PentaML at BLP-2025 Task 1: Linear Probing of Pre-trained Transformer-based Models for Bangla Hate Speech Detection

Intesar Tahmid[1,*], Rafid Ahmed[1,*], Md Mahir Jawad[1], Anam Borhan Uddin[1], Md Fahim[1,2,†], Md Farhad Alam Bhuiyan[1]

[1]*Penta Global Limited, Bangladesh*
[2]*Center for Computational & Data Sciences*
[*]Equal Contribution [†]Project Lead
Correspondence: {intesar3006, ahmedrafid023, pdcsedu}@gmail.com

Abstract

This paper presents our approach for the BLP Shared Task 1, where we implemented Linear Probing of Pre-trained Transformer-based Models for Bangla Hate Speech Detection. The goal of the task was to customize the existing models so that they're capable of automatically identifying hate speech in Bangla social media text, with a focus on YouTube comments. Our approach relied on fine-tuning several pre-trained BERT models, adapting them to the shared task dataset for improved classification accuracy. To further enhance performance, we applied linear probing on three of the fine-tuned models, enabling more effective utilization of the learned representations. The combination of these strategies resulted in a consistent top-15 ranking across all subtasks of the competition. Our findings highlight the effectiveness of linear probing as a lightweight yet impactful technique for enhancing hate speech detection in low-resource languages like Bangla.

1 Introduction

The rapid growth of numerous online platforms has amplified the prevalence of offensive and harmful content, making automatic hate speech detection a pressing challenge in Natural Language Processing (NLP). For low-resource languages like Bangla, the problem is further complicated by the limited availability of large, high-quality annotated datasets.

To advance research in this direction, the organizers of the BLP Shared Task 1 (Hasan et al., 2025a) curated one of the largest annotated corpora for Bangla hate speech detection. Within this framework, transformer-based architectures such as BanglaBERT (Bhattacharjee et al., 2022) have shown considerable promise, owing to their ability to capture nuanced contextual information in Bangla text.

Our work builds upon these advancements by leveraging fine-tuned transformer models alongside linear probing, a lightweight yet effective technique to exploit learned representations for classification. Through systematic experimentation, we demonstrate that this hybrid approach not only improves generalization but also achieves competitive results, securing consistent top-15 rankings across all subtasks of the competition. This paper details our methodology, experimental setup, and key findings, offering insights into the role of linear probing in enhancing hate speech detection in Bangla.

Subtask	Train	Dev	Test	Task & Num of Classes
1A				Single-label (6)
1B	35,522	2,512	10,200	Single-label (5)
1C				Multi-label: (6, 3, 5)

Table 1: Dataset Statistics Across Subtasks

2 Dataset Details

2.1 Task and Dataset Description

The primary objective of this shared task (Hasan et al., 2025a,b) is to advance robust Bangla hate speech detection through multitask learning. This benchmark aims to capture the deeper linguistic and social dimensions of hateful content in Bangla by requiring models to predict not just the presence of hate, but also its type, intensity, and target group.

The shared task consists of three subtasks: Among the three subtasks both 1A and 1B follow a TSV format with columns `id`, `text`, and `label`. The subtask 1C is annotated with three attributes—`hate_type`, `hate_severity`, and `to_whom`. The severity dimension is categorized into *Little to None*, *Mild*, and *Severe*.

Subtask 1A: Given a Bangla text, the model must identify the *type of hate* expressed in it. The possible categories include *Abusive*, *Sexism*, *Religious Hate*, *Political Hate*, *Profane*, and *None*.

Subtask 1B: This subtask focuses on identifying

Proceedings of the Second Workshop on Bangla Language Processing (BLP-2025), pages 538–543

the *target group* of hate speech. The label space includes *Individuals*, *Organizations*, *Communities*, and *Society*.

Subtask 1C: The final subtask integrates all previous dimensions in a joint learning setup where hate severity is the new addition to the classification task.
Such a multi-dimensional design encourages the development of systems capable of deeper contextual reasoning rather than keyword-based detection. The data were split into training, validation, and test subsets, maintaining balanced class distributions across all subtasks to support fair and meaningful evaluation.

3 Methodology

3.1 Linear Probing Fine-Tuning

Kumar et al. (Kumar et al., 2022) introduced *Linear Probing Fine-Tuning* (LP-FT), a hybrid training method that combines the benefits of linear probing and full fine-tuning. Their work showed that while full fine-tuning of a model—where both the backbone ϕ_M and classifier head ϕ_C are jointly trained—performs well on in-distribution (ID) tasks, it often fails to generalize to out-of-distribution (OOD) settings. In contrast, linear probing freezes the backbone ϕ_M and trains only the classifier ϕ_C, offering better OOD performance but limited adaptability. LP-FT addresses this by first training only the classifier ϕ_C with the frozen backbone ϕ_M, and then fine-tuning both ϕ_M and the pre-trained ϕ_C jointly on the downstream task. This two-step process improves performance across both ID and OOD scenarios.

3.2 FPT based Linear Probing-Fine Tuning

To adapt the model more effectively to our dataset, we leverage the Linear Probing Fine-Tuning (LP-FT) method (Kumar et al., 2022). Since the pre-trained backbone ϕ_M may not fully capture domain-specific knowledge, we first apply a Further Pre-Training (FPT) step using a Masked Language Modeling (MLM) objective. This step is motivated by insights from BanglaTLit (Fahim et al., 2024) and prior ITPT-based winning strategies (Fahim, 2023).

In MLM, a portion (m%) of the input tokens are randomly masked and replaced with a special token [MASK]. The model is then trained to predict the original tokens using contextual cues from the unmasked tokens. This procedure yields a domain-adapted backbone, denoted as ϕ_M^{FPT}.

Following FPT, we apply LP-FT using ϕ_M^{FPT} and a classification head ϕ_C. We consider two-stage training for FPT-based LP-FT. In the first stage, ϕ_M^{FPT} is frozen and only ϕ_C is trained on the downstream task. Once the classifier has been optimized, in the second stage, we jointly fine-tune both ϕ_M^{FPT} and ϕ_C, allowing the entire model (excluding any frozen parameters) to adapt to task-specific features.

4 Experiment Setup

Experimented Models We experimented with multiple transformer architectures, including BanglaBERT (Bhattacharjee et al., 2022), VACBERT (Bhattacharyya et al., 2023), XLM-RoBERTa (Conneau et al., 2020), and IndicBERT (Dabre et al., 2021). For comparison, we also implemented non-transformer baselines using LSTM and Bi-LSTM architectures with attention mechanisms.

Model Configuration Models were trained for 6 epochs using the AdamW optimizer with betas=(0.9, 0.999), epsilon=1e-6, and weight decay regularization. We set learning rate for the encoder to 2e-5. The batch size was set to 4 for larger transformer models to accommodate memory constraints, while smaller models used a batch size of 16. All inputs were truncated/padded to 256 tokens maximum length.
For all the subtasks, we implemented a cosine learning rate scheduler with warmup steps, gradually decreasing from the initial learning rate to a minimum of 1e-6. For Sub-Task 1C specifically, the training loop utilized BCEWithLogitsLoss for multi-label classification and saved the best-performing model based on validation score. Our code implementation featured differentiated parameter optimization, applying layer-specific learning rates and excluding bias/LayerNorm parameters from weight decay.

Linear Probing Model Configuration For the linear probing setup, we employed an MLM architecture where approximately 13% of the input tokens were randomly masked. The cross-entropy loss between the predicted and true tokens was used as the optimization objective. For all coding and model training processes, the PyTorch framework has been used. All experiments were conducted on Kaggle's P100 GPU infrastructure. We

Models	Sub-Task 1A			Sub-Task 1B			Sub-Task 1C		
	Precision	Recall	F1 Score	Precision	Recall	F1 Score	Precision	Recall	F1 Score
Deep Learning Models									
Non Transformer based									
LSTM	26.19	21.58	18.17	21.02	20.99	18.31	45.68	44.21	44.18
Bi-LSTM	36.97	28.45	28.77	47.21	28.13	29.32	49.43	46.47	46.13
LSTM + Attention	35.83	38.63	36.08	43.07	44.73	43.79	48.79	46.98	47.55
Bi-LSTM + Attention	31.70	33.21	30.10	47.75	45.03	45.57	51.19	47.39	48.44
Transformer based LMs									
IndicBERT	48.46	41.99	64.86	52.24	46.78	65.08	48.39	45.30	46.78
VACBERT	61.07	36.93	62.06	47.53	46.40	63.02	50.26	43.22	45.45
XLM-RoBERTa	54.23	50.64	**70.95**	56.45	57.22	70.09	52.29	50.38	51.68
BanglaBERT	55.41	51.82	70.23	61.70	52.54	71.06	54.29	51.54	52.67
Linear Probing Models									
VACBERT-MLM									
+ Linear Probing	45.34	35.87	61.82	50.13	41.39	61.26	47.38	42.71	44.93
+ Linear Probing-FT	46.58	37.90	63.35	51.10	42.72	63.79	49.43	44.46	46.21
IndicBERT-MLM									
+ Linear Probing	48.56	46.38	64.81	51.35	49.27	65.43	52.36	50.17	50.24
+ Linear Probing-FT	50.54	48.65	66.77	53.60	50.97	67.60	54.29	51.54	52.67
BanglaBERT-MLM									
+ Linear Probing	61.37	53.76	70.58	59.24	53.26	70.03	56.41	51.68	53.62
+ Linear Probing-FT	62.32	55.31	70.88	60.70	54.35	**71.23**	57.59	52.17	**54.22**

Table 2: Model benchmarking results on the **Test split** of the SHARED TASK 1 dataset are reported. **Blue** highlights the highest-performing model and submitted during the competition, while **Cyan** marks the best-performing models for each metric across the model types.

employed a training approach consisting of Further Pre-Training (FPT) using Linear Probing Fine-Tuning (LP-FT).

5 Result and Analysis

Non-Transformer based DL Models. For the non-transformer-based deep learning models, we consider both LSTM and Bi-LSTM architectures. We observe that the Bi-LSTM consistently outperforms the standard LSTM across all subtasks, achieving approximately a 10% improvement in F1 score for Subtask 1A, 11% for Subtask 1B, and 2% for Subtask 1C.

Following the approach proposed by (Yu et al., 2020), we incorporate an attention mechanism on top of both LSTM and Bi-LSTM models for text classification. The addition of attention leads to further performance gains. For the LSTM-based model, F1 scores nearly double for Subtask 1A, and improve by approximately 10% for Subtask 1B and 2% for Subtask 1C. Similarly, the Bi-LSTM model with attention shows improvements of 2% for Subtask 1A, around 15% for Subtask 1B, and 2% for Subtask 1C.

Pretrained LMs Performance We also present a comparative analysis of various transformer-based pretrained language models across all three subtasks. Among the models evaluated, XLM-RoBERTa achieves the highest F1 score for Subtask 1A (70.95), showing a balanced performance in both precision and recall. BanglaBERT also performs competitively, particularly in Subtask 1B and Subtask 1C, where it achieves the highest F1 scores (71.06 and 52.67, respectively), along with strong precision and recall values. VACBERT and IndicBERT show competitive results in precision for Subtask 1A but comparatively lower recall, leading to a moderate F1 score. IndicBERT, while showing consistent performance across subtasks, lags behind in terms of overall F1 scores.

These results demonstrate the effectiveness of multilingual transformer models like XLM-RoBERTa and domain-specific models like BanglaBERT in tackling nuanced classification tasks in Bangla.

Impact of Linear Probing For the linear probing experiments, we first further pretrained the models using MLM, as described in Section 3.2. The models evaluated were VACBERT, IndicBERT, and BanglaBERT. We then applied two training strategies: linear probing alone, and linear probing followed by fine-tuning (FT).

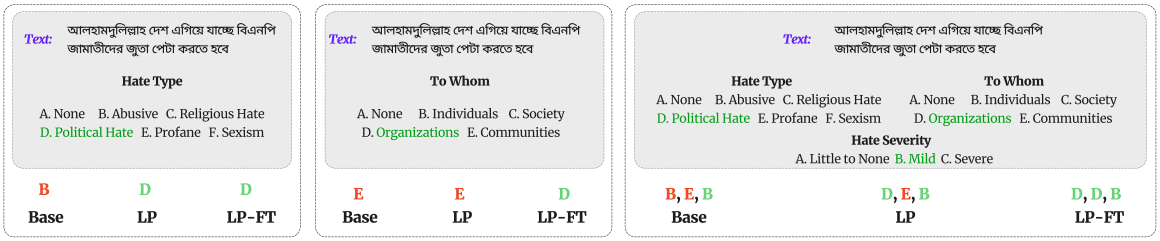

Figure 1: Error analysis of BanglaBERT variants for hate speech classification, comparing base model, linear probing (LP), and linear probing with full fine tuning (LP-FT) across hate type categories and target demographics.

The results show that linear probing by itself generally underperforms compared to the standard fine-tuning approach presented earlier. However, when linear probing is combined with fine-tuning, the models achieve substantial performance improvements, often surpassing the standard fine-tuning baseline across most models.

Notably, BanglaBERT-MLM with linear probing followed by fine-tuning achieved the highest F1 scores across all subtasks—70.88 for Subtask 1A, 71.23 for Subtask 1B, and 54.22 for Subtask 1C—alongside superior precision and recall in most cases. IndicBERT-MLM also showed notable gains with this combined strategy, consistently outperforming both linear probing alone and standard fine-tuning. Although VACBERT-MLM improved with the combined approach, it still lagged behind the other two models in overall performance.

Therefore, our benchmarking highlights two optimal models tailored to the subtasks: XLM-RoBERTa excels in Subtask 1A, while BanglaBERT-MLM with linear probing followed by fine-tuning (LP-FT) outperforms others in Subtasks 1B and 1C. This suggests that although a powerful multilingual transformer like XLM-RoBERTa is effective for certain tasks, leveraging a monolingual, language-specific model with targeted training strategies such as LP-FT can yield superior results for nuanced tasks in low-resource languages like Bangla. We also provide the result for the validation set in SHARED TASK 1 in Table 3 in the Appendix A.

6 Error Analysis

Figure 1 highlights model confusions across hate type, target, and severity using a representative example. The errors mainly arise from overlapping categories—such as political and abusive hate—and ambiguous phrasing that blurs target boundaries between individuals and communities. The base model often misinterprets these subtleties, while LP improves class awareness through better feature separation. LP-FT further refines contextual understanding, reducing cross-category confusion and capturing implicit hate cues more effectively, thereby aligning predictions closer to human interpretation.

7 Conclusion

In this work, we systematically evaluated Linear Probing followed by Full Fine-Tuning (LP-FT) combined with Further Pre-Training (FPT) for Bengali hate speech classification. Our results demonstrate that BanglaBERT-MLM with LP-FT achieves notable performance gains, highlighting the effectiveness of this two-stage adaptation strategy. This finding establishes that progressive fine-tuning, beginning with a stabilized feature space, is a powerful paradigm for enhancing model robustness in low-resource language contexts, paving the way for more effective and responsible content moderation.

References

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.

Pramit Bhattacharyya, Joydeep Mondal, Subhadip Maji, and Arnab Bhattacharya. 2023. Vacaspati: A diverse corpus of bangla literature. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, page

1118–1130. 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.

Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh M. Khapra, and Pratyush Kumar. 2021. Indicbart: A pre-trained model for natural language generation of indic languages. *CoRR*, abs/2109.02903.

Md Fahim. 2023. Aambela at blp-2023 task 2: Enhancing banglabert performance for bangla sentiment analysis task with in task pretraining and adversarial weight perturbation. In *Proceedings of the First Workshop on Bangla Language Processing (BLP-2023)*, pages 317–323.

Md Fahim, Fariha Shifat, Fabiha Haider, Deeparghya Barua, Md Sourove, Md Ishmam, and Md Bhuiyan. 2024. Banglatlit: A benchmark dataset for back-transliteration of romanized bangla. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14656–14672.

Md Arid Hasan, Firoj Alam, Md Fahad Hossain, Usman Naseem, and Syed Ishtiaque Ahmed. 2025a. Blp 2023 task 1: Hate speech identification. In *Proceedings of the Second International Workshop on Bangla Language Processing (BLP-2025)*, India. Association for Computational Linguistics.

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

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. 2022. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*.

Shujuan Yu, Danlei Liu, Wenfeng Zhu, Yun Zhang, and Shengmei Zhao. 2020. Attention-based lstm, gru and cnn for short text classification. *Journal of Intelligent & Fuzzy Systems*, 39(1):333–340.

A Result on Validation Set

Table 2 presents a comprehensive comparison of model performances on the validation split of the SHARED TASK 1 dataset, spanning non-transformer deep learning models, transformer-based language models, and linear probing variants.

Among the non-transformer models, the addition of an attention mechanism significantly improves performance. Specifically, Bi-LSTM with attention achieves the highest F1 scores within this group across all subtasks, demonstrating the value of incorporating attention for sequence modeling.

Transformer-based models outperform all non-transformer counterparts, with XLM-RoBERTa achieving the best F1 score in Subtask 1B (57.24), and strong performance across the other subtasks. BanglaBERT shows competitive results as well, particularly excelling in recall metrics, which suggests effective representation learning for Bangla text.

The linear probing experiments further highlight the benefits of task-specific adaptation. While linear probing alone improves over some baseline transformer models, the combination of linear probing with fine-tuning (LP-FT) yields the best results overall. BanglaBERT-MLM with LP-FT achieves the highest F1 scores for Subtasks 1A (56.75) and 1C (55.23), and remains highly competitive in Subtask 1B. IndicBERT-MLM also benefits from this training strategy, consistently improving across metrics.

Highlighted cells denote the best-performing models within each experimental category, as well as the top scores overall. These findings underscore the effectiveness of combining pretrained transformer architectures with targeted fine-tuning approaches, especially for challenging tasks in low-resource languages like Bangla.

Models	Sub-Task 1A			Sub-Task 1B			Sub-Task 1C		
	Precision	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**
Deep Learning Models									
Non Transformer based									
LSTM	9.62	16.66	12.20	12.22	20	15.17	13.50	17.80	15.10
Bi-LSTM	41.85	18.13	15.24	36.94	22.15	19.20	38.24	28.50	29.81
LSTM + Attention	35.28	34.12	32.44	40.71	41.04	40.63	45.29	43.93	44.79
Bi-LSTM + Attention	43.62	40.28	42.45	44.19	46.38	47.74	43.51	42.49	43.17
Transformer based LMs									
IndicBERT	46.04	40.86	43.00	51.53	47.16	49.02	47.92	44.31	45.69
VACBERT	43.42	40.46	41.66	46.97	46.38	46.55	47.62	45.07	45.74
XLM-RoBERTa	60.44	54.28	56.58	57.03	57.65	**57.24**	53.74	51.62	52.67
BanglaBERT	56.59	54.90	54.44	59.66	52.81	54.77	54.14	54.60	54.25
Linear Probing Models									
VACBERT-MLM									
+ Linear Probing	47.35	38.73	41.57	51.68	43.54	45.18	47.43	43.89	45.74
+ Linear Probing-FT	48.55	39.42	42.65	52.29	44.20	47.19	49.75	44.79	46.67
IndicBERT-MLM									
+ Linear Probing	53.47	47.39	51.78	52.19	49.63	51.27	53.72	50.16	51.47
+ Linear Probing-FT	54.16	50.98	52.37	53.41	50.83	52.00	55.15	51.00	52.11
BanglaBERT-MLM									
+ Linear Probing	56.78	55.49	55.87	60.17	53.98	56.24	52.87	57.61	54.97
+ Linear Probing-FT	57.50	56.89	**56.75**	60.55	54.27	56.72	53.98	58.21	**55.23**

Table 3: Model benchmarking results on the **Validation split** of the SHARED TASK 1 dataset are reported. **Blue** highlights the highest-performing model and submitted during the competition, while **Cyan** marks the best-performing models for each metric across the model types.