Ecstasy at BLP-2025 Task 1: TF-IDF Informed Prompt Engineering with LoRA Fine-tuning for Bangla Hate Speech Detection

Kazi Reyazul Hasan[1], Mubasshira Musarrat[1], Muhammad Abdullah Adnan[1]

[1]Department of Computer Science & Engineering,
Bangladesh University of Engineering & Technology,
Dhaka, Bangladesh

Correspondence: kazireyazulhasan@gmail.com, mubasshira31@gmail.com, abdullah.adnan@gmail.com

Abstract

We present a hybrid approach for Bangla hate speech detection that combines linguistic analysis with neural fine tuning. Our method first identifies category specific keywords using TF-IDF analysis on 35,522 training samples. These keywords then inform prompt engineering for Llama 3.1 8B model fine tuned with LoRA adapters. We incorporate distinctive Bangla terms directly into classification prompts to guide the model understanding of hate speech patterns. Our system achieved top 5 rankings across all three BLP 2025 Task 1 subtasks including hate type classification, target identification, and multi task prediction. The approach proved particularly effective for culturally specific hate speech patterns unique to Bangla social media discourse.

1 Introduction

Hate speech detection in Bangla social media presents unique challenges due to the language's complex morphology and culturally specific expressions of hate. The BLP 2025 Task 1 (Hasan et al., 2025b) addresses this critical need by providing a comprehensive dataset of YouTube comments labeled across multiple dimensions of hate speech. This shared task includes three subtasks that progressively increase in complexity. Subtask 1A requires categorizing text into six hate types including Abusive, Sexism, Religious Hate, Political Hate, Profane, or None. Subtask 1B focuses on identifying the target of hate as Individuals, Organizations, Communities, or Society. Subtask 1C combines both tasks in a multi task learning setup.

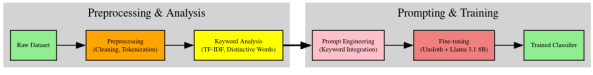

Figure 1: Overview of the Hate Speech Classification Pipeline

We approach these challenges through a unique combination of statistical text analysis and modern language modeling (see Figure 1). Our methodology begins with extensive TF-IDF analysis to identify the most distinctive vocabulary for each hate category. This analysis revealed strong linguistic markers such as religious terms like মুসলিম (muslim), আল্লাহ (allah), and হিন্দু (hindu) for Religious Hate, political party names like লীগ (league) and বিএনপি (BNP) along with ভোট (vote) for Political Hate, and explicit profanity like বাল, শালা for the Profane category. We discovered that certain categories exhibit significantly higher lexical distinctiveness than others. Political and Religious Hate showed average TF-IDF scores above 0.015 for their top keywords, while Abusive and None categories demonstrated more lexical overlap with other classes.

Building on these linguistic insights, we designed category specific prompts that incorporate the identified keywords as examples. This prompt engineering strategy helps the model recognize culturally specific hate patterns that might not be apparent from the text alone. We then fine tuned Llama 3.1 8B using Low Rank Adaptation with rank 64 and alpha 128, training on the full dataset while maintaining computational efficiency through 4 bit quantization. The model processes instructions rather than performing traditional token classification, allowing it to leverage its pretrained knowledge while adapting to Bangla specific hate speech patterns.

Our unified approach achieved competitive performance across all three subtasks, securing top 5 positions in each. The system demonstrated particular strength in identifying explicit profanity with 95 percent recall, though minority classes like Sexism remained challenging due to severe class imbalance. This work contributes both a effective methodology for Bangla hate speech detection and valuable insights into the linguistic patterns of online hate in South Asian social media contexts.

Proceedings of the Second Workshop on Bangla Language Processing (BLP-2025), pages 551–555

2 Related Work

Recent advances in Bangla hate speech detection have explored various neural architectures and multilingual models. Faruqe et al. (2023) employed transformer based models including BERT for hate speech classification, achieving high accuracy on social media texts. Mim et al. (2024) investigated ensemble methods combining CNN with traditional machine learning classifiers for multimodal hate detection from videos. There are works that highlighted the challenge of class imbalance in Bangla datasets.

Cross lingual approaches have shown promise for low resource scenarios. Ghosh and Senapati (2025) demonstrated that XLM-RoBERTa fine tuned on Hindi hate speech transfers reasonably to Bangla. Sharma et al. (2025) emphasized the importance of cultural context in South Asian hate speech, showing that generic multilingual models miss region specific slurs and references.

Recent work on prompt engineering for hate detection includes Prome et al. (2025) who used Llama2-7B with carefully crafted prompts for zero shot classification. However, their approach lacked language specific adaptations. Saha et al. (2024) combined lexicon based features with BERT embeddings, achieving improvements on hate detection. Our work differs by systematically extracting category specific keywords through TF-IDF analysis and incorporating them directly into prompts for instruction tuned models, bridging statistical analysis with modern LLM capabilities.

3 Methodology

3.1 Dataset Processing and Class Distribution

The BLP 2025 dataset (Hasan et al., 2025a) consists of YouTube comments exhibiting natural language variations including code mixing, transliteration, and informal spellings common in social media discourse. The training set contains 35,522 samples with severe class imbalance. The None category dominates with 19,954 samples (56.2%), followed by Abusive with 8,212 (23.1%), Political Hate with 4,227 (11.9%), Profane with 2,331 (6.6%), Religious Hate with 676 (1.9%), and Sexism with merely 122 samples (0.3%). This imbalance posed significant challenges for minority class detection. We maintained original distributions during training rather than synthetic balancing to preserve authentic hate speech patterns.

3.2 Keyword Extraction and Analysis

We begin by extracting category specific keywords from the training corpus $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ where x_i represents the text and $y_i \in \mathcal{C}$ denotes the hate category. For each category $c \in \mathcal{C}$, we compute the TF-IDF scores to identify distinctive vocabulary.

The term frequency for word w in document (a full comment here) d is calculated as:

$$\text{TF}(w, d) = \frac{f_{w,d}}{\sum_{w' \in d} f_{w',d}} \tag{1}$$

where $f_{w,d}$ represents the frequency of word w in document d. The inverse document frequency is:

$$\text{IDF}(w, \mathcal{D}) = \log \frac{|\mathcal{D}|}{|\{d \in \mathcal{D} : w \in d\}|} \tag{2}$$

For category specific analysis, we aggregate TF-IDF scores across all documents belonging to category c:

$$\text{Score}(w, c) = \frac{1}{|\mathcal{D}_c|} \sum_{d \in \mathcal{D}_c} \text{TF-IDF}(w, d) \tag{3}$$

We filter keywords appearing in multiple categories using a cross category threshold $\tau = 2$. A word w is retained for category c only if:

$$|\{c' \in \mathcal{C} : w \in \text{Top}_k(c')\}| \leq \tau \tag{4}$$

where $\text{Top}_k(c)$ denotes the top k words for category c. Our analysis identified 316 Bangla stopwords which were removed during preprocessing.

3.3 Prompt Engineering with Keywords

For each hate category c, we construct prompts incorporating the extracted keywords $K_c = \{w_1, w_2, ..., w_m\}$. The prompt template $P(x, K_c)$ is formulated as:

$$P(x, K_c) = \text{Inst} \oplus \bigcup_{c \in \mathcal{C}} \text{Desc}(c, K_c) \oplus x \oplus \text{Label} \tag{5}$$

where Inst represents task instructions, $\text{Desc}(c, K_c)$ provides category description with example keywords, and $\oplus$ denotes concatenation. Each category description includes the top scoring keywords from our TF-IDF analysis.

3.4 Low Rank Adaptation Fine Tuning

We employ LoRA to efficiently fine tune the Llama 3.1 8B model while preserving its general capabilities. The adaptation modifies weight matrices through low rank decomposition:

$$W' = W_0 + \Delta W = W_0 + BA \tag{6}$$

where $W_0 \in R^{d \times k}$ represents frozen pretrained weights, $B \in R^{d \times r}$ and $A \in R^{r \times k}$ are trainable matrices with rank $r \ll \min(d, k)$. We set $r = 64$ and scaling factor $\alpha = 128$.

The training objective minimizes the cross entropy loss over instruction response pairs:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} \log P(y_i^t | x_i, y_i^{<t}; \theta + \Delta\theta) \tag{7}$$

where $\Delta\theta = \{BA\}$ represents the LoRA parameters. We apply adapters to query, key, value, and output projections in attention layers, as well as the feed forward components.

Training employed gradient accumulation with effective batch size $b_{\text{eff}} = b \times g = 32$ where $b = 8$ is the per device batch size and $g = 4$ is the accumulation steps. We used AdamW optimizer with learning rate $\eta = 2 \times 10^{-4}$ and linear warmup over 10 percent of training steps. The model was quantized to 4 bits using QLoRA for memory efficiency, enabling training on a single Tesla V100 GPU with 32GB VRAM.

3.5 Inference and Prediction

During inference, we generate predictions using greedy decoding with temperature $T = 0$ for deterministic outputs. The predicted category $\hat{y}$ is extracted from the generated text through pattern matching on the instruction following response. For multi task scenarios in Subtask 1C, we parse multiple labels from the structured output format.

4 Results

4.1 Keyword Analysis Findings

Table 1 presents the top distinctive keywords identified through TF-IDF analysis for each hate category on train set. The analysis reveals culturally specific linguistic markers that traditional multilingual models often overlook.

The keyword analysis demonstrates clear lexical separation between categories. Religious and Political Hate exhibit the strongest distinctive vocabularies with average scores exceeding 0.017, while Abusive and None categories show significant overlap with other classes, scoring below 0.008.

Category	Top Keywords (Bangla)	Avg Score
Religious Hate	মুসলিম (0.045), আল্লাহ (0.037), হিন্দু (0.023), ইহুদি (0.022), ইসলাম (0.015)	0.0234
Political Hate	ভোট (0.028), সরকার (0.022), লীগ (0.020), বিএনপি (0.019), আওয়ামী (0.018)	0.0178
Profane	বাল (0.040), শালা (0.015), কুত্তার (0.011), খানকির (0.015), মাদার (0.007)	0.0171
Sexism	নারী (0.042), মহিলা (0.030), মেয়ে (0.021), পুরুষ (0.019), হিজরা (0.017)	0.0221
Abusive	মিথ্যা (0.007), পাগল (0.006), লজ্জা (0.005), চোর (0.008), দালাল (0.013)	0.0080
None	ভাই (0.011), খুব (0.006), দাম (0.005), ঠিক (0.005), সময় (0.007)	0.0069

Table 1: Category-specific keywords with TF-IDF scores

4.2 Classification Performance

able 2 shows the classification results across all three subtasks. Our unified approach achieved competitive performance with consistent results across different hate detection challenges. For Subtask 1C (multi-class classification), we employed a pattern-matching approach where the model directly predicts the next word as the label instead of relying on logits. This method proved more effective, as logits often introduce calibration issues and class imbalance bias, whereas direct next-word prediction aligns better with the generative nature of the model for discrete class outputs.

Subtask	Micro F1	Macro F1	Accuracy
1A: Hate Type	**73.28**	55.6	72.5
1B: Target	**73.17**	55.4	72.3
1C: Multi-task	**73.32**	55.3	72.2

Table 2: Overall performance metrics across subtasks on final test set

4.3 Per-Category Analysis

Detailed classification performance varies significantly across hate categories as shown in Table 3. The model excels at detecting explicit profanity but struggles with minority classes. In the multi-class setting of task 1C, the class imbalances along with multiple output prediction introduce slight confusions between different categories, raising the difficulty for the LLM to disentangle multiple hate

indicators within a single utterance but performs better for easier cases.

Category	Precision	Recall	F1	Support
None	81.3	85.2	83.2	1,451
Profane	78.4	**94.9**	85.9	157
Political Hate	58.2	53.3	55.6	291
Abusive	59.1	52.8	55.8	564
Religious Hate	28.6	21.5	24.6	38
Sexism	50.0	18.2	26.7	11
Weighted Avg	**71.8**	**73.2**	**72.3**	2,512

Table 3: Per-category classification performance on validation set (task 1A)

4.4 Ablation Study

We conducted ablation experiments to assess the contribution of each component in our pipeline. Table 4 demonstrates the importance of keyword-informed prompts.

Configuration	Micro F1	Δ
Full Model	**73.2**	–
w/o Keyword Prompts	70.4	-2.8
w/o TF-IDF Filtering	71.1	-2.1
w/o Stopword Removal	71.6	-1.6
Base Llama (Zero-shot)	42.3	-30.9
LoRA r=32 (vs r=64)	72.8	-0.4

Table 4: Ablation study showing component contributions on validation set (task 1A)

The ablation results highlight that keyword-informed prompts contribute 2.8 points to the final performance. Removing TF-IDF filtering degrades performance by 2.1 points, indicating the importance of category-specific vocabulary selection. The base model without fine-tuning achieves only 42.3% accuracy, primarily predicting the majority None class.

4.5 Discussion

Our results reveal several insights about Bangla hate speech patterns. The high recall for Profane content (94.9%) suggests that explicit profanity follows consistent linguistic patterns easily captured through keyword matching. Political Hate category benefits substantially from domain-specific vocabulary, explaining their strong TF-IDF scores and reasonable detection rates despite class imbalance.

The poor performance on Sexism and Relgious Hate stems from both data scarcity and subtler linguistic expressions. Unlike explicit profanity, gender-based hate often manifests through context-dependent statements requiring deeper semantic understanding. The model struggles to differentiate between legitimate gender discussions and sexist content, frequently misclassifying them as None.

Error analysis reveals that code-mixed content poses particular challenges. Comments mixing Bangla with English or romanized Bangla often escape detection, as our keyword extraction primarily focused on native script. Additionally, sarcastic or indirect hate speech remains problematic, as the model relies heavily on surface-level keyword indicators rather than contextual interpretation.

The consistent performance across subtasks suggests our approach successfully captures general hate patterns applicable to both type classification and target identification. The performance consistency from Subtask 1A to 1C indicates that multi-task prediction did not introduce additional complexity here.

5 Conclusion

This study presents a comprehensive pipeline for Bangla hate speech classification, integrating linguistic analysis with LLM fine-tuning to address the difficulties of multi-class detection. By identifying category-specific keywords via TF-IDF and incorporating them into structured prompts, our Unsloth-optimized Llama 3.1 8B model achieves a micro F1-score of 72.3% on the validation set. This approach not only enhances model interpretability but also bridges gaps in low-resource language NLP. Future work could extend to real-time deployment and cross-lingual transfer, fostering safer online spaces in Bangla-speaking communities. Our contributions underscore the value of hybrid methods for culturally sensitive moderation.

Limitations

Despite promising results, our model faces challenges from severe class imbalance, leading to confusions with broader content. The reliance on keyword-based prompting may overlook subtle evolving slang, potentially introducing biases from the training corpus. Computational demands of fine-tuning large LLMs limit scalability on resource-constrained devices, and evaluation on a single dataset may not generalize to diverse dialects. Addressing these through balanced augmentation and ensemble methods remains essential for robust, equitable hate speech mitigation.

References

Omar Faruqe, Mubassir Jahan, Md Faisal, Md Shahidul Islam, and Riasat Khan. 2023. Bangla hate speech detection system using transformer-based nlp and deep learning techniques. In *2023 3rd Asian Conference on Innovation in Technology (ASIANCON)*, pages 1–6. IEEE.

Koyel Ghosh and Apurbalal Senapati. 2025. Hate speech detection in low-resourced indian languages: An analysis of transformer-based monolingual and multilingual models with cross-lingual experiments. *Natural Language Processing*, 31(2):393–414.

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. *arXiv 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.

Shamrin Jahan Mim, Tanjim Mahmud, Md Hasan Ali, and Mohammad Tarek Aziz. 2024. Stacking ensemble framework for hate speech detection in bangla videos. In *2024 IEEE International Conference on Computing, Applications and Systems (COMPAS)*, pages 1–7. IEEE.

Ruhina Tabasshum Prome, Tarikul Islam Tamiti, and Anomadarshi Barua. 2025. Leveraging the potential of prompt engineering for hate speech detection in low-resource languages. *arXiv preprint arXiv:2506.23930*.

Sagor Kumar Saha, Afrina Akter Mim, Sanzida Akter, Md Mehraz Hosen, Arman Habib Shihab, and Md Humaion Kabir Mehedi. 2024. Bengalihatecb: A hybrid deep learning model to identify bengali hate speech detection from online platform. In *2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, pages 439–444. IEEE.

Deepawali Sharma, Tanusree Nath, Vedika Gupta, and Vivek Kumar Singh. 2025. Hate speech detection research in south asian languages: a survey of tasks, datasets and methods. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 24(3):1–44.