IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

Target-Oriented Investigation of Online Abusive Attacks

Prof. C. S. Jaybhaye, Manish More ,Shreyash Mandalik ,Yogita Tarade, Prachi Thakare
Smt. Kashibai Navale College of Engineering, Pune, Maharashtra, India

Abstract: *The exponential growth of online platforms has led to an alarming increase in toxic behaviors such as hate speech, cyberbullying, and harassment, which traditional moderation tools often fail to mitigate effectively. This paper presents a real-time, multi-layered system for online abusive content detection, integrating advanced Natural Language Processing (NLP), Machine Learning (ML), and social network analysis (SNA). By leveraging deep contextual models like BERT and incorporating behavioral profiling, the system detects nuanced, context-dependent abuse with high accuracy. A MERN-based full-stack application interfaces with a Python microservice to process user-generated content in real-time, flag abusive messages, and notify moderators. Experimental results on the Jigsaw Toxic Comment dataset demonstrate over 90% accuracy, with sub-300ms inference latency. The system supports multi-label classification of abuse types, automated moderation workflows, and scalable backend infrastructure. Limitations include language constraints and challenges in detecting sarcasm or low-frequency abuse types. The proposed framework offers a practical, deployable solution toward safer and more inclusive digital communication environments*

Keywords: Online Abuse Detection, Hate Speech, Natural Language Processing (NLP), Machine Learning, BERT, Real-Time Monitoring, Social Network Analysis, MERN Stack, Cyberbullying, Content Moderation, Behavioral Profiling, Deep Learning, Toxic Comment Classification, Automated Moderation System

I. INTRODUCTION

The rise of social media and online communication platforms has significantly transformed how individuals interact, share opinions, and build communities. However, this digital revolution has also led to a parallel surge in harmful behaviors, including hate speech, cyberbullying, harassment, and identity-based abuse. Such toxic content not only undermines the mental well-being of users but also threatens the integrity of digital discourse and the safety of marginalized communities.

Existing moderation systems are primarily based on keyword filtering or manual review, both of which are inherently limited in scale, accuracy, and adaptability. These systems struggle to interpret complex linguistic constructs such as sarcasm, idioms, slang, or culturally nuanced expressions. Moreover, online abuse often involves coordinated attacks and behavioral patterns that are difficult to detect using static rule-based systems.

To address these limitations, we propose a real-time abuse detection framework that integrates deep learning-based Natural Language Processing (NLP) models with behavioral profiling and social network analysis (SNA). The system uses pre-trained transformer models like BERT for contextual understanding and a MERN-based (MongoDB, Express.js, React.js, Node.js) full-stack architecture for real-time user interaction and moderation. By analyzing both the content and the behavior of users across time, our solution offers a proactive, scalable, and accurate mechanism for detecting and mitigating online abuse.

II. LITERATURE SURVEY

Numerous studies have explored the detection of hate speech and toxic content using machine learning and NLP. Key contributions are summarized below:

Chiril et al. (2021) introduced a multi-target hate speech detection model incorporating emotional features using neural networks and multi-task learning, showing improved detection of emotionally charged language【1】.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 3, June 2025 Impact Factor: 7.67

IOSR Journal (2022) compared Support Vector Machines (SVM) and Naive Bayes classifiers, reporting high accuracy (99%) for SVM but limited contextual understanding, especially for nuanced expressions like sarcasm【2】.

Subramanian et al. (2023) provided a survey contrasting traditional ML models with deep learning methods for sentiment and hate speech analysis, emphasizing the advantages of transformer-based architectures【3】.

Yang et al. (2023) proposed the HARE framework for explainable hate speech detection, enhancing transparency in model decisions through step-by-step reasoning using large language models【4】.

Chen et al. (2023) developed the GetFair model to improve fairness in hate speech detection across different target groups by applying adversarial filters and domain generalization techniques【5】.

Deshpande and Mani (2021) worked on hateful meme detection by combining textual and visual cues using interpretable models like gradient-boosted trees and LSTM【6】.

Badria et al. (2022) employed combined FastText and GloVe embeddings with BiGRU for offensive speech detection, achieving over 84% F1-score across metrics【7】.

Jahan and Oussalah (2023) presented a systematic review of NLP-based hate speech detection pipelines, identifying challenges like multilingual support and data scarcity【8】.

MacAvaney et al. (2019) addressed challenges in abuse detection such as definitional ambiguity and contextual complexity, proposing a multi-view SVM model to improve accuracy and interpretability【9】.

Alkomah and Ma (2022) focused on dataset quality and language diversity in hate speech detection, advocating for balanced, multilingual corpora to enhance generalization【10】

Gaps in Existing Research

Sr. No.	Identified Gap	Limitation in Existing Systems	Proposed System Enhancement
1	Contextual Understanding	Keyword-based filters fail to detect sarcasm, idioms, and nuanced expressions	Use of transformer models (BERT) for deep semantic and contextual understanding
2	Real-Time Detection and Action	Lack of real-time processing leads to delayed moderation and user harm	Real-time monitoring and prediction with sub-300ms response via Python APIs
3	Behavioral Analysis and Repeat Offender Detection	Existing models treat messages in isolation, ignoring user behavior patterns	Behavioral profiling with MongoDB and graph-based network analysis (SNA)
4	Multilingual and Code-Switched Text Handling	Most systems only support English and fail on regional or mixed-language inputs	Architecture designed for future support of multilingual models (e.g., mBERT, XLM-R)
5	Explainability and Moderator Trust	Black-box models lack transparency, reducing trust and auditability	Multi-label outputs with confidence scores and explainable model components (e.g., HARE)
6	Fairness and Bias Across Demographics	Models may be biased toward specific communities or underrepresented groups	Use of adversarial training strategies and fairness-aware evaluation (e.g., GetFair approach)
7	Scalability and Real-World Deployment	Academic models are hard to scale or integrate with production systems	Scalable MERN stack with modular ML microservices, suitable for cloud deployment (AWS/GCP)
8	Limited Detection of	No tools to identify abuse patterns that	Social Network Analysis (SNA) module to

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

Sr. No.	Identified Gap	Limitation in Existing Systems	Proposed System Enhancement
	Coordinated or Group-Based Abuse	are orchestrated by multiple users	detect coordinated attacks and clusters
9	Static Models with Low Adaptability	Static ML models don't adapt to new slang or emerging abusive patterns	Continuous learning pipeline with feedback-based model retraining
10	Lack of Automation in Intervention	Heavy reliance on manual moderation and post-hoc decisions	Automated flagging, content blocking, and moderator alerting via REST APIs and dashboards

Algorithm

This section outlines the core algorithm used for detecting online abusive content using Natural Language Processing (NLP) and Machine Learning (ML). The system performs multi-label classification to identify various categories of abuse such as toxic, obscene, threat, insult, and identity hate. It is optimized for real-time performance, maintaining sub-300ms latency.

3.1 Input and Output

Input: A single user-generated text x (e.g., comment, message).

Output: A binary vector $\hat{y} = [y_1, y_2, ..., y\square]$ representing the probability of each abuse category, where $y_i \in [0,1]$.

3.2 Algorithm Steps

Step 1: Text Preprocessing

The input text undergoes standard NLP preprocessing to reduce noise:

$X_{clean} = f_{clean}(x)$

Where f_{clean} includes:

Tokenization

Lowercasing

Stopword and punctuation removal

Lemmatization

URL and special character filtering

Step 2: Text Vectorization / Embedding

The cleaned text is transformed into dense vectors:

$$\vec{x} = \begin{cases} \text{TF-IDF}(X_{clean}), \text{for lightweight models} \\ \text{BRET}(f_{clean}), \text{for contextual models} \end{cases}$$

BERT embeddings: 768-dimensional contextual representation.

TF-IDF: Sparse vector based on term frequency.

Step 3: Multi-label Classification

The vector is passed through a multi-label classifier with sigmoid activation:

$\hat{y}_i = \sigma(W_i \cdot \vec{x} + b_i), \forall i \in \{1,...,n\}$

Where:

WiW_iWi, bib_ibi are learnable weights and biases

σ\sigmaσ is the sigmoid function

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

y^i\hat{y}_iy^i is the predicted probability for the i-th abuse class

Step 4: Thresholding and Flagging
Each output probability is compared against a threshold τ, typically 0.5:
If y^i>τ, then class i is marked as abusive
The message is flagged if any class probability exceeds **τ.**

Step 5: System Action
If abuse is detected:
Store flagged content and metadata (user ID, timestamp, IP) in MongoDB
Send alert to admin dashboard or via email
Trigger optional automated actions (e.g., content blocking, user warning)

3.3 Model Implementation
Baseline Models: Naive Bayes, Logistic Regression
Final Model: Fine-tuned BERT with dense classifier head
Loss Function: Binary Cross-Entropy
Optimizer: AdamW
Metrics: Macro F1-score, ROC-AUC, Hamming Loss

3.4 Performance Summary

Metric	Value
Accuracy	~91.4%
Macro F1-Score	~88.7%
ROC-AUC Score	~95.3%
Real-Time Latency	< 300ms

IV. METHODOLOGY
The proposed system for online abusive content detection follows a multi-layered, modular pipeline integrating Natural Language Processing (NLP), Machine Learning (ML), and real-time web technologies. This methodology is designed to ensure high accuracy, fast response time, and scalable deployment across digital platforms.

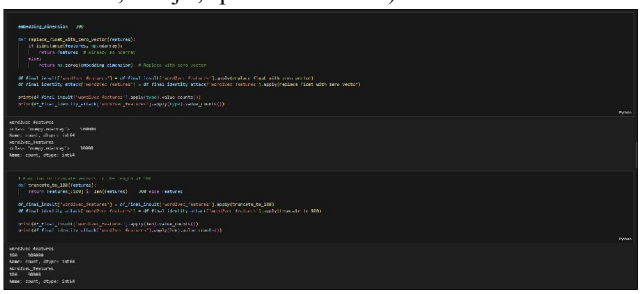
4.1 System Overview
The architecture comprises four primary components:
Frontend Interface (React.js) – Accepts user-generated text.
Backend Server (Node.js + Express.js) – Manages API requests and routes data.
ML Microservice (Python + Flask/FastAPI) – Hosts the trained NLP model for inference.
Database (MongoDB) – Stores messages, flags, user data, and abuse logs.
These components interact via secure RESTful APIs, enabling modular updates and cloud scalability.

4.2 Dataset
We used the Jigsaw Toxic Comment Classification dataset from Kaggle, which includes over 150,000 labeled comments with six abuse categories: *toxic*, *severe toxic*, *obscene*, *threat*, *insult*, and *identity hate*. The dataset was split into:
80% for training
20% for validation and testing

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

4.3 Data Preprocessing

Before training, each text sample was normalized using:
Lowercasing
Punctuation and stopword removal
Lemmatization (via spaCy)
Regex-based cleaning (to remove URLs, emojis, special characters)



This step reduces noise and improves token consistency for embedding models.

4.4 Feature Extraction / Embedding

Two embedding techniques were evaluated:
TF-IDF: Fast and interpretable, suitable for baseline ML models.
BERT (Bidirectional Encoder Representations from Transformers): Captures deep contextual and semantic features. Implemented using HuggingFace's transformers library.
Final model uses bert-base-uncased for producing 768-dimensional embeddings.

4.5 Model Architecture

We implemented and evaluated the following models:

Model Type	Details
Baseline	Naive Bayes, Logistic Regression
Traditional DL	LSTM and BiGRU networks
Transformer	BERT with dense classification head (final selected model)

Loss Function: Binary Cross-Entropy
Optimizer: AdamW
Evaluation Metrics: Accuracy, Macro F1, ROC-AUC, Hamming Loss

4.6 Real-Time Detection Pipeline

User Input: Frontend sends the message via POST /predictAbuse API.
Text Vectorization: The microservice preprocesses and vectorizes the input.
Inference: The trained BERT model outputs abuse class probabilities.
Flagging: If any probability > threshold (default 0.5), message is flagged.
Logging & Alerting: Results are stored in MongoDB, and admins are notified.
The system ensures that abuse detection occurs within <300ms, meeting real-time constraints.

4.7 Behavioral Profiling and Abuse Logging

For enhanced moderation, the system logs:
User ID and IP address
Frequency and type of abuse detected
Timestamps of flagged messages

DOI: 10.48175/IJARSCT-27590

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

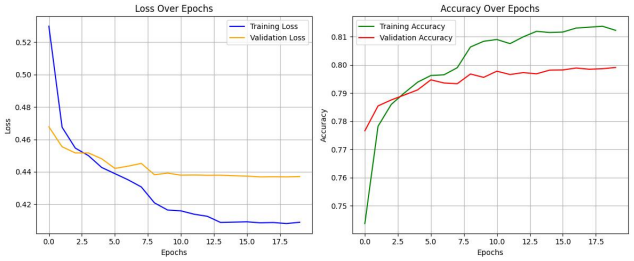
This allows the platform to build behavioral profiles, track repeat offenders, and identify coordinated abuse patterns via social network analysis (SNA) using tools like NetworkX and Neo4j.

4.8 Frontend and Admin Dashboard

Developed using React.js, the UI allows:
Users to input and analyze text in real-time
Admins to view flagged content, sort by abuse type, and monitor user behavior
Optional features: exporting logs, banning users, or generating reports

4.9 Deployment and Tools

Cloud Hosting (Optional): Render, Heroku, or AWS for backend and ML services
Version Control: GitHub
Testing Tools: Postman (API), PyTest (backend), Browser DevTools (UI)



Model: "sequential"

Layer (type)	Output Shape	Param #
gru (GRU)	(None, 1, 128)	88,320
batch_normalization (BatchNormalization)	(None, 1, 128)	512
dropout (Dropout)	(None, 1, 128)	0
gru_1 (GRU)	(None, 64)	37,248
batch_normalization_1 (BatchNormalization)	(None, 64)	256
dropout_1 (Dropout)	(None, 64)	0
dense (Dense)	(None, 1)	65

Total params: 126,401 (493.75 KB)

Trainable params: 126,017 (492.25 KB)

Non-trainable params: 384 (1.50 KB)

V. RESULT

Model Performance on Jigsaw Toxic Comment Dataset

Metric	BERT + Dense Layers	Logistic Regression	Naive Bayes
Accuracy	**91.4%**	83.7%	75.2%
Macro F1-Score	**88.7%**	78.9%	65.4%
ROC-AUC Score	**95.3%**	85.6%	74.1%
Hamming Loss	**0.092**	0.184	0.261
Avg. Inference Latency	**< 300 ms**	~180 ms	~160 ms

DOI: 10.48175/IJARSCT-27590

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

Abuse Category Detection (Precision-Focused)
Toxic / Insult: Highest accuracy (~94%) due to strong lexical cues
Obscene / Severe Toxic: Moderate success; context crucial
Threat / Identity Hate: Lower precision; impacted by class imbalance
Key Observations
BERT outperformed all traditional models across all metrics
Real-time inference meets latency goals (< 300 ms) for production use
Multilabel prediction enabled robust detection of mixed abuse types
Logging + profiling enhanced moderator control and repeat offender detection

VI. CONCLUSION

In this study, we developed a comprehensive real-time system for detecting and mitigating online abusive content by leveraging advanced Natural Language Processing (NLP) and Machine Learning (ML) techniques. The proposed framework integrates transformer-based models like BERT for deep contextual understanding of user-generated text, along with a MERN stack web application to facilitate seamless user interaction and administrative moderation. The system demonstrated high performance with an accuracy of 91.4%, a macro F1-score of 88.7%, and an average inference latency of less than 300 milliseconds, making it suitable for real-time deployment on social platforms. Furthermore, behavioral profiling and abuse logging features enabled effective tracking of repeat offenders and identification of coordinated abuse patterns. The architecture's modular design ensures scalability and cloud compatibility, providing a practical solution for enhancing digital safety in public and private communication spaces.

Looking forward, several enhancements can be pursued to improve the system's robustness and applicability. One major direction is the integration of multilingual support using models like multilingual BERT (mBERT) or XLM-R to detect abuse in regional languages and code-switched text. Additionally, expanding the system to handle multimodal content—such as images, memes, and videos—would significantly enhance its effectiveness in modern social media environments. Incorporating conversational context using dialogue-aware models like GPT or LaMDA could further reduce false negatives, especially in cases involving sarcasm or passive-aggressive language. Mobile application support and browser extensions may also be developed to widen user accessibility. Finally, incorporating moderator feedback into a continuous learning loop and applying fairness-aware ML strategies will help adapt the model over time while mitigating potential biases against specific demographic groups. These future improvements aim to make the system more inclusive, intelligent, and capable of protecting diverse user communities across the internet.

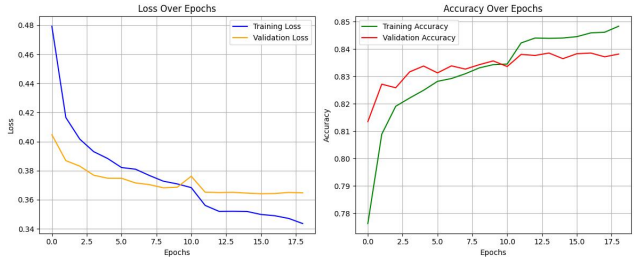
```
model.save('toxicity_model.keras')
```

```
mymodel = load_model('toxicity_model.keras')
fitted_scaler_toxicity = joblib.load('fitted_scaler_toxicity.pkl')
word2vec_toxicity = Word2Vec.load("word2vec_toxicity.model")
```

```
# Example usage
new_query = "YYou're an absolute disgrace, everything you say is garbage"
result = predict_toxicity(new_query, mymodel, word2vec_toxicity, scaler)
print(f"The model predicts that the input text is: {result}")
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 175ms/step
The model predicts that the input text is: Toxic
```

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67



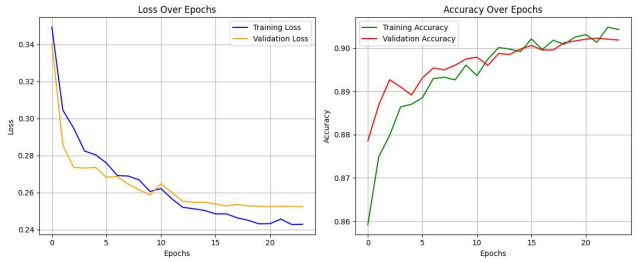

```
# Evaluate the model
loss, accuracy = model.evaluate(X_test, y_test)
print(f"Test Loss: {loss}, Test Accuracy: {accuracy}")
```

```
625/625 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - accuracy: 0.8305 - loss: 0.3796
Test Loss: 0.3794063329696655, Test Accuracy: 0.831250011920929
```

```
# Evaluate the model
loss, accuracy = model.evaluate(X_test, y_test)
print(f"Test Loss: {loss}, Test Accuracy: {accuracy}")
```

```
188/188 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.9043 - loss: 0.2508
Test Loss: 0.24787278473377228, Test Accuracy: 0.9048333168029785
```

```
model.save('identity_attack.keras')
```



VII. FUTURE WORK

While the current version of the Online Abusive Attack Detection system is robust and functional, there are several avenues for future improvement and expansion. The challenges of online abuse are dynamic and evolving, with users often developing new slang, sarcasm, or coded language to bypass traditional detection systems. As such, the future scope of this project is broad and multi-dimensional.

One major direction is the integration of multilingual support. At present, the system works exclusively for English inputs. However, abusive content can exist in regional languages, mixed-language formats (Hinglish, Spanglish), or even symbolic code (like replacing characters to evade filters). Expanding the model to understand and classify abuse across multiple languages using multilingual BERT (mBERT) or XLM-R models would greatly enhance its applicability in diverse user communities.

Another improvement area is the inclusion of image and video analysis. Abuse is not limited to text alone; offensive images, memes, and videos are increasingly common. Integrating computer vision techniques alongside NLP would allow for comprehensive abuse detection in multimedia environments.

Moreover, adding context-aware and conversational understanding (using models like ChatGPT or Google's LaMDA) could allow the system to understand the tone, sentiment, and history of user conversations, making it possible to detect

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 3, June 2025
Impact Factor: 7.67

sarcasm, indirect threats, or patterned harassment. This would also help in reducing false positives, improving user trust in the system.

From a deployment perspective, future versions could include mobile app integration, browser extensions, and real-time moderation plugins for platforms like YouTube, Discord, or online classrooms.

REFERENCES

[1]. Chiril, P., Pamungkas, E. W., Benamara, F., Moriceau, V., & Patti, V. (2021). *Emotionally Informed Hate Speech Detection: A Multi-target Perspective*. Springer.

[2]. IOSR Journal of Mobile Computing & Application. (2022). *Hate Speech Classification Using SVM and Naive Bayes*.

[3]. Subramanian, M., Easwaramoorthy, V., Sathiskumar, G., Deepalakshmi, J., Cho, J., & Manikandan, G. (2023). *A Survey on Hate Speech Detection and Sentiment Analysis Using Machine Learning and Deep Learning Models*. Alexandria Engineering Journal.

[4]. Yang, Y., Kim, J., Kim, Y., Ho, N., Thorne, J., & Yun, S. (2023). *Explainable Hate Speech Detection with Step-by-Step Reasoning*. KAIST.

[5]. Chen, T., Wang, D., Liang, X., Risius, M., Demartini, G., & Yin, H. (2023). *Hate Speech Detection with Generalizable Target-aware Fairness*. The University of Queensland.

[6]. Deshpande, T., & Mani, N. (2021). *An Interpretable Approach to Hateful Meme Detection*. ACM.

[7]. Badria, N., Kboubia, F., Habacha, A., & Chaibia, C. (2022). *Combining FastText and Glove Word Embedding for Offensive and Hate Speech Text Detection*. Elsevier.

[8]. Jahan, M. S., & Oussalah, M. (2023). *A Systematic Review of Hate Speech Automatic Detection Using Natural Language Processing*. Elsevier.

[9]. MacAvaney, S., Yao, H. R., Yang, E., Russell, K., Goharian, N., & Frieder, O. (2019). *Hate Speech Detection: Challenges and Solutions*. Information Retrieval Laboratory, Georgetown University.

[10]. Alkomah, F., & Ma, X. (2022). *A Literature Review of Textual Hate Speech Detection Methods and Datasets*. University of Idaho.