# UnityAI-Guard : Pioneering Toxicity Detection Across Low-Resource Indian Languages

**WARNING: The content contains samples that are offensive and toxic.**

**Himanshu Beniwal**[♡*], **Reddybathuni Venkat**[♡*], **Rohit Kumar**[♣*], **Birudugadda Srivibhav**[♡*], **Daksh Jain**[♡*], **Pavan Doddi**[♡], **Eshwar Dhande**[♡], **Adithya Ananth**[◇], **Kuldeep**[♡], **Mayank Singh**[♡]

[♡]Indian Institute of Technology Gandhinagar, [♣]Indian Institute of Technology Goa, [◇]Indian Institute of Technology Tirupati

**Correspondence:** lingo@iitgn.ac.in

## Abstract

This work introduces UnityAI-Guard, a framework for binary toxicity classification targeting low-resource Indian languages. While existing systems predominantly cater to high-resource languages, UnityAI-Guard addresses this critical gap by developing state-of-the-art models for identifying toxic content across diverse Brahmic/Indic scripts. Our approach achieves an impressive average F1-score of 84.23% across six languages, leveraging a dataset of 567k training instances and 30k manually verified test instances. By advancing multilingual content moderation for linguistically diverse regions, UnityAI-Guard also provides public API access to foster broader adoption and application.

-Website UnityAI-Guard
-Demo bit.ly/UnityAI-Guard


Figure 1: Different examples of toxic prompts in four different scripts. ***Takeaway***: UnityAI-Guard *detects toxicity in six languages, transliteration for ease, and speech recognition for better usability.*

## 1 Introduction

India represents one of the most linguistically diverse regions in the world, with 22 officially recognized languages and hundreds of dialects (Singh et al., 2020; Gala et al., 2023). Among these, Hindi leads with approximately 528 million speakers, while Telugu (81 million), Marathi (83 million), Punjabi (33 million), and Urdu (51 million) each represent significant linguistic communities[1] (Kakwani et al., 2020; Pakray et al., 2025). Despite their large speaker bases, most of these languages remain computationally low-resourced, especially in specialized NLP applications like content moderation (Dongare, 2024; Narayan et al., 2023; Hegde et al., 2023). This has created vibrant low-resource language users but simultaneously presents significant challenges in content moderation and user safety (Costa-jussà et al., 2023; Jain et al., 2024; Rahman et al., 2024).

Online toxicity—including hate speech, abusive language, and harassment—has emerged as a critical concern across social platforms (Aodhora et al., 2025). These languages face multiple constraints that hinder effective NLP implementations: limited digitized corpora, scarce annotated datasets for toxicity, inadequate linguistic resources, and minimal pre-trained language models (Maity et al., 2024; Khan et al., 2024; Gala et al., 2023; LekshmiAmmal et al., 2022). Furthermore, linguistic phenomena such as code-mixing with English and other regional languages, script variations (particularly in Urdu), and dialectal differences create additional complications for toxicity detection systems (Saeed et al., 2021). Lastly, a popular tool such as "*Perspective-API*"[2] is still available in 17 widely spoken languages and only supports Hindi

[*]Equal Contribution

[1]https://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers_in_India

[2]https://perspectiveapi.com/



*Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 471–479
November 4-9, 2025 

| | Train | | Test | |
|---|---|---|---|---|
| **Lang.** | **Tox.** | **Neu.** | **Tox.** | **Neu.** |
| *Hindi* | 39691 | 32473 | 2500 | 2500 |
| *Telugu* | 62309 | 95282 | 2500 | 2500 |
| *Marathi* | 18092 | 20079 | 2500 | 2500 |
| *Urdu* | 32690 | 32690 | 2500 | 2500 |
| *Punjabi* | 18186 | 16159 | 2648 | 2352 |
| *Tamil* | 100000 | 100000 | 2500 | 2500 |

Table 1: The dataset split and the balance for the toxic and neutral pairs. Note that "Tox" and "Neu" represent the number of toxic and neutral instances, respectively. ***Takeaway***: *We keep at least 5k manually-verified test pairs for the evaluation.*

and Hinglish, not other Indian languages. However, even `llama-guard-3-8B` (Dubey et al., 2024) only supports Hindi as the Indian language, out of its support for eight languages.

To address this critical research gap, we introduce UNITYAI-GUARD, a comprehensive framework that supports six languages widely spoken in the Indian subcontinent with multiple functionalities: toxicity classification, transliteration capabilities, speech recognition integration, and programmatic access through API endpoints. We define the entire architecture in Figure 2, where we illustrate the front-end and back-end of the framework. More details in Section 4.

**Contributions** Our contributions can be summarized as follows:

- We introduce state-of-the-art binary toxicity classification models across six languages with diverse scripts.
- We created the largest toxicity dataset comprising 567,651 training instances and 30k test instances that underwent rigorous quality assurance through manual verification of all samples by two native speaker annotators per language, ensuring authentic content rather than synthetically generated data.
- We present a comprehensive web portal equipped with accessibility features, including transliteration support, speech recognition capabilities, and programmatic access via API endpoints.

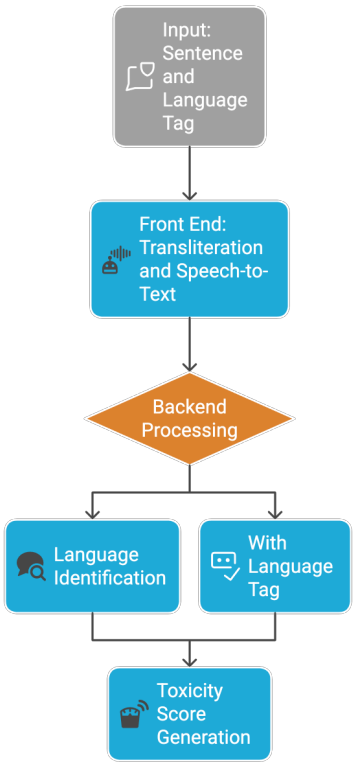


Figure 2: Architectural view for UNITYAI-GUARD. ***Takeaway***: *Front-end has speech-to-text and transliteration, whereas backend detects toxicity with/without language tags.*

## 2 Related Works

### 2.1 Content Moderation Datasets

Content moderation has been extensively studied in high-resource languages such as English, German, and French, with seminal work by Ye et al. (2023) and Wang et al. (2025) establishing foundational datasets for detecting hate speech in English. For non-English European languages, Farhan (2025) presented cross-lingual approaches for Italian and Spanish moderation.

In contrast to high-resource languages, content moderation for Indian languages has received comparatively limited attention. Bohra et al. (2018) and Srivastava (2025) explored the detection of hate speech in Hindi-English code-mixed text, while Garain et al. (2021), Raihan et al. (2023), and Kedia and Nandy (2021) presented benchmark datasets for the identification of offensive languages in Dravidian languages.

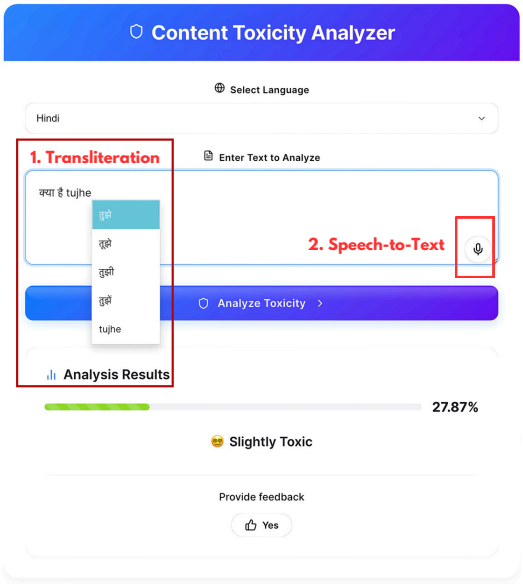


Figure 3: Two major add-ons for the UI improvment. ***Takeaway***: *We present transliteration and speech-to-text for the non-native speakers.*

### 2.2 Existing Systems and Tools for Content Moderation in Indian Languages

A popular tool named "*Perspective-API*" has gained significant traction but restricts support to only Hindi and Hinglish. Kakwani et al. (2020) introduced the IndicNLPSuite and Khan et al. (2024) presented Indic-align; however, both approaches rely on synthetically generated toxic content, which fails to capture the complex nuances of naturally occurring harmful language. The previous works by Saikh et al. (2024) and Raihan et al. (2023) focus on fine-tuned models; they lack support for other languages and do not provide the UI and API-support (Saeed et al., 2021; LekshmiAmmal et al., 2022; Kedia and Nandy, 2021).

### 2.3 Research Limitations and Our Contributions

Previous research efforts in content moderation for Indian languages have been constrained by limited language coverage and insufficient manually verified data (Gala et al., 2023; Dongare, 2024; Kedia and Nandy, 2021; Rahman et al., 2024). To address this limitation, we propose comprehensive datasets and models for low-resource Indian languages with particular emphasis on diverse scripts, including Tamil, Punjabi, and Marathi, which have remained largely unexplored in previous research efforts. Our work addresses these limitations by contributing a substantial corpus of manually inspected samples across multiple low-resource languages. We also conduct a comparative analysis of classification performance across models of varying capacity.

Our contribution represents the first large-scale dataset with manual verification of test samples, establishing a more reliable benchmark for evaluating content moderation models across Indian languages. Additionally, our work provides users and developers with an easy-to-use setup and comprehensive API support.

## 3 Experiments

### 3.1 Dataset

For dataset construction, we curated online sources[3] for six low-resource Indian languages: Hindi (hi), Telugu (te), Marathi (mr), Urdu (ur), Tamil (ta), and Punjabi (pa). The selection criteria were based on resource scarcity and script diversity, as each language employs a distinctly different writing system. In addition, we account for the transliteration and code-mixing phenomena between these languages and English. The distribution of toxic and neutral pairs between the train test splits is presented in Table 1. Note that given the budget constraints, only the test set (30,000 instances) was formulated by manual inspection, where we had 567,651 instances in the training set. We plan to release the dataset under the CC-BY-4.0 license.

**Quality Control** The dataset underwent annotation by two native speakers of each language. These annotators were unpaid volunteer students who participated in curating a comprehensive toxicity-labeled dataset over a 30-day period. Annotators were instructed to classify content as toxic if it aligned with any of the categories defined by `Llama-Guard-3-8B`[4]. Additionally, we implemented a keyword-based approach whereby content containing specific toxic terms was systematically categorized as toxic.

### 3.2 Models

To demonstrate variation in understanding low-resource languages with diverse scripts, we employ three differently sized models: ***(1)*** `mbert-base-uncased` (560M parameters, (Devlin

[3]The dataset is curated from News Channels, Keyword-based-searches, and licensed sources only. More details in the Appendix §A.1

[4]https://huggingface.co/meta-llama/Llama-Guard-3-8B

et al., 2019)), ***(2)*** `llama-3.2-1B` (1B parameters, (Dubey et al., 2024)), and ***(3)*** `aya-expanse-8B` (8B parameters, (Dang et al., 2024)). This selection represents a strategic gradient of model capacities, allowing us to analyze how parameter count correlates with cross-lingual understanding in Indic languages. We train the models with the classification task, where text is passed as the input and label as the output.

We present the zero-shot evaluation results in Table 3. The models demonstrate significantly diminished performance in zero-shot settings. For these experiments, inference was conducted by simply providing the input sentence and using the generated output as the predicted label. We aim to release the models under the MIT license.

### 3.3 Metrics

For evaluation, we utilize standard classification metrics, including Precision (P), Recall (R), F1 score (F1), and Accuracy (A), to assess model performance across the six target languages.

The experiments demonstrate that `aya-expanse-8b` achieves superior accuracy (87.21%) compared to `mbert-base` (83.40%), as detailed in Table 2. Similarly, we observe a consistent improvement in F1 scores as the model size increases, with values of 83.67%, 84.28%, and 86.96%, respectively. These results indicate that larger models demonstrate enhanced capability in processing and understanding diverse Indian scripts than the zero-shot models.

## 4 System Architecture

### 4.1 Front-End

We have designed an intuitive user interface that enables experimentation with our language-specific models. As illustrated in Figure 3, the system incorporates supplementary functionalities: ***(1)*** *Transliteration*[5] capabilities to facilitate script conversion, and ***(2)*** *Speech-to-Text*[6] functionality to enhance user accessibility.

**Feedback Collection** We also collect optional user feedback in a structured form as illustrated in Figure 4. The feedback mechanism captures multiple evaluation dimensions, including the user's input text, language identification, assigned toxicity score, score relevance assessment, user-selected content category, and a quantitative rating of model response quality on a 10-point scale. This data is systematically archived via Google Sheets integration for subsequent analysis. To maintain methodological consistency with established benchmarks as implemented in `llama-guard-3-8B` (Dubey et al., 2024), we adopt the same 14 taxonomic categories for content classification.

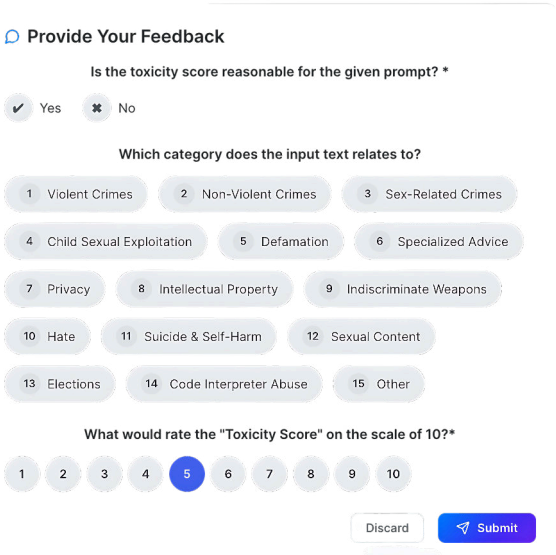


Figure 4: Feedback form over the user's input and model's output. ***Takeaway***: *We capture the feedback over the input and output using the Google Sheets integration.*

### 4.2 Back-End

We also deploy the trained models using FastAPI[7]. We show a sample code snippet to call the models by passing the input prompt and the language tag. It also showcases the output in a JSON format with the toxicity score and confidence score[8]. We keep the rate limit of 60 requests per minute; after that, the message with the rate limit is popped on.

[5] `https://www.npmjs.com/package/react-transliterate`

[6] `https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API`

[7] `https://github.com/fastapi/fastapi/`

[8] More details are available here: `https://lingo.iitgn.ac.in/unityai-guard/#/docs`

| | mBERT | | | | llama-3.2-1B | | | | aya-expanse-8b | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Language** | **P** | **R** | **F1** | **A** | **P** | **R** | **F1** | **A** | **P** | **R** | **F1** | **A** |
| *Hindi* | 83.04 | 81.97 | 82.50 | 82.63 | 83.81 | 84.21 | 84.01 | 83.99 | 90.30 | 82.05 | 85.98 | 86.63 |
| *Telugu* | 93.23 | 83.12 | 87.88 | 88.54 | 93.69 | 82.52 | 87.75 | 88.48 | 96.72 | 79.04 | 86.99 | 88.18 |
| *Marathi* | 85.47 | 85.23 | 85.35 | 85.38 | 89.81 | 88.80 | 89.31 | 89.37 | 91.05 | 87.36 | 89.17 | 89.39 |
| *Urdu* | 78.92 | 85.04 | 81.87 | 81.17 | 86.40 | 81.75 | 84.01 | 84.45 | 84.32 | 91.03 | 87.55 | 87.06 |
| *Punjabi* | 77.15 | 81.71 | 79.37 | 77.43 | 81.40 | 81.52 | 81.46 | 80.29 | 81.73 | 80.65 | 81.19 | 80.14 |
| *Tamil* | 80.08 | 83.48 | 81.75 | 81.36 | 76.15 | 81.88 | 78.91 | 78.12 | 83.79 | 74 | 78.59 | 79.84 |

Table 2: Accuracy scores (in percentage) over different languages using the three *fine-tuned* models. We report performance metrics as Precision (P), Recall (R), Accuracy (A), and F1. ***Takeaway***: *We observe that the smaller and larger LMs are very comparable.*

| | mBERT | | | | llama-3.2-1B | | | | aya-expanse-8b | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Language** | **P** | **R** | **F1** | **A** | **P** | **R** | **F1** | **A** | **P** | **R** | **F1** | **A** |
| *Hindi* | 50.85 | 86.89 | 64.16 | 52.14 | 48.52 | 23.75 | 31.89 | 49.31 | 69.09 | 78.81 | 73.63 | 71.80 |
| *Telugu* | 52.85 | 97.40 | 68.52 | 55.26 | 54.03 | 22.76 | 32.03 | 51.70 | 59.13 | 52.08 | 55.38 | 58.04 |
| *Marathi* | 50.72 | 95.91 | 66.35 | 51.40 | 53.24 | 31.90 | 39.90 | 51.98 | 59.98 | 70.79 | 64.94 | 61.81 |
| *Urdu* | 51.40 | 97.10 | 67.22 | 52.67 | 53.80 | 26.73 | 35.72 | 51.90 | 63.69 | 60.15 | 61.87 | 62.94 |
| *Punjabi* | 53.67 | 87.85 | 66.63 | 53.25 | 54.19 | 55.28 | 54.73 | 51.41 | 68.05 | 32.34 | 43.84 | 55.98 |
| *Tamil* | 57.00 | 41.68 | 48.15 | 55.12 | 5.06 | 1 | 1.67 | 41.12 | 29.71 | 14.20 | 19.22 | 40.30 |

Table 3: Accuracy scores (in percentage) over different languages using the three *Zero-Shot* models. We report performance metrics as Precision (P), Recall (R), Accuracy (A), and F1. ***Takeaway***: *The zero-shot models are performing really poor.*

Code Snippet for API

```
import requests

headers = {
   "X-API-Key": "your_api_key_here",
   "Content-Type": "application/json"
}

data = {
    "text": "tu bohot ganda hai",
    "lang": "hi"
}

response = requests.post(
    "https://lingo.iitgn.ac.in/
    unityai-guard/api",
    headers=headers,
    json=data
)

print(response.json())

# Output: {
  "confidence": 78.24, "is_toxic":
  true, "toxicity": 21.76
}
```

## 5 Conclusion and Future Directions

We presented UNITYAI-GUARD, a comprehensive framework for toxicity detection across six low-resource Indian languages with diverse scripts. Our contributions include the largest annotated toxicity dataset for Hindi, Telugu, Marathi, Urdu, Punjabi, and Tamil, validated by two native speakers, and an accessible web portal with transliteration, speech recognition, and API capabilities. Experimental results demonstrate the efficacy of our approach across different model capacities. Future work will focus on expanding language coverage to include additional Indian languages, mainly from Dravidian and North-Eastern families, incorporating contextual understanding for culturally-specific nuances and developing robust cross-lingual transfer approaches to accommodate India's script diversity. We intend to expand this work by releasing the complete dataset with fine-grained annotations across 14 distinct toxicity categories in the near future.

## Limitations

Due to the time and cost-intensive nature of the manual inspection, our study was constrained to six languages (and 30k test instances), as seeking qualified native-speaker annotators for a broader

language set presented significant logistical challenges (Looking for Punjabi, Tamil, and Telugu speakers was extremely challenging). We implemented language code as a critical parameter in our methodology to address a fundamental limitation in toxicity detection across Indian languages: lexical items that are neutral in one linguistic context may carry offensive connotations in another. For example, particular North Indian family surnames may be perceived as toxic content by South Indian users and vice versa. Consequently, we employed language-specific parameterization to facilitate targeted model selection for individual languages rather than implementing a single unified classification model that could not account for these cross-linguistic semantic variations.

## Ethics

Our toxicity detection research prioritizes ethical considerations throughout dataset creation and model development. Annotators were informed about potentially harmful content and provided with mental health resources. We anonymized all personally identifiable information and developed culturally-sensitive annotation guidelines with native speakers to respect linguistic nuances. While our system aims to detect online harm, we acknowledge the risk of false positives potentially silencing legitimate speech, and emphasize that automated tools should supplement rather than replace human moderation in sensitive contexts.

## Acknowledgments

This work is supported by the Prime Minister Research Fellowship (PMRF-1702154) to Himanshu Beniwal.

## References

Sumaiya Rahman Aodhora, Shawly Ahsan, and Mohammed Moshiul Hoque. 2025. CUET_HateShield@NLU of Devanagari script languages 2025: Transformer-based hate speech detection in Devanagari script languages. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 260–266, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A dataset of Hindi-English code-mixed social media text for hate speech detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 36–41, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Marta Costa-jussà, Eric Smith, Christophe Ropers, Daniel Licht, Jean Maillard, Javier Ferrando, and Carlos Escolano. 2023. Toxicity in multilingual machine translation at scale. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9570–9586, Singapore. Association for Computational Linguistics.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, Acyr Locatelli, Hangyu Lin, Dwarak Talupuru, Bharat Venkitesh, David Cairuz, Bowen Yang, Tim Chung, Wei-Yin Ko, Sylvie Shang Shi, Amir Shukayev, Sammie Bae, Aleksandra Piktus, Roman Castagné, Felipe Cruz-Salinas, Eddie Kim, Lucas Crawhall-Stein, Adrien Morisot, Sudip Roy, Phil Blunsom, Ivan Zhang, Aidan Gomez, Nick Frosst, Marzieh Fadaee, Beyza Ermis, Ahmet Üstün, and Sara Hooker. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *Preprint*, arXiv:2412.04261.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Pratibha Dongare. 2024. Creating corpus of low resource indian languages for natural language processing: Challenges and opportunities. In *Proceedings of the 7th Workshop on Indian Language Data: Resources and Evaluation*, pages 54–58.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Md Shariq Farhan. 2025. Hyderabadi pearls at multilingual counterspeech generation : HALT : Hate speech alleviation using large language models and transformers. In *Proceedings of the First Workshop on Multilingual Counterspeech Generation*, pages 65–76, Abu Dhabi, UAE. Association for Computational Linguistics.

Jay Gala, Pranjal A Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, et al. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *arXiv preprint arXiv:2305.16307*.

Avishek Garain, Atanu Mandal, and Sudip Kumar Naskar. 2021. JUNLP@DravidianLangTech-EACL2021: Offensive language identification in Dravidian langauges. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 319–322, Kyiv. Association for Computational Linguistics.

Asha Hegde, Kavya G, Sharal Coelho, and Hosahalli Lakshmaiah Shashirekha. 2023. MUCS@DravidianLangTech2023: Leveraging learning models to identify abusive comments in code-mixed Dravidian languages. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 266–274, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Devansh Jain, Priyanshu Kumar, Samuel Gehman, Xuhui Zhou, Thomas Hartvigsen, and Maarten Sap. 2024. Polyglotoxicityprompts: Multilingual evaluation of neural toxic degeneration in large language models. *Preprint*, arXiv:2405.09373.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. IndicNLPSuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4948–4961, Online. Association for Computational Linguistics.

Kushal Kedia and Abhilash Nandy. 2021. indic-nlp@kgp at DravidianLangTech-EACL2021: Offensive language identification in Dravidian languages. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 330–335, Kyiv. Association for Computational Linguistics.

Mohammed Safi Ur Rahman Khan, Priyam Mehta, Ananth Sankar, Umashankar Kumaravelan, Sumanth Doddapaneni, Suriyaprasaad B, Varun G, Sparsh Jain, Anoop Kunchukuttan, Pratyush Kumar, Raj Dabre, and Mitesh M. Khapra. 2024. IndicLLMSuite: A blueprint for creating pre-training and fine-tuning datasets for Indian languages. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15831–15879, Bangkok, Thailand. Association for Computational Linguistics.

Hariharan LekshmiAmmal, Manikandan Ravikiran, and Anand Kumar Madasamy. 2022. NITK-IT_NLP@TamilNLP-ACL2022: Transformer based model for toxic span identification in Tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 75–78, Dublin, Ireland. Association for Computational Linguistics.

Krishanu Maity, A.S. Poornash, Sriparna Saha, and Pushpak Bhattacharyya. 2024. ToxVidLM: A multimodal framework for toxicity detection in code-mixed videos. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11130–11142, Bangkok, Thailand. Association for Computational Linguistics.

Nikhil Narayan, Mrutyunjay Biswal, Pramod Goyal, and Abhranta Panigrahi. 2023. Hate speech and offensive content detection in indo-aryan languages: A battle of lstm and transformers. *arXiv preprint arXiv:2312.05671*.

Partha Pakray, Alexander Gelbukh, and Sivaji Bandyopadhyay. 2025. Natural language processing applications for low-resource languages. *Natural Language Processing*, 31(2):183–197.

Md. Rahman, Abu Raihan, Tanzim Rahman, Shawly Ahsan, Jawad Hossain, Avishek Das, and Mohammed Moshiul Hoque. 2024. Binary_Beasts@DravidianLangTech-EACL 2024: Multimodal abusive language detection in Tamil based on integrated approach of machine learning and deep learning techniques. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 212–217, St. Julian's, Malta. Association for Computational Linguistics.

Md Nishat Raihan, Umma Tanmoy, Anika Binte Islam, Kai North, Tharindu Ranasinghe, Antonios Anastasopoulos, and Marcos Zampieri. 2023. Offensive language identification in transliterated and code-mixed Bangla. In *Proceedings of the First Workshop on Bangla Language Processing (BLP-2023)*, pages 1–6, Singapore. Association for Computational Linguistics.

Hafiz Hassaan Saeed, Muhammad Haseeb Ashraf, Faisal Kamiran, Asim Karim, and Toon Calders. 2021. Roman urdu toxic comment classification. *Language Resources and Evaluation*, pages 1–26.

Tanik Saikh, Soham Barman, Harsh Kumar, Saswat Sahu, and Souvick Palit. 2024. Emojis trash or treasure: Utilizing emoji to aid hate speech detection. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 554–561, AU-KBC Research Centre, Chennai, India. NLP Association of India (NLPAI).

Amitoj Singh, Virender Kadyan, Munish Kumar, and Nancy Bassan. 2020. Asroil: a comprehensive survey for automatic speech recognition of indian languages. *Artificial Intelligence Review*, 53(5):3673–3704.

Varad Srivastava. 2025. DweshVaani: An LLM for detecting religious hate speech in code-mixed Hindi-English. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 46–60, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Minjia Wang, Pingping Lin, Siqi Cai, Shengnan An, Shengjie Ma, Zeqi Lin, Congrui Huang, and Bixiong

Xu. 2025. STAND-guard: A small task-adaptive content moderation model. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 1–20, Abu Dhabi, UAE. Association for Computational Linguistics.

Meng Ye, Karan Sikka, Katherine Atwell, Sabit Hassan, Ajay Divakaran, and Malihe Alikhani. 2023. Multilingual content moderation: A case study on Reddit. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3828–3844, Dubrovnik, Croatia. Association for Computational Linguistics.

# A Appendix

## A.1 Dataset

Our dataset construction process involved content extraction from online sources. We employed web scraping techniques and implemented classification using a manually curated lexicon of toxic terms. Text instances containing items from this predefined vocabulary received "toxic" labels, while those without such terms were classified as "non-toxic". To simplify the annotation framework, we converted the initial multi-class labeling scheme to binary: content originally marked non-toxic retained this designation, while all gradations of toxicity ("moderately toxic", "highly toxic") were consolidated into a single "toxic" category.

We present the language-wise source and license list as:

**Hindi** Hindi hate speech data[9], MACD[10], textdetox/multilingual_toxicity_dataset[11], QCRI/LlamaLens-Hindi[12](Apache 2.0, CC BY-NC-SA 4.0, Creative Commons Attribution 4.0, Open Rail++-M, and MIT.).

**Telugu** HOLD-Telugu[13], Detecting Insults in Social Commentary[14], and mounikaiiith/Telugu-Hatespeech[15] (CC-BY-4.0 and CC BY-ND 3.0).

**Marathi** MahaHate[16], MOLD[17] (CC-BY-4.0).

**Urdu** Roman-Urdu-Toxic-Corpus[18], Parallel-Urdu-Roman-Urdu-Corpus[19], and

[9] https://www.kaggle.com/datasets/yash3056/hindi-hate-speech-data

[10] https://github.com/ShareChatAI/MACD

[11] https://huggingface.co/datasets/textdetox/multilingual_toxicity_dataset/viewer/default/hi

[12] https://huggingface.co/datasets/QCRI/LlamaLens-Hindi/viewer/Offensive_Speech_Detection

[13] https://github.com/Mussabat/HateSpeech-EACL-2024/

[14] https://www.kaggle.com/c/detecting-insults-in-social-commentary/data?select=test.csv

[15] https://www.kaggle.com/c/detecting-insults-in-social-commentary/

[16] https://github.com/l3cube-pune/MarathiNLP/tree/main/L3Cube-MahaHate

[17] https://github.com/TharinduDR/MOLD/tree/master/MOLD_1.0

[18] https://huggingface.co/datasets/hafiz-hassaan-saeed/Roman-Urdu-Toxic-Corpu

[19] https://huggingface.co/datasets/hafiz-hassaan-saeed/Parallel-Urdu-Roman-Urdu-Corpus

| Language | Agreement | Cohen's Kappa |
|---|---|---|
| *Hindi* | 99.60 | 99.2 |
| *Telugu* | 99.08 | 98.16 |
| *Marathi* | 98 | 96.01 |
| *Urdu* | 98.10 | 92.61 |
| *Punjabi* | 89.02 | 77.95 |
| *Tamil* | 88.97 | 83.10 |

Table 4: Percentage agreement and Inter Annotator Agreement scores for the six languages. We report both the scores in percentages. ***Takeaway***: *There is a high agreement that the content is actually toxic.*

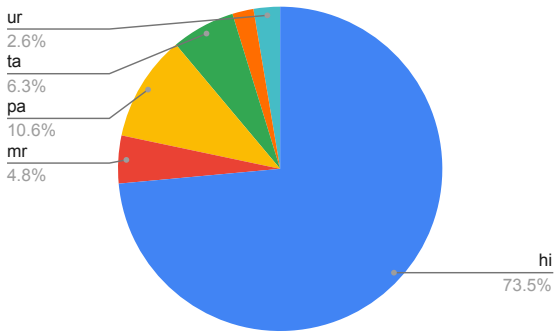


Figure 5: The proportion of languages inferred in the UI interface. ***Takeaway***: *Hindi has been inferred the maximum.*

roman_urdu_hate_speech[20] (CC-BY-4.0 and MIT).

**Punjabi** News18 Punjab, BBC News Punjabi, PTC News (CC-BY-4.0).

**Tamil** CulturaX[21], Thanthi[22], and Siruvar Malar[23] (MC4 License, OSCAR license, and CC BY-SA 4.0). More details about the sources, labeling strategy, and licenses are available on our project page[24].

We also computed the test-set's percentage agreement and inter-annotator agreement (IAA) as shown in Table 4. We compute Cohen's Kappa using the inbuilt function and percentage agreement by the sum of exact matches divided by a total number of instances.

[20] https://github.com/haroonshakeel/roman_urdu_hate_speech

[21] https://huggingface.co/datasets/uonlp/CulturaX/tree/main/ta

[22] https://www.dailythanthi.com/news/tamilnadu

[23] https://www.tamilsiruvarkathaigal.com/

[24] https://github.com/himanshubeniwal/IndicToxic

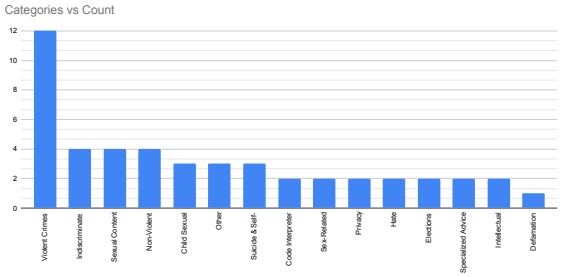


Figure 6: Categories reported by users during the feedback system. ***Takeaway***: *Majority of the violent crimes were reported by the users.*

## B Inception Since Launch

Since the framework's launch in last week of March 2025, the UI has recorded over 200 requests. Figure 5 presents the language distribution, showing the highest inference frequency for Hindi ("*hi*"), while low-resource languages such as Tamil ("*ta*") and Urdu ("*ur*") exhibited minimal usage. Toxicity analysis revealed that 33% of responses were flagged as toxic, with 66.7% classified as non-toxic. Among predicted labels, 58.1% of users reported relevant generation quality. Notably, toxicity definitions may vary across users. Figure 6 displays user-reported feedback categories.