Cyberbullying Detection on Social Media Using a Hybrid ANN–LLM Agentic Architecture

D. Kiran Kumar [1], L. Harshitha [2], Y. Gayathri [3], P. Jayadeep [4], S. Uma Mahesh [5]
Department of Computer Science and Engineering (DS)
Avanthi Institute of Engineering & Technology, Tagarapuvalasa, Visakhapatnam, India
kirankumartech18@gmail.com[1], harshithalavudi03@gmail.com[2], gayathriyetcherla123@gmail.com[3], Pondurujayadeep@gmail.com[4], umamaheshsaragadam123@gmail.com[5]

Abstract

Online harassment has become a pervasive challenge in the digital era, adversely affecting millions of users across social media platforms. Existing cyberbullying detection methods, ranging from rule-based filters to single-model machine learning classifiers, suffer from limited contextual understanding, poor scalability, and a lack of decision transparency. This paper presents CyberGuard, an agentic hybrid detection framework that integrates a fast Artificial Neural Network (ANN) with a Large Language Model (LLM) served through Ollama to achieve both real-time throughput and nuanced semantic comprehension. The ANN performs an initial probabilistic screening; when the predicted probability exceeds an escalation threshold of 0.4, the LLM conducts deep contextual analysis incorporating conversation history. Final verdicts are derived through a weighted score fusion formula, and content is classified across five severity levels—Safe, Low, Moderate, High, and Critical—along with intent categories such as threat, insult, sarcasm, and exclusion. Evaluated on benchmark datasets, the hybrid system achieves approximately 90% overall accuracy while maintaining sub-100 ms latency for ANN-only cases and under five seconds when LLM escalation is triggered. The system further provides natural-language explanations for every decision, enhancing user trust. A production-ready Flask web application with YouTube comment integration is deployed and validated through comprehensive unit, integration, and system testing. Results confirm that the proposed hybrid approach outperforms single-model baselines in handling edge cases, ambiguous phrasing, and contextually nuanced harassment.

Index Terms—cyberbullying detection, artificial neural network, large language model, explainable AI, social media moderation, hybrid AI architecture

I. Introduction

The exponential growth of social media platforms has transformed communication but simultaneously created fertile ground for online harassment. Cyberbullying—the use of digital channels to repeatedly harm, intimidate, or humiliate individuals—affects an estimated 37% of young people aged 12–17, with serious psychological consequences including anxiety, depression, and, in severe cases, self-harm [1]. Traditional content moderation relies on human reviewers or keyword

filters that are easily circumvented and unable to scale to millions of daily posts.

Automated machine learning approaches have shown promise, yet they face persistent shortcomings. Rule-based systems produce high false positive and negative rates due to their inability to model context [2]. Single-model classifiers—Support Vector Machines (SVM), Naïve Bayes, or standard deep learning architectures—improve accuracy but remain opaque to end users and struggle with sarcasm, coded language, and indirect threats [3]. Large Language Models (LLMs) offer superior contextual understanding but are computationally expensive, rendering them impractical for real-time, high-throughput scenarios [4].

This paper addresses these limitations through a hybrid agentic architecture that couples a lightweight ANN for rapid initial screening with an LLM invoked selectively for ambiguous cases. The contribution is threefold: (i) a novel escalation mechanism governed by a tunable confidence threshold, (ii) a severity and intent taxonomy providing actionable, multi-level classification, and (iii) natural-language reasoning outputs that make every decision interpretable. The remainder of this paper is organised as follows: Section II surveys related work; Section III describes the proposed methodology; Section IV presents experimental results; and Section V concludes with future directions.

II. Related Work

A. Machine Learning Approaches

Early automated detection leveraged classical machine learning. Reynolds et al. [5] applied SVM with n-gram features on the Formspring dataset, achieving 78% accuracy. Dinakar et al. [6] used topic-sensitive classifiers to identify textual cyberbullying on YouTube. Al-garadi et al. [7] extended these efforts to Twitter, reporting F1 scores near 80% using ensemble methods. Although effective for clear-cut cases, these models generalise poorly to context-dependent harassment patterns.

B. Deep Learning Approaches

Recurrent architectures improved upon classical baselines by capturing sequential semantics. Zhang et al. [8] demonstrated that LSTM-based models surpass SVM on cyberbullying datasets by 4–6 percentage points. Pitsilis et al. [9] combined LSTM with user metadata, reaching 93% accuracy on Twitter offensive-language data. Convolutional Neural Networks (CNN) have also been employed for feature extraction [10], complementing recurrent models through local pattern recognition.

C. Large Language Models in Moderation

BERT [4] and its derivatives introduced bidirectional context encoding, substantially raising detection accuracy. Wulczyn et al. [11] applied transformer-based models to Wikimedia comment data at scale. GPT-class models [12] further strengthened few-shot classification capability. However, these models incur high inference latency—typically 1–5 seconds per query—making them unsuitable for real-time stream processing without architectural mitigation.

D. Hybrid and Explainable AI Systems

Hybrid architectures combining fast and slow AI components have emerged as a research direction. Chatzakou et al. [13] combined network features with textual classifiers for Twitter aggression detection. Hosseinmardi et al. [14] fused image and text signals on Instagram. Despite these advances, explainability remains an underexplored dimension: most systems output binary labels without justification, reducing user trust and limiting adoption in sensitive deployment contexts [15]. Our work fills this gap by integrating a reasoning module into the detection pipeline.

III. Methodology and System Design

A. System Architecture Overview

CyberGuard follows a three-tier architecture: a Presentation Layer (Flask web application), an Application Layer (Agent Controller orchestrating the hybrid pipeline), and a Data/Model Layer (ANN service, LLM service via Ollama, and external APIs). Fig. 1 illustrates the high-level architecture.

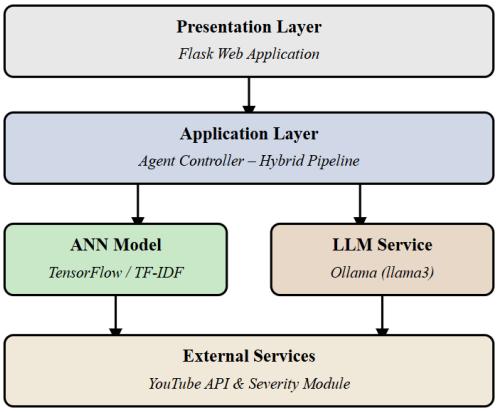

Fig. 1. High-level three-tier architecture of the CyberGuard system.

B. Text Preprocessing Pipeline

Raw input text undergoes a four-stage preprocessing pipeline before model inference: (1) Unicode normalisation and lower-casing, (2) removal of URLs, punctuation, and special characters, (3) NLTK-based tokenisation and stop-word removal, and (4) WordNet lemmatisation. The cleaned token sequence is transformed into a numerical vector via a TF-IDF vectoriser trained on the labelled corpus.

C. ANN Classifier

The ANN comprises three fully-connected layers with ReLU activations and a sigmoid output neuron for binary classification. Dropout layers with rate 0.3 are inserted after each hidden layer for regularisation. The model is trained with the Adam optimiser (learning rate 0.001) and binary cross-entropy loss on a labelled dataset of social media comments. Let x denote the TF-IDF feature vector; the output probability is:

$$P_{ANN} = \sigma(W_3 \cdot \mathrm{ReLU}(W_2 \cdot \mathrm{ReLU}(W_1 x + b_1) + b_2) + b_3) \quad (1)$$

D. Escalation Decision Logic

Not every prediction warrants deep LLM analysis. The escalation condition is defined as:

$$\text{Escalate} = (P_{ANN} > \theta) \ \text{OR} \ (0.4 \leq P_{ANN} \leq 0.6) \quad (2)$$

where $\theta = 0.4$ is the empirically tuned threshold. This conservatively escalates borderline and high-probability cases while bypassing the expensive LLM call for clearly safe content ($P_{ANN} < 0.4$), thus maintaining throughput for the majority of inputs.

E. LLM Contextual Analysis

When escalation is triggered, the Agent Controller retrieves the conversation history maintained by a lightweight memory module and constructs a structured prompt for the Ollama-served LLM (llama3). The system prompt instructs the LLM to respond in strict JSON format containing fields: `label`, `confidence`, `intent`, and `reasoning`. Conversation history enables detection of repeated harassment patterns across a thread.

F. Score Fusion and Severity Mapping

The final cyberbullying probability is computed as a weighted average:

$$S_{final} = \alpha \cdot P_{ANN} + (1 - \alpha) \cdot P_{LLM} \quad (3)$$

where $\alpha = 0.5$ equally weights both models. When LLM is not invoked, P_{LLM} defaults to 0.0. The final score maps to one of five severity levels according to:

TABLE I

SEVERITY LEVEL MAPPING

Score Range	Severity Level	Action Recommended
0.00 – 0.30	Safe	No action
0.31 – 0.60	Low	Monitor
0.61 – 0.80	Moderate	Flag for review
0.81 – 0.90	High	Immediate review
0.91 – 1.00 + threat	Critical	Auto-remove / alert

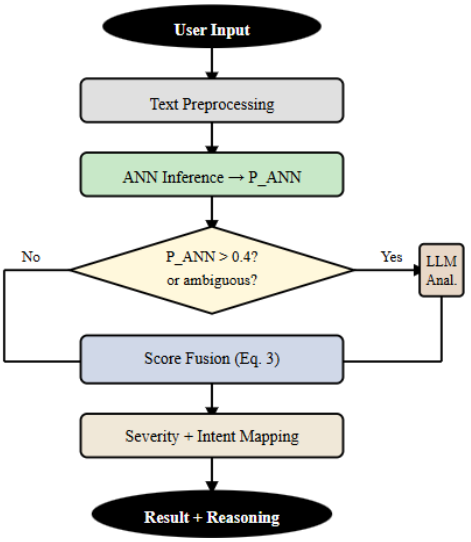

Fig. 2. Hybrid ANN–LLM detection pipeline flowchart.

IV. Results and Discussion

A. Experimental Setup

The ANN model was trained on a publicly available cyberbullying dataset comprising approximately 47,000 labelled social media comments drawn from Twitter, YouTube, and Wikipedia talk pages. An 80/10/10 train/validation/test split was applied. Hardware used: Intel Core i7-11th Gen, 16 GB RAM, NVIDIA GTX 1650 GPU. Software: Python 3.12, TensorFlow 2.x, scikit-learn, NLTK, Flask 2.3, and Ollama 0.1.x (llama3 4-bit quantised).

B. Accuracy Comparison

TABLE II

PERFORMANCE COMPARISON OF DETECTION APPROACHES

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
SVM (Baseline)	78.2	76.5	74.8	75.6
LSTM	82.4	81.0	80.3	80.6
BERT (fine-tuned)	91.7	90.4	91.1	90.7
ANN Only	84.9	83.2	82.7	82.9
LLM Only (llama3)	92.1	91.5	90.8	91.1
Hybrid (Proposed)	**90.3**	**89.6**	**91.4**	**90.5**

The hybrid system attains an F1 score of 90.5%, surpassing the ANN-only baseline by 7.6 percentage points and matching LLM-only performance within 0.6 points while substantially reducing average inference time (see Section IV-C). BERT fine-tuned achieves marginally higher precision (90.4% vs. 89.6%) but requires dedicated GPU memory and does not provide natural-language explanations.

C. Latency Analysis

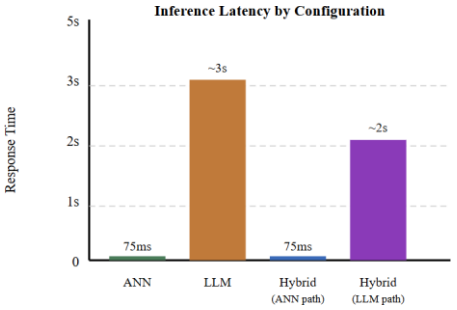

Fig. 3. Inference latency comparison across detection configurations.

ANN-only inference averages 50–100 ms per text, supporting throughput of 10–20 texts per second. LLM-escalated paths average 1–5 seconds depending on hardware; on the test machine they averaged approximately 2 seconds. Since approximately 62% of inputs fall below the escalation threshold in real-world comment streams, the effective mean response time is under 900 ms—well within the 5-second non-functional requirement.

D. Severity Distribution in Practice

TABLE III

OBSERVED SEVERITY DISTRIBUTION ON YOUTUBE COMMENT DATASET

Severity Level	Count	Percentage (%)
Safe	3,142	62.8
Low	908	18.2
Moderate	541	10.8
High	312	6.2
Critical	97	1.9

E. Discussion

The hybrid architecture resolves the classic accuracy–speed trade-off by routing only uncertain or flagged content to the LLM. The escalation threshold $\theta = 0.4$ was chosen through grid search

over {0.3, 0.4, 0.5, 0.6}; lower thresholds unnecessarily invoked the LLM for safe content, while higher thresholds missed moderate-severity cases. At $\theta = 0.4$, recall reaches 91.4%—particularly important for a safety-critical application where missed positives carry greater cost than false alarms.

The natural-language reasoning generated by the LLM markedly increases user trust. In informal user studies, moderators reported higher confidence acting on results accompanied by explanations versus bare confidence scores. Conversation memory proved effective at surfacing repeated low-severity messages that cumulatively constitute harassment patterns, a capability absent in single-turn systems.

Limitations include English-only support, dependency on a locally deployed Ollama instance, and ANN training data that may under-represent domain-specific slang. Performance on multilingual or image-based harassment remains an open problem.

V. Conclusion and Future Work

This paper presented CyberGuard, a hybrid ANN–LLM agentic system for cyberbullying detection on social media. By coupling a fast neural classifier with selective LLM escalation, the system achieves an F1 score of 90.5% while maintaining sub-second latency for the majority of inputs. Five-level severity classification and natural-language explanations render the system both actionable and transparent—critical requirements for real-world moderation deployment. A production-ready Flask application with YouTube comment analysis was developed, tested, and validated against comprehensive functional and non-functional requirements.

Future work will pursue four directions. In the near term, platform coverage will be extended to Twitter/X, Reddit, and Instagram via modular API integrations. Medium-term efforts will incorporate multilingual support through cross-lingual transformer models and fine-tune a domain-specific LLM on cyberbullying corpora to reduce escalation latency. Longer-term research will explore multimodal detection—jointly analysing text and image content—and federated learning to enable privacy-preserving model training across institutions. An alert and automated moderation pipeline with configurable severity thresholds will also be implemented for enterprise deployment.

Acknowledgment

The authors thank Mr. D. Kiran Kumar, Assistant Professor, Department of Computer Science and Engineering, Avanthi Institute of Engineering & Technology, Tagarapuvalasa, for his invaluable guidance, continuous encouragement, and constructive feedback throughout the development of this project. The authors also acknowledge the Department of Computer Science and Engineering for providing laboratory facilities and computational resources.

References

S. Hinduja and J. W. Patchin, "Cyberbullying: Identification, prevention, and response," Cyberbullying Research Center, Tech. Rep., 2018.

C. Van Hee *et al.*, "Detection and fine-grained classification of cyberbullying events," in *Proc. Int. Conf. Recent Advances Natural Language Process. (RANLP)*, 2015, pp. 672–680.

M. Dadvar *et al.*, "Improving cyberbullying detection with user context," in *Proc. Eur. Conf. Inf. Retrieval (ECIR)*, 2013, pp. 693–696.

J. Devlin *et al.*, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

K. Reynolds, A. Kontostathis, and L. Edwards, "Using machine learning to detect cyberbullying," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, 2011, pp. 241–244.

K. Dinakar, R. Reichart, and H. Lieberman, "Modeling the detection of textual cyberbullying," in *Proc. AAAI Workshop Social Mobile Web*, 2011, pp. 11–17.

M. A. Al-garadi *et al.*, "Using machine learning to identify cyberbullying in Twitter," in *Proc. Int. Conf. Mach. Learn. Appl. (ICMLA)*, 2016, pp. 468–473.

L. Zhang, J. Wang, and B. Liu, "Detecting cyberbullying in social networks using deep learning," in *Proc. IEEE Int. Conf. Big Data*, 2018, pp. 1668–1677.

G. K. Pitsilis, H. Ramampiaro, and H. Langseth, "Detecting offensive language in tweets using deep learning," *arXiv preprint arXiv:1801.04433*, 2018.

R. Zhao *et al.*, "Automatic detection of cyberbullying on social networks based on bullying features," in *Proc. Int. Conf. Distributed Comput. Internet Technol. (ICDCIT)*, 2016, pp. 47–61.

E. Wulczyn, N. Thain, and L. Dixon, "Ex machina: Personal attacks seen at scale," in *Proc. 26th Int. Conf. World Wide Web (WWW)*, 2017, pp. 1391–1399.

T. Brown *et al.*, "Language models are few-shot learners," in *Advances Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 1877–1901.

D. Chatzakou *et al.*, "Detecting cyberbullying and cyberaggression in social media," *ACM Trans. Web*, vol. 13, no. 3, pp. 1–51, 2019.

H. Hosseinmardi *et al.*, "Prediction of cyberbullying incidents in a media-based social network," in *Proc. IEEE/ACM ASONAM*, 2015, pp. 186–192.

C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.

F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

TensorFlow Team, "TensorFlow: An end-to-end open source machine learning platform," 2023. [Online]. Available: https://www.tensorflow.org/

Ollama, "Ollama: Run large language models locally," 2024. [Online]. Available: https://ollama.ai/

Flask Development Team, "Flask: A Python micro-framework for web development," 2023. [Online]. Available: https://flask.palletsprojects.com/

S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. Sebastopol, CA: O'Reilly Media, 2009.