.
[33] Rosa, H., Pereira, N., Ribeiro, R., Ferreira, P. C., Carvalho, J. P., & Oliveira, R. (2022). Automatic cyberbullying detection: A systematic review. Computers in Human Behavior, 93, 333–345.
[34] Salawu, S., He, Y., &Lumsden, J. (2022). A survey on cyberbullying detection. ACM Computing Surveys (CSUR), 53(3), 1–35.
[35] Devlin, J., Chang, M. W., Lee, K., &Toutanova, K. (2022). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805.