CUET-NLP_Zenith at BLP-2025 Task 1: A Multi-Task Ensemble Approach for Detecting Hate Speech in Bengali YouTube Comments

Md. Refaj Hossan, Kawsar Ahmed, and Mohammed Moshiul Hoque
Department of Computer Science and Engineering
Chittagong University of Engineering & Technology, Chittagong 4349, Bangladesh
{u1904007, u1804017}@student.cuet.ac.bd
moshiul_240@cuet.ac.bd

Abstract

Hate speech on social media platforms, particularly in low-resource languages like Bengali, poses a significant challenge due to its nuanced nature and the need to understand its type, severity, and targeted group. To address this, the Bangla Multi-task Hate Speech Identification Shared Task at BLP 2025 adopts a multi-task learning framework that requires systems to classify Bangla YouTube comments across three subtasks simultaneously: type of hate, severity, and targeted group. To tackle these challenges, this work presents **BanTriX**, a transformer ensemble method that leverages BanglaBERT-I, XLM-R, and BanglaBERT-II. Evaluation results show that the **BanTriX**, optimized with cross-entropy loss, achieves the highest weighted micro F1-score of 73.78% in Subtask 1C, securing our team 2nd place in the shared task.

1 Introduction

Hate speech identification relies on detecting and classifying harmful or offensive language in text, with careful analysis of its type (such as personal attack or communal hate), severity (ranging from mild to severe), and targeted group (including gender and religion); these factors play a critical role in fostering safe online environments (Fayaz et al., 2025). Particularly in low-resource languages (LRLs) like Bengali, the limited availability of annotated datasets and the inherent linguistic complexity present significant challenges. The necessity for multi-task learning, where models must classify multiple related objectives at once and reflect the interconnectedness of real-world scenarios, further complicates this task. The scarcity of comprehensive datasets has hindered progress, the contextual subtlety of Bengali hate speech, and a lack of previous multi-task learning frameworks for LRLs like Bengali. In response, the BLP Workshop@IJCNLP-AACL 2025 organized a shared task (Hasan et al., 2025b) centering on multi-task hate speech identification in Bengali YouTube comments, with classification by type (abusive, religious, or political), severity (mild or severe), and targeted group (society or organization). This collaborative effort highlights the importance of synergy in advancing robust and interpretable hate speech detection systems. Such collaboration forms the central motivation for our work. Our main contributions are summarized as follows:

- We developed **BanTriX**, a robust ensemble that merges BanglaBERT-I, XLM-R, and BanglaBERT-II for multi-task hate speech classification in Bengali.
- By evaluating diverse deep learning, transformer models and their ensembles with comprehensive metrics and ablation studies, we identify the optimal multi-task strategy.
- To enhance interpretability, we employ LIME to highlight feature importance and illuminate the decision processes of our proposed architecture.

2 Related Work

In recent years, researchers have explored harmful online behaviors, e.g., cyberbullying and abusive language, often treating them as related to hate speech. Within this space, automated hate speech detection has progressed rapidly, initially focusing on English datasets (Davidson et al., 2017; Founta et al., 2018), and later expanding to languages such as Arabic (Omar et al., 2020), Spanish (del Arco et al., 2021), and Bengali (Das et al., 2022). This shift was facilitated by shared tasks such as HASOC (Mandl et al., 2025), CHiPSAL (Sarveswaran et al., 2025), and DravidianLangTech@NAACL 2025 (G et al.,

Proceedings of the Second Workshop on Bangla Language Processing (BLP-2025), pages 443–452

2025). Several studies have provided a comprehensive overview of hate speech detection techniques, highlighting key contexts (Maruf et al., 2024; Nandi et al., 2024).

A study by Acharya et al. (2025) evaluated FastText and BERT for hate speech detection and target identification, finding that FastText with data augmentation performed best for hate speech (F1 score of 0.8552). At the same time, BERT excelled in target identification (F1 score of 0.5785). Farsi et al. (2024) explored LR, SVM, CNN, XLM-R, and MuRIL, achieving the best result with Indic-SBERT (macro F1 of 0.7013). A FastText model for Hindi offensive text classification achieved 92.2% accuracy on the DHOT dataset (Jha et al., 2020). However, several works addressed aggressive content in Bengali. For instance, Remon et al. (2022) introduced a 10,133-comment Facebook dataset, where SVM with FastText embeddings performed best. Fayaz et al. (2025) proposed BIDWESH, covering regional dialects with 9k+ samples for fair detection. Sharif et al. (2022) presented M-BAD with 15,650 texts for aggression and target detection, achieving a weighted F1 of 0.92 and 0.83 using BanglaBERT. A 30k-comment Bengali dataset from YouTube and Facebook annotated in 7 categories, with SVM reaching 87.5% accuracy (Romim et al., 2020).

Despite significant progress in hate speech and offensive content detection, including multimodal approaches (Hossain et al., 2022; Hee et al., 2023), to the best of our knowledge, no prior studies have addressed a multi-task setup that simultaneously predicts hate type, severity, and targeted group in Bengali. Building on this gap, this study presents a multi-task learning scenario using Bengali text from YouTube comments, aiming to develop robust systems for comprehensive analysis of hate speech.

3 Task and Dataset Description

This study develops a system to classify Bengali YouTube comments by hate type (such as *Abusive*, *Political Hate*, or *Profane*), severity (like *Mild* or *Severe*), and the group targeted (for example, *Individual*, *Organization*, or *Society*). This multi-task approach helps capture how different aspects of hate speech are connected in Bengali. The dataset (Hasan et al., 2025a) contains annotated Bengali comments divided into training, validation, and test sets. For example, the training set has 8,212 *Abusive* and 4,227 *Political Hate* cases, 23,489 with *little* to *no severity*, and 5,646 targeting *individuals*. These figures show the dataset's variety, which supports strong model development. However, there is an imbalance in the dataset, with more samples labeled as *None hate* type (19,954 in training, 1,451 in validation, and 5,751 in test) compared to other categories, as shown in Table 1. Appendix A provides further exploratory data analysis.

Subtask	Classes	Train	Valid	Test	W_T
Hate Type	Abusive	8212	564	2312	153869
	Political Hate	4227	291	1220	109447
	Profane	2331	157	709	43618
	Religious Hate	676	38	179	14659
	Sexism	122	11	29	2396
	None	19954	1451	5751	341607
	Total	**35522**	**2512**	**10200**	**665596**
Hate Severity	Little to None	23489	1703	6737	414639
	Mild	6853	483	2001	146920
	Severe	5180	326	1462	104037
	Total	**35522**	**2512**	**10200**	**665596**
Targeted Group	Individual	5646	364	1571	102771
	Organization	3846	292	1152	84773
	Community	2635	179	759	59800
	Society	2205	141	625	52132
	None	21190	1536	6093	366120
	Total	**35522**	**2512**	**10200**	**665596**

Table 1: Class-wise distribution of datasets used for the task, where W_T denotes total words.

4 System Overview

This section presents the implementation details of the proposed architecture, encompassing both deep learning and transformer-based models.

4.1 Problem Formulation

Given a set of Bangla YouTube comments $C = \{C_1, \ldots, C_{|C|}\}$, each comment C_i is represented by a text sequence X_i. The goal is to learn a mapping f that assigns three labels $Y_i = \{Y_i^{ht}, Y_i^{hs}, Y_i^{tw}\}$ corresponding to hate type (6 classes), hate severity (3 classes), and targeted group (5 classes), formulating a multi-task classification problem: $f : X_i \rightarrow Y_i$. The models performance is governed by a summed cross-entropy loss function $\mathcal{L} = \mathcal{L}_{ht} + \mathcal{L}_{hs} + \mathcal{L}_{tw}$, which quantifies the divergence between predicted and true labels across tasks. To achieve this, the model solves the optimization problem as shown in Eq. 1.

$$\min_{f} \sum_{i=1}^{|C|} \big[\mathcal{L}_{ht}(f_{ht}(X_i), Y_i^{ht}) + \mathcal{L}_{hs}(f_{hs}(X_i), Y_i^{hs}) + \mathcal{L}_{tw}(f_{tw}(X_i), Y_i^{tw})\big], \quad (1)$$

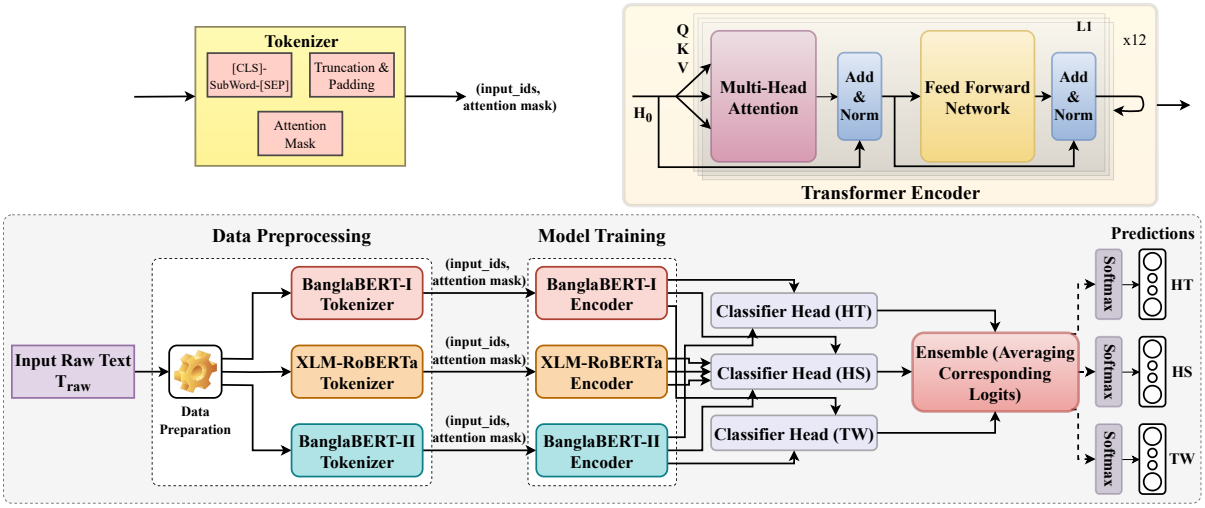


Figure 1: Overview of the proposed architecture for multi-task classification.

where f_{ht}, f_{hs}, f_{tw} are task-specific heads on shared transformer features, and $\mathcal{L}_{ht}, \mathcal{L}_{hs}, \mathcal{L}_{tw}$ are cross-entropy losses for each task.

4.2 Baselines

Several deep learning and transformer-based models were explored to develop the proposed system.

4.2.1 Deep Learning Models

Deep learning models such as CNN (Kim, 2014), BiLSTM (Huang et al., 2015), BiLSTM+CNN, shared a common 128-dimensional word embedding layer trained from scratch on the preprocessed corpus using a `Tokenizer` with `<OOV>` handling. Input sequences were standardized to 128 tokens through post-padding and truncation. The BiLSTM model stacked two bidirectional LSTM layers (128 units each) with a 10% dropout and three parallel output heads. The CNN model applied two Conv1D layers (128 filters, kernel size 5) with max pooling and global max pooling, followed by dropout and identical output heads. The hybrid model combined a 64-unit BiLSTM branch and a 64-filter CNN branch, merged their outputs, and connected them to the output heads. All models were optimized with Adam using task-specific learning rates (2.15×10^{-4}, 1.25×10^{-3}, and 1.5×10^{-4}, respectively), trained with `SparseCategoricalCrossentropy` loss and early stopping (patience of 13) over up to 25–31 epochs with a batch size of 32.

4.2.2 Transformer-Based Models

Transformer-based models such as BanglaBERT-I (SagorSarker[1]), XLM-R (Conneau et al., 2020), and BanglaBERT-II (Bhattacharjee et al., 2022) were fine-tuned for the task using AdamW with learning rates of 2.25e-5, 2e-5, and 2.35e-5, respectively, over 9 epochs and a batch size of 32. All models processed inputs with a maximum sequence length of 128, applied 10% dropout after the [CLS] pooled output, and shared three linear classification heads for hate type, severity, and target prediction. Training used summed CrossEntropy loss across tasks, with linear learning rate warmup (50 steps for BanglaBERT-I, 175 for BanglaBERT-II, 225 for XLM-R) followed by decay over total training steps. After evaluating transformer-based models, BanglaBERT-I, XLM-RoBERTa, and BanglaBERT-II emerged as the best performers, significantly outperforming mBERT. These three models were then ensembled through systematic exploration to identify the optimal strategy (detailed in Appendix C), using their complementary strengths in contextual understanding to develop **BanTriX**.

4.3 Proposed Approach

Figure 1 illustrates the proposed architecture **(BanTriX)** for Bengali hate speech detection in YouTube comments, integrating a shared transformer backbone with task-specific heads and ensembles of BanglaBERT-I, XLM-R, and BanglaBERT-II. Full implementation details can

[1] https://huggingface.co/sagorsarker/bangla-bert-base

be found in the GitHub repository[2].

4.3.1 Data Preparation and Input Layer

The data preparation began with processing raw Bengali text sequences from YouTube comments. Using pre-trained tokenizers (e.g., for BanglaBERT or XLM-R), each text t was converted into input IDs I and attention masks A as depicted in Eq. 2.

$$\{\mathrm{I}, \mathrm{A}\} = \text{tokenizer}(t), \tag{2}$$

where $\mathrm{I}, \mathrm{A} \in \mathbb{R}^{B \times L}$, with batch size $B = 32$ and maximum sequence length $L = 128$.

4.3.2 Shared Encoder

The core of the architecture is a pre-trained transformer encoder (e.g., BanglaBERT-I, XLM-R, or BanglaBERT-II), which extracts contextual embeddings shared across all tasks as shown in Eq. 3.

$$\mathrm{H} = f_{\text{base}}(\mathrm{I}, \mathrm{A}), \tag{3}$$

where $\mathrm{H} \in \mathbb{R}^{B \times L \times D}$ and $D = 768$ is the hidden dimension. This shared encoder captures the linguistic nuances critical for the tasks.

4.3.3 Pooling and Dropout

To create a compact representation, the sequence embeddings were pooled, typically using the `[CLS]` token, yielding $\mathrm{P} \in \mathbb{R}^{B \times D}$. Dropout (0.1) was applied for regularization, as shown in Eq. (4).

$$\mathrm{D} = \text{Dropout}(\mathrm{P}), \tag{4}$$

4.3.4 Task-Specific Heads

From the pooled features D, three linear classifiers were used to produce logits for each task, i.e., hate type ($C_{ht} = 6$), hate severity ($C_{hs} = 3$), and targeted group ($C_{tw} = 5$), as shown in Eq. 5.

$$\begin{aligned} &\mathrm{L}_{ht} = \mathrm{W}_{ht}\mathrm{D} + \mathrm{b}_{ht}, \quad \mathrm{L}_{hs} = \mathrm{W}_{hs}\mathrm{D} + \mathrm{b}_{hs}, \\ &\mathrm{L}_{tw} = \mathrm{W}_{tw}\mathrm{D} + \mathrm{b}_{tw}, \end{aligned} \tag{5}$$

where $\mathrm{L}_{ht} \in \mathbb{R}^{B \times 6}$, $\mathrm{L}_{hs} \in \mathbb{R}^{B \times 3}$, $\mathrm{L}_{tw} \in \mathbb{R}^{B \times 5}$, and W, b are learnable parameters. During inference, probabilities were computed via softmax as $\mathrm{Pr}_{ht} = \text{softmax}(\mathrm{L}_{ht})$, and selected predictions as $\hat{y}_{ht} = \arg\max(\mathrm{Pr}_{ht})$.

[2] https://github.com/RJ-Hossan/BLP_T1_2025

4.3.5 Ensemble Mechanism

To enhance performance, three models were ensemble by averaging their logits as shown in Eq. 6.

$$\bar{\mathrm{L}}_{ht} = \frac{1}{3} \sum_{m=1}^{3} \mathrm{L}_{ht}^{(m)}, \tag{6}$$

and similarly for $\bar{\mathrm{L}}_{hs}$ and $\bar{\mathrm{L}}_{tw}$. Final predictions are obtained via softmax on $\bar{\mathrm{L}}$.

4.3.6 System Requirements

The proposed architecture was trained on Kaggle's free-tier environment using two NVIDIA T4 GPUs in a distributed setup, requiring approximately 5 GB of system RAM and ∼15 GB of GPU memory, with a total training time of around 65 minutes. Table B.1 in Appendix B provides the tuned hyperparameters used in the proposed architecture for the tasks.

5 Results and Discussion

Table 2 summarizes the performance of various approaches for the task, with performance metrics including the overall Weighted Micro F1-Score (μ-F1), True Positive Rate (TPR), and Balanced Error Rate (BER). The following insights are drawn from these results.

Which tasks are hard to solve? The three classification tasks present challenges, particularly due to class imbalances. The CNN+BiGRU model shows a high BER for Hate Type (56.06%), indicating difficulty in correctly classifying all classes. Even transformer-based models like BanglaBERT-II exhibit elevated BER for Hate Type (47.41%). The proposed ensemble achieves lower BERs (e.g., 41.07% for Hate Severity). Still, these values remain relatively high, highlighting that Hate Type and Targeted Group are tough to classify accurately due to their complex class distributions (clarified by error analysis in Appendix E).

Does the ensemble approach improve the result? The result clarifies that ensemble approaches significantly improve performance. The XLM-R+BanglaBERT-II ensemble achieves an overall μ-F1 of 72.52%, surpassing the single BanglaBERT-II model's μ-F1 of 70.49% by 2.88% and improving Hate Type μ-F1 by 2.33%. The proposed ensemble further enhances performance, achieving an overall μ-F1 score of 73.78% (+1.74% compared to XLM-R+BanglaBERT-II). In task-wise, **BanTriX** excels with Hate Type μ-F1 of 73.38% (+4.87% than XLM-R), Hate Sever-

Approaches	Overall				Hate Type		Hate Severity		Targeted Group	
	Pr(%)	Re(%)	μ-F1(%)	TPR(%)	μ-F1(%)	BER(%)	μ-F1(%)	BER(%)	μ-F1(%)	BER(%)
CNN	64.63	67.77	67.77	41.13	65.39	66.70	71.71	44.68	66.21	65.23
BiLSTM	62.45	66.74	66.74	42.31	64.75	62.82	71.06	45.31	64.40	64.93
BiLSTM+CNN	62.62	64.77	64.77	38.29	61.47	69.17	69.45	46.27	63.39	69.69
BiLSTM+BiGRU	63.95	68.05	68.05	42.93	66.12	62.29	72.00	46.18	66.04	62.75
CNN+BiGRU	67.06	69.50	69.50	47.31	68.30	56.06	72.31	44.30	67.87	57.71
BanglaBERT-I	67.21	67.24	67.24	51.75	65.96	50.50	70.12	43.54	65.63	50.71
BanglaBERT-II	70.70	70.49	70.49	56.52	69.66	47.41	72.54	38.82	69.28	44.22
XLM-R	69.27	70.45	70.45	54.76	69.97	47.65	72.11	40.84	69.27	47.23
mBERT	59.69	66.53	66.53	39.60	63.57	66.02	70.63	50.14	65.40	65.04
(BiLSTM+CNN)+BiLSTM	62.89	69.03	69.03	40.64	67.52	64.11	72.59	48.32	66.97	65.66
BiGRU+(BiLSTM+CNN)	64.79	69.34	69.34	43.33	68.15	62.07	72.59	45.37	67.27	62.57
BanglaBERT-I+XLM-R	69.37	71.49	71.49	51.85	70.84	51.06	73.33	42.14	70.29	51.24
XLM-R+BanglaBERT-II	71.77	72.52	72.52	56.40	71.28	47.35	74.12	39.98	72.17	43.46
Proposed (BanTriX)	71.91	73.78	**73.78**	54.97	**73.38**	47.70	**74.95**	41.07	**73.02**	46.32

Table 2: Performance comparisons on test data across different approaches, where Pr, Re, μ-F1, TPR, and BER denote Precision, Recall, Weighted Micro-F1 score, True Positive Rate, and Balanced Error Rate, respectively.

ity μ-F1 of 74.95%, and Targeted Group μ-F1 of 73.02% (+5.41% than XLM-R), demonstrating that combining Bengali-specific transformers enhances robustness and accuracy across all tasks.

Does CCC loss configuration help? Ablation study in Appendix C shows that the CCC setup (C=Cross-Entropy Loss across three models) achieves the best overall performance, surpassing other loss variants. Poor-performing variants, such as FCW and WFL, indicate heavy overfitting. In a task-wise comparison, CCC provides a more balanced performance across all tasks, demonstrating the effectiveness of the proposed CCC loss function combination in stabilizing training and improving generalization.

6 Conclusion

The study introduced the **BanTriX** architecture tailored for the Bengali multi-task hate speech identification, achieving an overall weighted micro F1-Score of 73.78%. An ablation study highlighted that the optimal configuration, using a token length of 128 with the cross-entropy loss combination, excelled in terms of LIME-based interpretability, confirming its focus on hate-indicative features. Future work will explore the integration of LLMs and advanced techniques, such as dynamic loss optimization, to further enhance rare class detection across diverse datasets.

Limitations

While the study yields strong results for detecting Bengali hate speech, it also presents some apparent limitations. Using a fixed token length of 128 means longer posts may lose important context. The CCC loss setup works well overall; however, it struggles with rare hate categories. It also tends to perform better on predicting *None* cases, which risks missing more subtle hate expressions. Moreover, the study acknowledges class imbalance but does not address it, potentially affecting underrepresented categories. It emphasizes model integration and empirical analysis, relying on pretrained transformers with limited task-specific adaptation or efficiency optimization. Finally, more advanced methods, such as large language models, have yet to be explored.

Ethics Statement

We acknowledge the dataset's annotation biases, though mitigated by clear guidelines and schemas. As it contains only non-identifiable comments, no privacy risks arise. Our model adds no ethical concerns, and fairness was ensured by evaluating across hate types and communities. Overall, the dataset enables hate speech detection to support healthier online discourse, with human oversight mitigating misclassification risks and promoting equitable outcomes.

Acknowledgments

This work is supported by the Directorate of Research and Extension and NLP LAB, Chittagong University of Engineering & Technology (CUET), Chittagong, Bangladesh.

References

Darwin Acharya, Sundeep Dawadi, Shivram Saud, and Sunil Regmi. 2025. Paramananda@NLU of Devanagari script languages 2025: Detection of language, hate speech and targets using FastText and BERT. In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 334–338, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Mithun Das, Somnath Banerjee, Punyajoy Saha, and Animesh Mukherjee. 2022. Hate speech and offensive language detection in Bengali. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 286–296, Online only. Association for Computational Linguistics.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference on Web and Social Media*, 11(1):512–515.

Flor Miriam Plaza del Arco, María Dolores Molina-González, L. Alfonso Ureña-López, and María Teresa Martín-Valdivia. 2021. Comparing pre-trained language models for spanish hate speech detection. *Expert Syst. Appl.*, 166:114120.

Salman Farsi, Asrarul Eusha, Jawad Hossain, Shawly Ahsan, Avishek Das, and Mohammed Moshiul Hoque. 2024. CUET_Binary_Hackers@DravidianLangTech EACL2024: Hate and offensive language detection in Telugu code-mixed text using sentence similarity BERT. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 193–199, St. Julian's, Malta. Association for Computational Linguistics.

Azizul Hakim Fayaz, MD. Shorif Uddin, Rayhan Uddin Bhuiyan, Zakia Sultana, Md. Samiul Islam, Bidyarthi Paul, Tashreef Muhammad, and Shahriar Manzoor. 2025. Bidwesh: A bangla regional based hate speech detection dataset. *Preprint*, arXiv:2507.16183.

Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. *Proceedings of the International AAAI Conference on Web and Social Media*, 12(1).

Jyothish Lal G, Premjith B, Bharathi Raja Chakravarthi, Saranya Rajiakodi, Bharathi B, Rajeswari Natarajan, and Ratnavel Rajalakshmi. 2025. Overview of the shared task on multimodal hate speech detection in Dravidian languages: DravidianLangTech@NAACL 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 114–122, Acoma, The Albuquerque Convention Center, Albuquerque, New Mexico. Association for Computational Linguistics.

Md Arid Hasan, Firoj Alam, Md Fahad Hossain, Usman Naseem, and Syed Ishtiaque Ahmed. 2025a. Llm-based multi-task bangla hate speech detection: Type, severity, and target. *Preprint*, arXiv:2510.01995.

Md Arid Hasan, Firoj Alam, Md Fahad Hossain, Usman Naseem, and Syed Ishtiaque Ahmed. 2025b. Overview of blp 2025 task 1: Bangla hate speech identification. In *Proceedings of the Second International Workshop on Bangla Language Processing (BLP-2025)*, India. Association for Computational Linguistics.

Ming Shan Hee, Wen-Haw Chong, and Roy Ka-Wei Lee. 2023. Decoding the underlying meaning of multimodal hateful memes. *Preprint*, arXiv:2305.17678.

Eftekhar Hossain, Omar Sharif, and Mohammed Moshiul Hoque. 2022. MUTE: A multimodal dataset for detecting hateful memes. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 32–39, Online. Association for Computational Linguistics.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *Preprint*, arXiv:1508.01991.

Vikas Kumar Jha, Hrudya P, Vinu P N, Vishnu Vijayan, and Prabaharan P. 2020. Dhot-repository and classification of offensive tweets in the hindi language. *Procedia Computer Science*, 171:2324–2333. Third International Conference on Computing and Network Communications (CoCoNet'19).

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *Preprint*, arXiv:1408.5882.

Thomas Mandl, Koyel Ghosh, Nishat Raihan, Sandip Modha, Shrey Satapara, Tanishka Gaur, Yaashu Dave, Marcos Zampieri, and Sylvia Jaki. 2025. Overview of the hasoc track 2024: Hate-speech identification in english and bengali. In *Proceedings of the 16th Annual Meeting of the Forum for Information Retrieval Evaluation*, FIRE '24, page 12, New York, NY, USA. Association for Computing Machinery.

Abdullah Maruf, Ahmad Jainul Abidin, Md Haque, Zakaria Masud, Aditi Golder, Raaid Alubady, and Zeyar Aung. 2024. Hate speech detection in the bengali language: a comprehensive survey. *Journal of Big Data*, 11.

Arpan Nandi, Kamal Sarkar, Arjun Mallick, and Arkadeep De. 2024. A survey of hate speech detection in indian languages. *Social Network Analysis and Mining*, 14(1):70.

Ahmed Omar, Tarek M. Mahmoud, and Tarek Abd-El-Hafeez. 2020. Comparative performance of machine learning and deep learning algorithms for arabic hate speech detection in osns. In *Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020)*, pages 247–257, Cham. Springer International Publishing.

Nasif Istiak Remon, Nafisa Hasan Tuli, and Ranit Debnath Akash. 2022. Bengali hate speech detection in public facebook pages. In *2022 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pages 169–173.

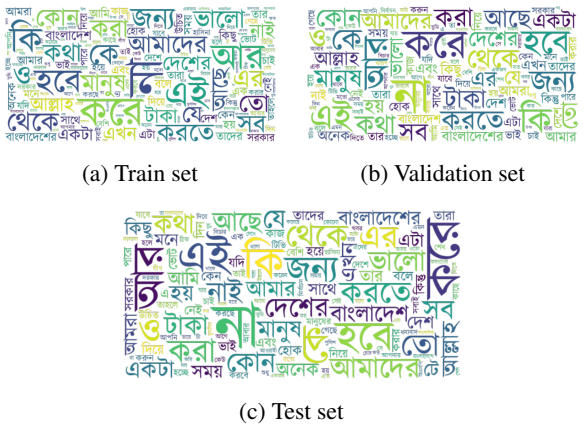
Nauros Romim, Mosahed Ahmed, Hriteshwar Talukder, and Md Saiful Islam. 2020. Hate speech detection in the bengali language: A dataset and its baseline evaluation. *Preprint*, arXiv:2012.09686.

Kengatharaiyer Sarveswaran, Surendrabikram Thapa, Sana Shams, Ashwini Vaidya, and Bal Krishna Bal. 2025. A brief overview of the first workshop on challenges in processing South Asian languages (CHiPSAL). In *Proceedings of the First Workshop on Challenges in Processing South Asian Languages (CHiPSAL 2025)*, pages 1–8, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Omar Sharif, Eftekhar Hossain, and Mohammed Moshiul Hoque. 2022. M-BAD: A multilabel dataset for detecting aggressive texts and their targets. In *Proceedings of the Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situations*, pages 75–85, Dublin, Ireland. Association for Computational Linguistics.

A Exploratory Data Analysis

Figure A.1 shows the word cloud generated from the train, validation, and test datasets, where frequently occurring words appear larger in size. This visualization highlights dominant patterns and recurring terms in the corpus, offering quick insights into the lexical distribution of the dataset and serving as an effective tool for understanding key themes and guiding preprocessing decisions.



(a) Train set (b) Validation set

(c) Test set

Figure A.1: Word clouds (top 200 words) across all three datasets.

Figure A.2 illustrates the feature correlation map, which reveals the three subtasks: Hate Type, Hate Severity, and Targeted Group. They are

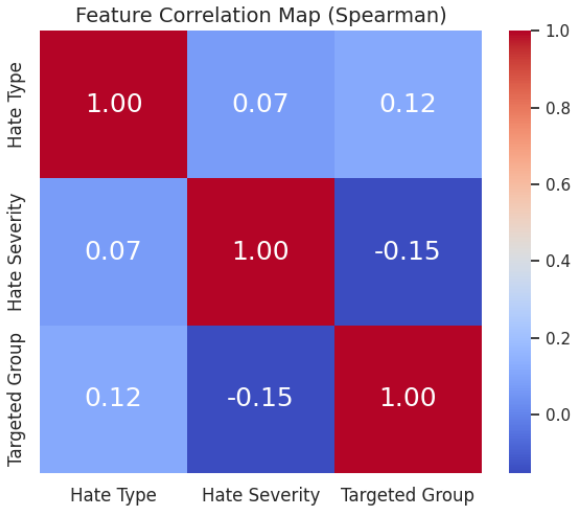


Figure A.2: Feature correlation map across three datasets.

largely independent, with only weak associations. For example, Hate Type vs. Hate Severity shows a very low positive correlation of 0.07, Hate Type vs. Targeted Group is slightly higher at 0.12, while Hate Severity vs. Targeted Group exhibits a weak

negative correlation of -0.15. These low values indicate that each label contributes distinct information, justifying the multi-task setup and highlighting the datasets richness for modeling diverse aspects of hate speech.

B Tuned Hyperparameters

Table B.1 summarizes the key hyperparameters used for training the proposed architecture (**BanTriX**), including a dropout rate of 0.1, a token length of 128, a batch size of 32, model-specific learning rates (2.5e-5 for BanglaBERT-I, 2e-5 for XLM-R and BanglaBERT-II), AdamW optimizer, linear warmup scheduler with 0 warmup steps, and two training epochs.

Attribute	Value
Dropout	0.1
Token Length	128
Batch Size	32
Learning Rate	2.5e-5 (BanglaBERT-I) 2e-5 (XLM-R) 2e-5 (BanglaBERT-II)
Optimizer	AdamW
Scheduler	Linear Schedule with Warmup
Warmup Steps	0
Epochs	2
Loss Function	Cross-Entropy Loss

Table B.1: Hyperparameters used for training of **BanTriX** in multi-task hate speech detection.

C Ablation Study

The ablation study (Table C.1) examines the impact of various loss function combinations and maximum token lengths (T_L) on performance, using an epoch size of 2. The study reveals that the optimal configuration ($T_L = 128$) achieves an overall μ-F1 score of 73.78% and a TPR of 54.97% (slightly lower than the best), thereby balancing context capture and generalization.

C.1 Loss Function Combinations

The proposed architecture uses Cross-Entropy Loss (C) across three transformer models (BanglaBERT-I, XLM-R, BanglaBERT-II). Still, we explored combinations of Cross-Entropy (C), Focal Loss (F), Weighted Cross-Entropy (W), and Label-Smoothed Cross-Entropy (L), as defined in Eq. C.7.

$$\mathcal{L}_{CE} = -\sum_{c=1}^{C} y_c \log(\hat{y}_c) \tag{C.7a}$$

$$\mathcal{L}_{WCE} = -\sum_{c=1}^{C} w_c y_c \log(\hat{y}_c), \tag{C.7b}$$

$$\mathcal{L}_{FL} = -\sum_{c=1}^{C} (1-\hat{y}_c)^{\gamma} y_c \log(\hat{y}_c) \tag{C.7c}$$

$$\mathcal{L}_{LSCE} = -(1-\epsilon)\sum_{c=1}^{C} y_c \log(\hat{y}_c) - \frac{\epsilon}{C}\sum_{c=1}^{C} \log(\hat{y}_c) \tag{C.7d}$$

where reduction="mean", $\gamma = 2.0$, $\alpha =$ None, and $\epsilon = 0.1$ used.

The **CFL** setup (Cross-Entropy, Focal Loss, Label-Smoothed) achieves the best results with 73.03% μ-F1 and 53.64% TPR, surpassing **FFF** (all Focal Loss) at 72.52% μ-F1 and 55.23% TPR by +0.70% μ-F1 and -2.96% TPR. Poor-performing variants **FCW**, **WFL**, and **LWF** yield only 32.05% μ-F1 and 23.33% TPR, reflecting heavy overfitting. In task-wise, CFL records 72.40% μ-F1 for Hate Type (vs. 71.57% of FFF), 74.68% for Severity (vs. 74.46%), and 72.00% for Targeted Group (vs. 71.53%), with slightly higher BER (+5.87%) on Hate Type but overall more balanced performance after the proposed `CCC` loss function combination.

C.2 Maximum Token Length

Varying the maximum token length (T_L) impacts context capture. At T_L of 156, the proposed method achieves the 2nd best (after T_L of 128) overall μ-F1 of 73.72% and TPR of 56.72%, surpassing T_L of 64 (73.51% μ-F1) by 0.29% in μ-F1, and T_L of 256 (72.91% μ-F1, 59.16% TPR) by +1.11% in μ-F1 but -4.30% in TPR. In task-wise, T_L of 156 yields Hate Type μ-F1 of 72.94% (0.27% better than T_L of 64) with BER of 47.58%, Hate Severity μ-F1 of 75.21% with BER of 38.53%, and Targeted Group μ-F1 of 73.01%, indicating optimal performance (except token length of 128) at token length of 156.

C.3 Batch Size Analysis

Figure C.1 shows that a batch size of 32 yields the best results, with μ-F1 scores of 73.38% (Hate

Attributes	Overall				Hate Type		Hate Severity		Targeted Group	
	Pr(%)	Re(%)	μ-F1(%)	TPR(%)	μ-F1(%)	BER(%)	μ-F1(%)	BER(%)	μ-F1(%)	BER(%)
Loss Function Combinations										
CFL	71.01	73.03	73.03	53.64	72.40	49.39	74.68	40.69	72.00	48.99
FCW	16.44	32.05	32.05	23.33	22.67	83.33	66.05	66.67	7.44	80.00
WFL	16.44	32.05	32.05	23.33	22.67	83.33	66.05	66.67	7.44	80.00
FFF	71.35	72.52	72.52	55.23	71.57	46.65	74.46	40.08	71.53	47.58
LWF	16.44	32.05	32.05	23.33	22.67	83.33	66.05	66.67	7.44	80.00
Maximum Token Length, T_L										
T_L = 64	72.40	73.51	73.51	57.32	72.74	46.41	74.87	39.26	72.91	42.37
T_L = 128	71.91	73.78	73.78	54.97	73.38	47.70	74.95	41.07	73.02	46.32
T_L = 156	72.71	73.72	73.72	56.72	72.94	47.58	75.21	38.53	73.01	43.74
T_L = 224	71.89	73.57	73.57	55.17	72.81	47.84	75.20	40.17	72.70	46.48
T_L = 256	73.19	72.91	72.91	59.16	72.08	45.14	74.28	36.80	72.37	40.56

Table C.1: Ablation study of **BanTriX** on test data with loss function combinations and maximum token length. Here, Pr, Re, μ-F1, TPR, and BER denote Precision, Recall, Weighted Micro-F1 score, True Positive Rate, and Balanced Error Rate, respectively.

Type), 74.95% (Severity), and 73.02% (Targeted Group), along with the lowest BERs. Smaller sizes (8, 16) achieve around 72–74.6% μ-F1, while larger sizes (64, 96) drop to about 69–74.1% μ-F1 with higher BERs, confirming batch size 32 as the optimal choice.

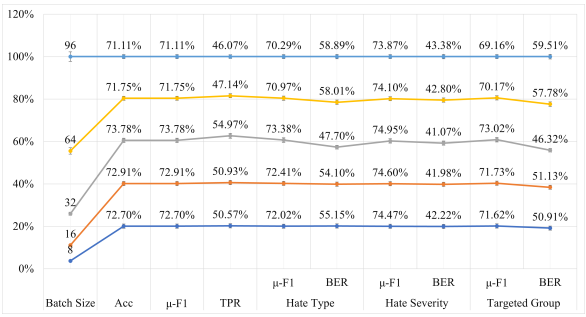


Figure C.1: Overview of the batch size impact while the maximum token length is 128 and cross-entropy loss is used.

D Performance Comparison

Table D.1 compares teams' performance with baselines in the task, with *CUET-NLP_Zenith* (our team) achieving a Weighted Micro F1-Score (μ-F1) of 73.78%, ranking second. It trails *mahim_ju's* performance by 0.16% but beats *shifat_islam's* performance by 0.23%. Compared to the baselines provided by the organizers, the proposed **BanTriX** outperforms the Majority Baseline by 21.51% and the n-gram Baseline by 17.02%, demonstrating its superiority over traditional approaches.

Baseline/Team	**μ-F1 (%)**	**Rank**
mahim_ju	73.92	1
CUET-NLP_Zenith	73.78	2
shifat_islam	73.61	3
reyazul	73.32	4
Random Baseline	23.04	-
Majority Baseline	60.72	-
n-gram Baseline	63.05	-

Table D.1: Performance comparison of the proposed architecture with other teams' approaches.

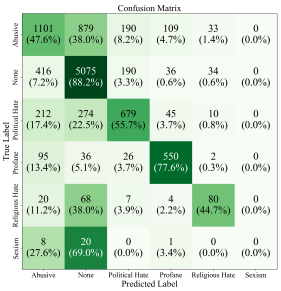
E Error Analysis

A thorough quantitative and qualitative error analysis was conducted to gain an in-depth understanding of the proposed architecture's performance in the task.

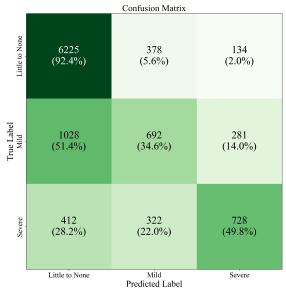
E.1 Quantitative Analysis

The confusion matrices presented in Figure E.1 reveal distinct performance patterns across hate types (Figure E.1a), hate severity (Figure E.1b), and targeted groups (Figure E.1c).

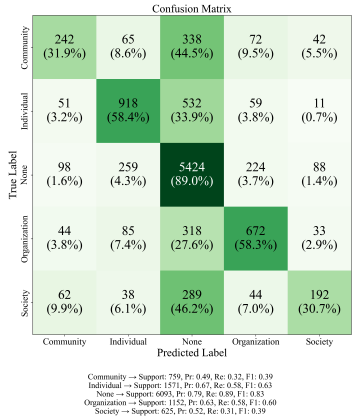
For *hate severity*, the model performs well on *Little to None* (92.4% accuracy, F1 of 0.86) but struggles with *Mild* (F1 of 0.41, often misclassified as *Little to None*) and *Severe* (F1 of 0.56, with frequent downgrading). For targeted groups, performance is weak due to dataset skew, with *Society* achieving only 0.39 F1 and few instances for *Community*. Regarding hate types, strong results are observed for *Profane* (F1 of 0.76) and *None* (F1 of 0.84). In contrast, *Abusive* exhibits limited recall (0.48), and *Sexism* performs poorly, often



(a) Hate Type



(b) Hate Severity



(c) Targeted Group

Figure E.1: Confusion matrices for different categories in the task.

being misclassified as *None*. Overall, the model excels at the majority classes but struggles with minority hate categories (e.g., *sexism*), reflecting bias toward predicting *None* in ambiguous cases.

E.2 Qualitative Analysis

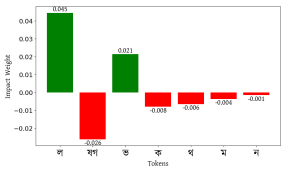
The qualitative analysis of sample classifications shown in Table E.1 illustrates varied model performance in detecting hate speech in Bengali YouTube comments. The model demonstrates strong performance on neutral samples (IDs 266764 and 653626), both labeled as *None*, where the predictions match perfectly. This indicates robustness in handling straightforward non-hate content. In contrast, it struggles with nuanced cases, e.g., sample 241030 (*Political Hate*) was misclassified as *Abusive*, likely due to overlapping sarcastic or abusive tones, while sample 742298 (*Abusive*) was predicted as *None*, reflecting difficulties with cultural subtleties and dataset imbalance. Overall, the model reliably detects clear non-hate instances but faces challenges with context-dependent hate and minority categories.

Sample No.	Text	#	Hate Type	Hate Severity	Targeted Group
241030	ভারতীয় দালাল সময় টিভিকে বয়কট করুন	Actual	*Political Hate*	*Mild*	*Organization*
		Prediction	*Abusive* ✗	✓	✓
266764	অস্ত্র সহ সেনাবাহিনী পাঠানো হোক	Actual	*None*	*Little to None*	*None*
		Prediction	✓	✓	✓
742298	আর্জেন্টিনা ১৯৮৬ আর ১৯৯০ এ কি করছে আবুল সাংঘাতিক	Actual	*Abusive*	*Mild*	*Individual*
		Prediction	*None* ✗	*Little to None* ✗	*None* ✗
653626	ভালো মানুষগুলো ভালো থাকুক	Actual	*None*	*Little to None*	*None*
		Prediction	✓	✓	✓

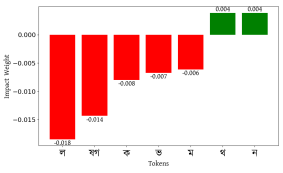
Table E.1: Few sample predictions by **BanTriX** in the task. The $\surd$ mark indicates the correct predictions, and ***X*** denotes incorrect predictions.

F Model Interpretability

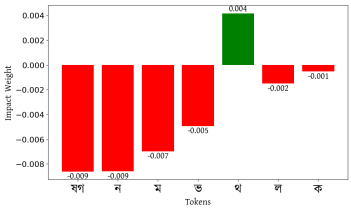
The LIME-based explanation bar plots (see Figure F.1) for the last sample of Table E.1, labeled as *None* across hate type, severity, and targeted group, provide insights into token contributions to the model's predictions. For hate severity, the plot shows that tokens like "ল *(la)*" and "ক *(ka)*" negatively impact (red bars) the prediction of *Little to None* severity, reducing its likelihood, while other tokens positively impact (green bars), supporting the *None* prediction. In the hate type plot, "ন *(na)*" negatively affects the predictions, whereas some tokens like "ল *(la)*", "ভ *(bha)*" positively reinforce the *None* classification, highlighting these tokens as key neutral indicators. Overall, the model relies heavily on neutral tokens to correctly classify this sample as *None* across all categories.



(a) LIME: Hate Type



(b) LIME: Hate Severity



(c) LIME: Targeted Group

Figure F.1: LIME-based model interpretability for the last sample of Table E.1.