LP-FT-LoRA : A Three-Stage PEFT Framework for Efficient Domain Adaptation in Bangla NLP Tasks

Tasnimul Hossain Tomal[1], Anam Borhan Uddin[1], Intesar Tahmid[1], Mir Sazzat Hossain[1,2], Md Fahim[1,2,†], Md Farhad Alam[1]

[1]*Penta Global Limited* [2]*Center for Computational & Data Sciences*
[†] *Project Lead*

Correspondence: {tomal.tasnimul,fahimcse381}@gmail.com

Abstract

Adapting large pre-trained language models (LLMs) to downstream tasks typically requires fine-tuning, but fully updating all parameters is computationally prohibitive. Parameter-Efficient Fine-Tuning (PEFT) methods like Low-Rank Adaptation (LoRA) reduce this cost by updating a small subset of parameters. However, the standard approach of jointly training LoRA adapters and a new classifier head from a cold start can lead to training instability, as the classifier chases shifting feature representations. To address this, we propose LP-FT-LoRA , a novel three-stage training framework that decouples head alignment from representation learning to enhance stability and performance. Our framework first aligns the classifier head with the frozen LM backbone via linear probing, then trains only the LoRA adapters to learn task-specific features, and finally performs a brief joint refinement of the head and adapters. We conduct extensive experiments on five Bangla NLP benchmarks across four open-weight compact transformer models. The results demonstrate that LP-FT-LoRA consistently outperforms standard LoRA fine-tuning and other baselines, achieving state-of-the-art average performance and showing improved generalization on out-of-distribution datasets. Code for this paper is available at https://github.com/tomal66/lp-ft-lora.

1 Introduction

The paradigm of pre-training and fine-tuning has become the de-facto standard for natural language processing (NLP), with large language models (LLMs) based on the Transformer architecture (Devlin et al., 2019) demonstrating remarkable capabilities across a wide array of tasks (Zhao et al., 2024). To adapt these powerful but general-purpose models to specific downstream applications, fine-tuning is essential. However, updating all the parameters of a multi-billion parameter model is computationally extensive, requiring substantial memory and GPU resources. This has spurred the development of Parameter-Efficient Fine-Tuning (PEFT) methods.

PEFT techniques aim to adapt LLMs by updating only a small fraction of their total parameters, drastically reducing the computational burden while often matching or even exceeding the performance of full fine-tuning. Among these, Low-Rank Adaptation (LoRA) (Hu et al., 2022) has emerged as a particularly effective and widely adopted method. LoRA injects trainable low-rank matrices into the model's layers, allowing for efficient task-specific adaptation without modifying the original pre-trained weights.

Concurrently, linear probing remains a canonical and lightweight transfer learning protocol (Kumar et al., 2022). In this approach, the entire pre-trained LM backbone is frozen, and only a newly added classification head is trained. While fast and memory-efficient, its success hinges on the strong assumption that the downstream task's classes are already linearly separable in the model's frozen feature space. This assumption often breaks down under significant domain shifts, limiting its effectiveness for more complex adaptation scenarios.

Although both LoRA and linear probing are powerful, they present distinct challenges when applied in isolation. Standard LoRA fine-tuning, which typically involves jointly training the LoRA adapters (ϕ_{LoRA}) and a randomly initialized classifier head (ϕ_C), can suffer from training instability. The classifier head must learn to interpret features that are themselves being modified, a "moving target" problem that can lead to noisy gradients and slow convergence (Rajput and Mehta, 2025). On the other hand, linear probing's inability to adapt the backbone's representations makes it unsuitable for tasks where the pre-trained features are insufficient.

In this paper, we identify and address a critical gap: the need for a framework that systematically stabilizes the fine-tuning process while en-

Proceedings of the Second Workshop on Bangla Language Processing (BLP-2025), pages 212–222

abling robust representation learning. We propose **LP-FT-LoRA** , a novel three-stage fine-tuning framework that explicitly decouples classifier head alignment from adapter-based representation learning. LP-FT-LoRA mitigates the instabilities of standard LoRA fine-tuning and overcomes the representational rigidity of simple linear probing. Our contributions are threefold:

- We introduce LP-FT-LoRA , a novel three-stage training framework that synergistically combines linear probing and LoRA for efficient and stable adaptation of LLMs.
- Through extensive experiments on five Bengali NLP datasets and four different model architectures, we demonstrate that LP-FT-LoRA consistently outperforms standard LoRA fine-tuning and other strong baselines.
- We provide a detailed analysis of the framework's robustness to out-of-distribution data and conduct comprehensive ablation studies to dissect the impact of key hyperparameters.

The rest of this paper is organized as follows: Section 2 reviews related work, Section 3 provides preliminaries on the core techniques, Section 4 details our proposed method, Section 5 describes the experimental setup, Section 6 presents our results and analysis, and Section 7 concludes the paper.

2 Related Work

In this section, we situate our approach within prior work on linear probing, parameter-efficient fine-tuning, and multi-stage fine-tuning, highlighting how existing methods motivate and contrast with our proposed framework.

2.1 Linear Probing

Linear probing and fine-tuning are canonical transfer learning protocols, where staged approaches like LP-FT improve performance and preserve representations under distribution shifts (Tomihari and Sato, 2024; Kumar et al., 2022). Advances in linear probing evaluation have introduced more robust metrics and demonstrated its utility in separating evaluation contexts from deployment prompts (Thilak et al., 2024; Nguyen et al., 2025).

2.2 Parameter Efficient Fine-Tuning

Parameter-efficient fine-tuning (PEFT) methods, such as adapter modules and the widely adopted LoRA, adapt large models by updating a small fraction of parameters (Houlsby et al., 2019; Hu et al., 2022). Recent LoRA variants enhance performance by using different learning rates, dynamically allocating parameter budgets based on layer importance, or employing layer-wise adaptive rank allocation (Hayou et al., 2024; Mao et al., 2024; Gu et al., 2025). Other extensions explore multiplicative updates or combine quantization with adaptive rank selection for highly compressed models (Bihany et al., 2025; Kim et al., 2024). The theoretical underpinnings for these methods are provided by the delta-tuning framework, with recent insights also revealing benefits of non-zero initialization practices (Ding et al., 2023; Li et al., 2025).

2.3 Multi-stage Fine-tuning

Multi-stage fine-tuning has been explored through progressive frameworks that mitigate catastrophic forgetting and in continual learning settings that manage knowledge conflicts (Hou et al., 2024; Guan et al., 2025). Hybrid PEFT methods and advanced multi-task architectures enable uncertainty quantification and fine-grained task specialization (Chai et al., 2025; Ning et al., 2025). While linear probing and LoRA are each well studied, a single multi-stage framework that integrates linear probing with LoRA fine-tuning for domain-specialized classification appears to be unaddressed.

3 Preliminaries

This section formalizes the linear probing fine-tuning and LoRA fine-tuning protocols and delineates the research scope that motivates our proposed framework.

3.1 Linear Probing Fine-Tuning

Kumar et al. (Kumar et al., 2022) explored the theoretical foundations and operational mechanisms of linear probing, particularly in the context of out-of-distribution (OOD) tasks. Their study also compared the performance of linear probing and full fine-tuning. The experimental results demonstrated that while fine-tuning outperforms linear probing on in-distribution (ID) tasks, it struggles with generalization on OOD tasks. Based on these observations, the authors proposed a hybrid approach called *Linear Probing Fine-Tuning* (LP-FT).

Omitting theoretical details, the operational workflow of LP-FT is as follows. Given a pre-trained LM backbone denoted as ϕ_M, a classifier

head ϕ_C is appended on top. In standard fine-tuning, the entire network $[\phi_{LM}, \phi_C]$ is jointly trained based on the loss from the downstream task. In contrast, linear probing keeps ϕ_{LM} frozen and trains only the classifier ϕ_C. LP-FT introduces a two-stage training strategy:

Stage 1 (Linear Probing): The backbone ϕ_{LM} is frozen, and only the classifier ϕ_C is trained using the downstream loss.

Stage 2 (Fine-Tuning): Both the backbone ϕ_{LM} and the previously trained classifier ϕ_C are jointly fine-tuned on the downstream task.

This staged approach utilizes the robustness of linear probing in the initial phase and the representational flexibility of fine-tuning in the later phase, resulting in improved generalization across both ID and OOD settings.

3.2 LoRA Fine-Tuning

Low-Rank Adaptation (LoRA) (Hu et al., 2022) is a Parameter-Efficient Fine-Tuning (PEFT) technique designed to reduce the computational and memory overhead associated with traditional fine-tuning of large pre-trained models. Rather than updating all model parameters, LoRA introduces a low-rank decomposition to capture task-specific adaptations while training a classifier head jointly for the target task. A visual architecture of the procedure is shown in Figure 1a.

Let the pre-trained model backbone be denoted as ϕ_{LM}, with its associated weight matrix $\mathbf{W}$. During LoRA fine-tuning, $\mathbf{W}$ remains frozen. Instead of directly updating $\mathbf{W}$, LoRA introduces a learnable, low-rank update matrix $\Delta\mathbf{W}$, defined as:

$$\Delta\mathbf{W} = \mathbf{A}\mathbf{B}^T$$

where $\mathbf{A} \in \mathbb{R}^{d\times r}$ and $\mathbf{B} \in \mathbb{R}^{r\times d}$, with $r \ll d$. This low-rank factorization ensures that the number of additional trainable parameters is significantly smaller than in full-rank updates. The matrices $\mathbf{A}$ and $\mathbf{B}$ encode the task-specific information required for adaptation, with $\mathbf{A}$ representing learned transformations across output dimensions and $\mathbf{B}$ across input dimensions. We denote the LoRA adapter parameters as $\phi_{LoRA} = \{\mathbf{A}, \mathbf{B}\}$.

The final adapted weight matrix $\mathbf{W}'$ used during inference is given by:

$$\mathbf{W}' = \mathbf{W} + \Delta\mathbf{W} = \mathbf{W} + \mathbf{A}\mathbf{B}^T$$

In addition to the LoRA adapters ϕ_{LoRA}, a task-specific classifier head $\phi_C \in \mathbb{R}^{h\times C}$ is introduced, where h is the hidden dimension of the backbone and C is the number of classes. This classifier head is randomly initialized and trained jointly with ϕ_{LoRA}, enabling the model to learn both feature adaptations and classification mappings simultaneously during fine-tuning.

3.3 Research Scope

With the backbone ϕ_{LM} frozen in Linear Probing Fine-Tuning, training only the head ϕ_C implicitly assumes downstream classes are linearly separable in the pretrained feature space. Therefore, for under domain shift, this approach often fails, and ϕ_C can merely reweight insufficient features.

In the case of LoRA Fine-Tuning, jointly optimizing $\{\phi_{\text{LoRA}}, \phi_C\}$ from a cold start requires representation learning and classification. It induces noisy gradients and acute sensitivity to LoRA rank r and learning rate (Hayou et al., 2024). Furthermore, if ϕ_C is trained jointly from scratch, the head can chase moving features, which shifts ϕ_{LM}'s pretrained representations, resulting in slow convergence (Tomihari and Sato, 2024).

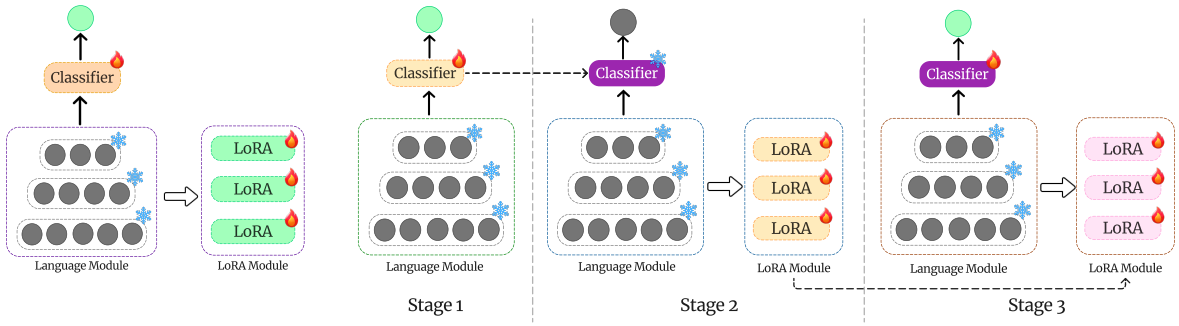
In our proposed framework LP-FT-LoRA , we mitigate the above-mentioned issues through a three-stage fine-tuning process. To the best of our knowledge, LP-FT-LoRA is the first framework to combine *linear probing* of the classifier (ϕ_C), *LoRA-only* probing of adapters (ϕ_{LoRA}) on a frozen backbone (ϕ_{LM}), and a brief joint refinement of $\phi_{\text{LoRA}}, \phi_C$ for explicit decoupling of head alignment from adapter representation learning.

4 Proposed Method: LP-FT-LoRA

In this work, we propose LP-FT-LoRA , a three-stage training framework that integrates LP-FT into a LoRA-augmented network. The objective is to enable efficient and effective adaptation to downstream tasks through structured, progressive training.

Let ϕ_{LM} represent the frozen pre-trained language model backbone. The LoRA-specific trainable parameters associated with this model are denoted as ϕ_{LoRA}, and the classifier head is represented by ϕ_C. The overall model architecture can thus be described as $[\phi_{LM}, \phi_{\text{LoRA}}, \phi_C]$. The overall design is visualized in Figure 1b. The training process proceeds in the following three stages:

Stage 1: Linear Probing. In the initial stage,



(a) Std. LoRA FT

(b) Proposed LP-FT-LoRA

Figure 1: Standard LoRA Fine-tuning and proposed LP-FT-LoRA architecture

the pretrained language model's backbone ϕ_{LM} remains frozen, and a classifier head ϕ_C is added on top. Only the classifier ϕ_C is trained using a downstream loss function—specifically, the cross-entropy loss in our classification setup. This step allows the classifier to adapt to the output space of the frozen backbone.

Stage 2: LoRA Linear Probing. In this stage, we initialize the LoRA parameters ϕ_{LoRA} with randomly initialized low-rank matrices $\mathbf{A}$ and $\mathbf{B}$, inserted into the backbone ϕ_{LM}. The previously trained classifier ϕ_C is retained, but both ϕ_{LM} and ϕ_C are kept frozen during this phase. Only the LoRA parameters are updated. The goal of this stage is to enable the LoRA layers to learn task-specific or domain-specific representations without modifying the backbone or classifier.

Stage 3: Fine-Tuning. In the final stage, we jointly train the LoRA parameters ϕ_{LoRA} and the classifier head ϕ_C, while continuing to keep the backbone ϕ_{LM} frozen. Crucially, the training resumes from the previously learned weights: ϕ_{LoRA} from Stage 2 and ϕ_C from Stage 1. This step refines both the adaptation layers and the classifier for improved performance on the downstream task.

5 Experimental Setup

This section details the datasets, model architectures, training configuration, and baseline methods used to evaluate LP-FT-LoRA .

5.1 Datasets

We evaluate LP-FT-LoRA across five Bangla NLP benchmarks covering fake news detection, sarcasm detection, sentiment analysis, and emotion recognition. The datasets are:

- **BanFakeNews** (Hossain et al., 2020): A fake news detection dataset containing approximately 48K authentic and 1K fake Bangla news articles across multiple categories. The task involves binary classification to determine whether a news article is authentic or fake.
- **Sarcasm Detection**: A Kaggle competition dataset[1] comprising around 50K news headlines labeled as either Sarcastic (1) or Not-Sarcastic (0).
- **SentNoB** (Islam et al., 2021): A sentiment analysis dataset of public comments collected from social media on news and videos, labeled as Positive, Negative, or Neutral. The dataset contains 13.5K training samples with 1.5K validation and 1.5K test samples across 13 different domains.
- **Emotion Detection** (Irtiza Tripto and Eunus Ali, 2018): A YouTube comments dataset for emotion classification with five categories: anger/disgust, joy, sadness, fear/surprise, and none. The dataset captures diverse emotional expressions in Bangla user-generated content.
- **Sentiment Classification** (Irtiza Tripto and Eunus Ali, 2018): A fine-grained sentiment analysis task using the same comment corpus as emotion detection, with five sentiment classes: Strongly Positive, Positive, Neutral, Negative, and Strongly Negative.
- **EmoNoBa** (Islam et al., 2022): A dataset for fine-grained, multi-label emotion analysis on

[1] https://www.kaggle.com/competitions/nlp-competition-cuet-ete-day-2022/data

noisy Bangla texts collected from social media. It includes labels for six basic emotions: joy, sadness, anger, disgust, fear, and surprise.

- **BanglaSarc** (Apon et al., 2022): A dataset for sarcasm detection in Bangla, compiled from comments on public Facebook posts. The task is a binary classification to identify text as either sarcastic or not sarcastic.

5.2 Model Architecture

We evaluate LP-FT-LoRA across four transformer-based open-weight models ranging from 360M to 1.5B parameters, covering compact to small scales. The selected backbones are: `SmolLM2-360M`, `Qwen3-0.6B`, `Gemma3-1B`, and `Qwen2.5-1.5B`. Their specifications are summarized in Table 1.

Model	Params	Layers	Hidden
SmolLM2-360M	360M	24	960
Qwen3-0.6B	600M	24	1024
Gemma3-1B	1B	18	2048
Qwen2.5-1.5B	1.5B	28	1536

Table 1: Architecture specifications of backbone models.

For adaptation, we apply LoRA with task-specific classifier heads that match hidden dimensions and output 2, 3, or 5 classes depending on the task.

5.3 Training Configuration

On each stage, we train with *AdamW* optimizer under a cosine learning rate schedule with a warm-up ratio of 0.03. We use a maximum sequence length of 512, a base learning rate of 2×10^{-4}, 4 training epochs, and a per-device batch size of 8 with gradient accumulation of 8 (effective batch size $8 \times 8 = 64$ sequences per update on a single device). We employ the SDPA attention path for training in this framework.

For LoRA, we use rank $r = 16$, targeting attention and MLP projections. The scaling factor α is 16 across all models, with a dropout rate of 0.05. All experiments are conducted on a single NVIDIA Tesla P100 (16 GB) GPU in the Kaggle environment. Implementations use Python 3.11 with PyTorch, Hugging Face `transformers`, `datasets`, `accelerate`, and `peft`.

5.4 Baseline Methods

We compare LP-FT-LoRA against the following baseline approaches:

- **Linear Probing (LP):** Training only the classifier head ϕ_C while freezing backbone ϕ_{LM}.
- **Standard LoRA Fine-Tuning:** Joint training of LoRA parameters ϕ_{LoRA} and classifier ϕ_C from random initialization.
- **LoRA Linear Probing:** Training of LoRA parameters ϕ_{LoRA} and keeping the classifier ϕ_C frozen.

6 Result and Analysis

This section presents the empirical results of LP-FT-LoRA and analyzes their implications across tasks, models, and baselines along with detailed ablation studies.

6.1 Performance of LP-FT-LoRA

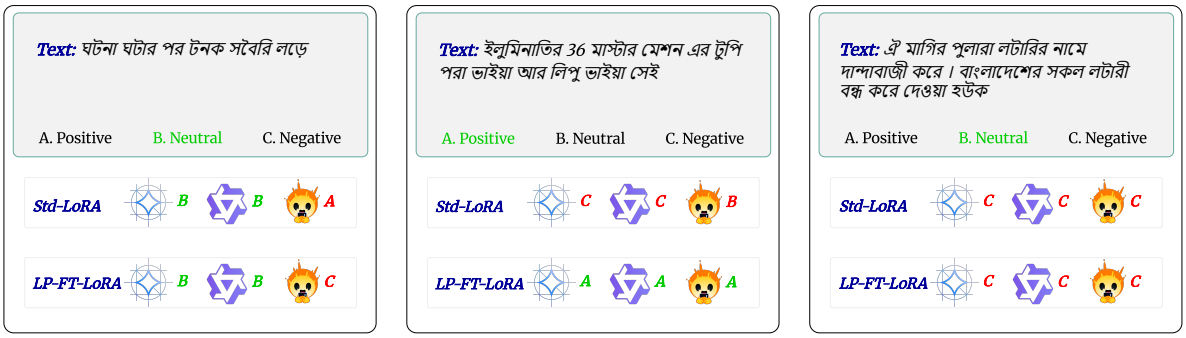
Analysis Across Datasets. As shown in Table 2, LP-FT-LoRA demonstrates consistent improvements across diverse task types. On the *BanFake* fake news detection dataset, LP-FT-LoRA achieves the best performance across all four backbone models, with accuracies ranging from 94.83% (SmolLM2-360M) to 98.82% (Gemma3-1B), outperforming Standard-LoRA-FT by 0.82–3.05 percentage points. For the Emotion classification task, LP-FT-LoRA consistently secures the top position on Qwen3-0.6B and Gemma3-1B, showing notable gains over Standard-LoRA-FT, with the most significant improvement observed on Qwen3-0.6B (58.48% vs. 52.25%). On SmolLM2-360M, LP-FT-LoRA achieves the second-best emotion score (50.19%), marginally behind Standard-LoRA-FT (51.03%). The Sarcasm detection task also favors LP-FT-LoRA , achieving best or tied-best results across all models with accuracies between 93.65% and 95.36%. In contrast, on the Sentiment classification task, LP-FT-LoRA exhibits more mixed results, achieving the best score on SmolLM2-360M (58.81%), while being slightly outperformed by Standard-LoRA-FT or LoRA Linear Probing on other models. This variability suggests that the three-stage training approach is particularly effective for tasks requiring nuanced semantic understanding and binary classification (emotion, fake news, sarcasm) but may offer diminishing returns on certain fine-grained multi-class sentiment distinctions.

Models	SentNoB	BanFake	Sarcasm	Emotion	Sentiment	Avg
Qwen3-0.6B						
Standard-LoRA-FT	69.99%	96.59%	94.37%	52.25%	**65.21%**	75.68%
Linear Probing	58.89%	87.66%	90.21%	42.91%	56.00%	67.13%
LoRA Linear Probing	68.10%	97.18%	94.41%	54.33%	64.87%	75.78%
LP-FT-LoRA	**71.31%**	**97.41%**	**94.54%**	**58.48%**	63.52%	**77.05%**
Gemma3-1B						
Standard-LoRA-FT	71.69%	95.77%	95.26%	52.94%	63.97%	75.93%
Linear Probing	63.18%	94.36%	91.96%	43.60%	55.44%	69.71%
LoRA Linear Probing	71.25%	98.47%	**95.36%**	53.98%	**65.10%**	76.83%
LP-FT-LoRA	**72.07%**	**98.82%**	**95.36%**	**55.02%**	63.75%	**77.00%**
SmolLM2 360M						
Standard-LoRA-FT	**67.91%**	93.15%	91.01%	**51.03%**	57.19%	72.06%
Linear Probing	50.44%	84.72%	88.37%	23.18%	45.23%	58.39%
LoRA Linear Probing	62.61%	92.48%	92.72%	38.64%	53.76%	68.04%
LP-FT-LoRA	65.70%	**94.83%**	**93.65%**	50.19%	**58.81%**	**72.64%**
Qwen2.5-1.5B						
Standard-LoRA-FT	**69.36%**	96.12%	94.45%	**56.10%**	62.74%	75.75%
Linear Probing	59.21%	86.49%	90.02%	32.18%	51.07%	63.79%
LoRA Linear Probing	68.60%	96.94%	94.37%	53.29%	**65.21%**	75.68%
LP-FT-LoRA	67.91%	**97.88%**	**94.68%**	56.06%	62.96%	**75.90%**

Table 2: Evaluation results (accuracy) on five datasets (SentNoB, BanFake, Sarcasm, Emotion, Sentiment) for four base models under different training strategies. **Bold** marks the best score within each model; underline marks the second-best.

Model-wise Comparison. Across all four backbone architectures, LP-FT-LoRA achieves the highest average accuracy: Qwen3-0.6B (77.05%), Gemma3-1B (77.00%), SmolLM2-360M (72.64%), and Qwen2.5-1.5B (75.90%), demonstrating its robustness across different model families and parameter scales. Notably, on Qwen3-0.6B and Gemma3-1B, LP-FT-LoRA outperforms Standard-LoRA-FT by 1.37 and 1.07 percentage points, respectively. On SmolLM2-360M, LP-FT-LoRA surpasses Standard-LoRA-FT by 0.58 percentage points (72.64% vs. 72.06%), while on Qwen2.5-1.5B, the improvement is 0.15 percentage points (75.90% vs. 75.75%). Compared to the two-stage LoRA Linear Probing approach, LP-FT-LoRA consistently delivers superior performance, with improvements of 0.17–4.60 percentage points, indicating that the additional fine-tuning stage (Stage 3) provides meaningful refinement. The gap between LP-FT-LoRA and Linear Probing is substantial across all models, underscoring the critical role of LoRA adaptation in the proposed framework.

Impact of Model Size. The experimental results reveal a non-linear relationship between model size and the effectiveness of LP-FT-LoRA . The 360M-parameter SmolLM2 achieves an average accuracy of 72.64%, while increasing model size to 600M (Qwen3-0.6B) yields a substantial 4.41-point improvement to 77.05%. Further scaling to 1B parameters (Gemma3-1B) provides only marginal changes (77.00%, essentially equivalent performance), and the 1.5B-parameter Qwen2.5 model achieves 75.90%, lower than the smaller Qwen3-0.6B and Gemma3-1B. This pattern suggests that LP-FT-LoRA is highly effective at extracting task-specific knowledge from compact models (600M-1B range), where the progressive training stages can efficiently leverage limited capacity. The diminishing returns at 1.5B parameters may indicate that larger models require different optimization strategies or that the chosen datasets reach a performance ceiling around 76-77% average accuracy, regardless of increased model capacity. Interestingly, Gemma3-1B and Qwen3-0.6B achieve nearly identical performance despite a significant parameter difference, suggesting that architectural design and pre-training quality may matter as much as raw model size for Bangla NLP tasks.



(a) (b) (c)

Figure 2: Error analysis with Gemma3-1B, Qwen3-0.6B and SmolLM2-360M on three samples from the *SentNoB* dataset.

6.2 Error Analysis.

To assess LP-FT-LoRA 's effectiveness, we analyze three challenging examples from the *SentNoB* dataset using Gemma3-1B, Qwen3-0.6B, and SmolLM2-360M with both Standard LoRA (Std-LoRA) and LP-FT-LoRA . We identify two main error patterns: (i) confusion between similar sentiment classes and (ii) difficulty with sparse or conflicting sentiment cues.

The first example in Figure 2a shows both methods correctly predicting the label for straightforward text with clear polarity. This suggests LP-FT-LoRA maintains baseline performance on simple cases. In the second example (Figure 2b), Std-LoRA incorrectly predicts Negative/Neutral due to misleading named entities, while LP-FT-LoRA correctly identifies the Positive sentiment. This demonstrates LP-FT-LoRA 's improved handling of deceptive lexical cues.

The third example (Figure 2c) contains Bangla slang and strong language within a policy statement. The gold label is Neutral, but both methods predict Negative. This reveals a common bias where informal language and emphatic expressions override the actual discourse intent.

6.3 Ablation Studies

We ablate one hyperparameter at a time around a fixed configuration: LoRA rank r=16, LoRA scaling α=32, *Attn+MLP* target modules, batch size 8, learning rate 2×10^{-4}, and 4 epochs. The Macro-F1 scores are visualized in Figure 3a for LoRA and Figure 3b for training hyperparameters.

Impact of LoRA Hyperparameters

Rank. On *Sentiment*, performance peaks near the baseline r=16, with both lower (r=8) and higher (r=32) ranks underperforming. On *Emotion*, a smaller adaptation (r=8) is preferable, while larger ranks yield diminishing returns. Overall, moderate rank values are most reliable across datasets (Figure 3a).

Scaling factor α. Increasing α improves robustness. For *Sentiment*, α=64 delivers the strongest scores, substantially surpassing smaller values. *Emotion* also benefits from a higher scale, with α=64 slightly outperforming α=16 and α=32. This suggests that stronger LoRA scaling helps the adapter better fit both tasks when other settings are fixed (Figure 3a).

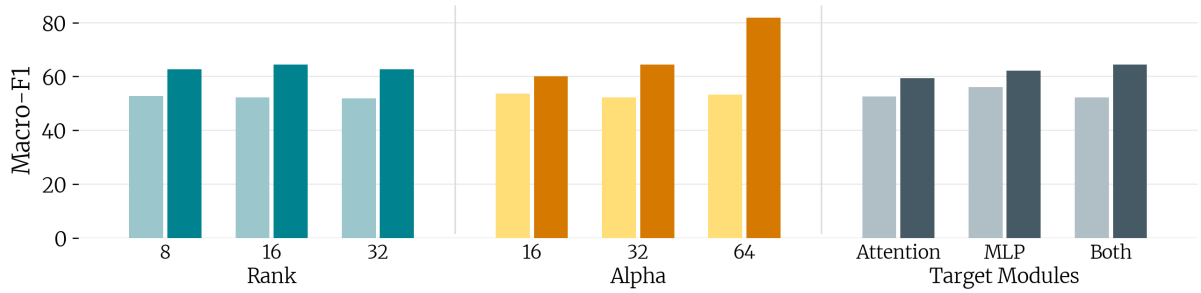
Target modules. Updating both *Attention* and *MLP* consistently outperforms targeting a single block on *Sentiment*. For *Emotion*, adapting only *MLP* edges out other choices, indicating dataset-specific sensitivities to where capacity is added. In practice, *Attention+MLP* is a safe default; *MLP*-only can be competitive for fine-grained tasks (Figure 3a).

Impact of Training Hyperparameters

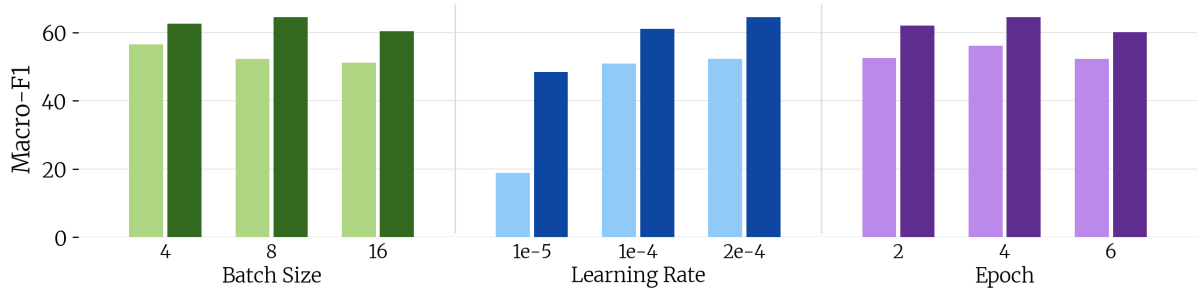
Batch size. The baseline 8 works best for *Sentiment*, whereas a smaller batch (4) is preferable for *Emotion*. This mirrors the common trade-off that smaller batches can aid optimization on noisier, fine-grained tasks (Figure 3b).

Learning rate. A moderate step size is consistently favorable: 2×10^{-4} is optimal on both datasets, while too small harms performance (Figure 3b).

Training epochs. With other settings fixed, *Sentiment* peaks at 4 epochs and degrades with fewer or more updates, suggesting mild overfitting beyond



(a) LoRA Hyperparameter Ablation Study



(b) Training Hyperparameter Ablation Study

Figure 3: Hyperparameter ablation studies on Gemma3-1B. The darker bars represent the *Sentiment* dataset and lighter bars represent the *Emotion* dataset.

the optimum. *Emotion* benefits from a slightly longer schedule, with best results at 6 epochs. Tuning the stopping point remains important even for lightweight adapters (Figure 3b).

Across tasks, (i) moderate LoRA capacity ($r \approx 8$-16) with higher scaling ($\alpha \approx 64$) is effective; (ii) adapting both *Attention* and *MLP* is a strong default, though *MLP*-only can win on *Emotion*; and (iii) a mid-range training setup (batch 4-8, LR 2×10^{-4}, 4–6 epochs) consistently yields the best trade-off between stability and accuracy.

6.4 Cross-Dataset Evaluation

To evaluate the generalization ability of LP-FT-LoRA , we conduct cross-dataset experiments by training models on one dataset and directly testing, in a zero-shot manner, on another dataset from the same domain. We compare the performance of our proposed LP-FT-LoRA against Standard-LoRA fine-tuning across sentiment, emotion, and sarcasm domains using Gemma3-1B and Qwen3-0.6B backbones.

Sentiment. Transferring between the *Sentiment Classification* dataset and *SentNoB* demonstrates clear advantages for LP-FT-LoRA . When trained on Sentiment Classification and tested on SentNoB, LP-FT-LoRA achieves 55.55% (Gemma3-1B) and 51.70% (Qwen3-0.6B), surpassing Standard-LoRA by 1.33 and 2.20 percentage points, respectively. Conversely, when trained on *SentNoB* and tested on *Sentiment Classification*, LP-FT-LoRA yields stronger gains, improving by 4.37 points on Gemma3-1B and 1.68 points on Qwen3-0.6B.

Emotion. We examine transfer between *Emotion Detection* and *EmoNoBa*. Both models achieve nearly identical performance here, with Standard-LoRA slightly outperforming LP-FT-LoRA (34.98% vs. 34.67% for Gemma3-1B, and 35.87% vs. 35.85% for Qwen3-0.6B). This indicates that transferring across fine-grained emotion datasets remains a considerable challenge, and the incremental gain from progressive training is less pronounced compared to other domains.

Sarcasm. Cross-dataset transfer between the *Sarcasm Detection* dataset and *BanglaSarc* shows consistent improvements with LP-FT-LoRA . On Gemma3-1B, LP-FT-LoRA achieves 65.10% compared to 62.81% with Standard-LoRA, while on Qwen3-0.6B, it attains 52.27% versus 50.78%. These results highlight the effectiveness of LP-FT-LoRA in capturing transferable features for sarcasm recognition, which often relies on subtle pragmatic cues.

Domain	Train Dataset	Test Dataset	Gemma3-1B		Qwen3-0.6B	
			LP-FT-LoRA	Std-LoRA	LP-FT-LoRA	Std-LoRA
Sentiment	Sentiment	SentNoB	**55.55%**	54.22%	**51.70%**	49.50%
	SentNoB	Sentiment	**60.94%**	56.57%	**51.29%**	49.61%
Emotion	Emotion	EmoNoBa	34.67%	**34.98%**	35.85%	**35.87%**
Sarcasm	Sarcasm	Bangla Sarc	**65.10%**	62.81%	**52.27%**	50.78%

Table 3: Cross-dataset evaluation (accuracy). Models are trained on *Dataset 1* and evaluated on *Dataset 2*. Best score is in **bold**.

6.5 Computation and Training Time

Model	Std LoRA	LP-FT LoRA		
		S1	S2	S3
Qwen3-0.6B	2282	968	2181	2109
Gemma3-1B	2046	769	1861	1963
Qwen2.5-1.5B	4988	2403	4779	4767
SmolLM2-0.3B	2324	942	2144	2136

Table 4: Comparison of the average training time per epoch (in seconds) between Standard LoRA and LP-FT-LoRA across the previously mentioned dataset for various model architectures. Here S1 means Stage 1 and so on

The table 4 presents the average per-epoch training time (in seconds) for Standard LoRA and LP-FT-LoRA across three. The LP-FT LoRA method shows notable reductions in training time during Stage 1 (S1), where models such as Qwen3-0.6B and Gemma3-1B decrease from 2282 s to 968 s and from 2046 s to 769 s, respectively. Similar trends appear for SmolLM2-0.3B (2324 s $\rightarrow$ 942 s) and Qwen2.5-1.5B (4988 s $\rightarrow$ 2403 s). In later stages (S2 and S3), the training time approaches the Standard LoRA baseline, reflecting the increasing proportion of parameters being updated.

7 Conclusion

In this work, we introduced LP-FT-LoRA , a novel three-stage fine-tuning framework that integrates linear probing and LoRA to improve the adaptation of pre-trained language models for specialized classification tasks. Our approach methodically decouples classifier head alignment from adapter representation learning by first training the classifier on frozen features, then training the LoRA adapters while freezing the LM backbone and classifier, and finally jointly refining both components. This structured process is designed to mitigate the noisy gradients and slow convergence associated with standard end-to-end fine-tuning.

Our extensive experiments across four language models and five Bangla NLP datasets demonstrated that LP-FT-LoRA consistently outperforms standard LoRA fine-tuning and other strong baselines. While this study confirms the effectiveness of our method on Bangla classification tasks, future work could extend the framework to other languages, larger models, and different task formats, such as text generation. Further research could also explore dynamic stage transitions or integrate more advanced PEFT techniques to build upon these findings.

Limitations

This study has several limitations that should be acknowledged. The evaluation of the proposed LP-FT-LoRA framework is primarily conducted on medium-scale transformer models up to 1.5 billion parameters. Consequently, its effectiveness and scalability on larger models remain unverified, and different optimization strategies may be required for such settings.

The method has been tested exclusively on classification tasks within the Bangla language domain. While the results demonstrate strong performance gains, the generalizability of the approach to other languages or to different NLP tasks such as text generation has yet to be established. The multi-stage training process involves several hyperparameters and requires careful tuning specific to each dataset. This tuning complexity may limit out-of-the-box applicability and could introduce additional overhead in practical deployments.

Recent studies on transliteration and code-mixed datasets for Bangla are gaining increased attention (Fahim et al., 2024; Haider et al., 2024; Ahmed et al., 2024). It would be valuable to investigate how our model performs on these alternative text forms. Instead of focusing solely on standard Bangla datasets, future work could explore the model's effectiveness on transliterated and code-mixed Bangla data.

References

Fahim Ahmed, Md Fahim, Md Ashraful Amin, Amin Ahsan Ali, and AKM Rahman. 2024. Improving the performance of transformer-based models over classical baselines in multiple transliterated languages. In *ECAI 2024*, pages 4043–4050. IOS Press.

Tasnim Sakib Apon, Ramisa Anan, Elizabeth Antora Modhu, Arjun Suter, Ifrit Jamal Sneha, and MD Golam Rabiul Alam. 2022. Banglasarc: A dataset for sarcasm detection. *arXiv preprint arXiv:2209.13461*.

Harsh Bihany, Shubham Patel, and Ashutosh Modi. 2025. Lorma: Low-rank multiplicative adaptation for llms. In *Findings of the Association for Computational Linguistics: ACL 2025*.

Yidong Chai, Yang Liu, Yonghang Zhou, Jiaheng Xie, and Daniel Dajun Zeng. 2025. A bayesian hybrid parameter-efficient fine-tuning method for large language models. *arXiv preprint arXiv:2508.02711*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235.

Md Fahim, Fariha Tanjim Shifat, Fabiha Haider, Deeparghya Dutta Barua, MD Sakib Ul Rahman Sourove, Md Farhan Ishmam, and Md Farhad Alam 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.

Jiancheng Gu, Jiabin Yuan, Jiyuan Cai, Xianfa Zhou, and Lili Fan. 2025. La-lora: Parameter-efficient fine-tuning with layer-wise adaptive low-rank adaptation. *Neural Networks*, 194:108095.

Changhao Guan, Chao Huang, Hongliang Li, You Li, Ning Cheng, Zihe Liu, Yufeng Chen, Jinan Xu, and Jian Liu. 2025. Multi-stage llm fine-tuning with a continual learning setting. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5484–5498.

Fabiha Haider, Fariha Tanjim Shifat, Md Farhan Ishmam, Deeparghya Dutta Barua, Md Sakib Ul Rahman Sourove, Md Fahim, and Md Farhad Alam. 2024. Banth: A multi-label hate speech detection dataset for transliterated bangla. *arXiv preprint arXiv:2410.13281*.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2024. Lora+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*.

Md Zobaer Hossain, Md Ashraful Rahman, Md Saiful Islam, and Sudipta Kar. 2020. BanFakeNews: A dataset for detecting fake news in Bangla. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2862–2871, Marseille, France. European Language Resources Association.

Guiyang Hou, Yongliang Shen, and Weiming Lu. 2024. Progressive tuning: Towards generic sentiment abilities for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14545–14558.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Nafis Irtiza Tripto and Mohammed Eunus Ali. 2018. Detecting multilabel sentiment and emotions from bangla youtube comments. In *2018 International Conference on Bangla Speech and Language Processing (ICBSLP)*, pages 1–6.

Khondoker Ittehadul Islam, Sudipta Kar, Md Saiful Islam, and Mohammad Ruhul Amin. 2021. Sentnob: A dataset for analysing sentiment on noisy bangla texts. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3265–3271.

Khondoker Ittehadul Islam, Tanvir Hossain Yuvraz, Md Saiful Islam, and Enamul Hassan. 2022. EmoNoBa: A dataset for analyzing fine-grained emotions on noisy Bangla texts. 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 2: Short Papers)*, pages 128–134, Online only. Association for Computational Linguistics.

Minsoo Kim, Sihwa Lee, Wonyong Sung, and Jungwook Choi. 2024. Ra-lora: Rank-adaptive parameter-efficient fine-tuning for accurate 2-bit quantized large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15773–15786.

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*.

Shiwei Li, Xiandi Luo, Xing Tang, Haozhao Wang, Hao Chen, Weihong Luo, Yuhua Li, Xiuqiang He,

and Ruixuan Li. 2025. Beyond zero initialization: Investigating the impact of non-zero initialization on lora fine-tuning dynamics. *arXiv preprint arXiv:2505.23194*.

Yulong Mao, Kaiyu Huang, Changhao Guan, Ganglin Bao, Fengran Mo, and Jinan Xu. 2024. Dora: Enhancing parameter-efficient fine-tuning with dynamic rank distribution. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11662–11675.

Jathan Nguyen et al. 2025. Probing and steering evaluation awareness of language models. *arXiv preprint arXiv:2507.01786*.

Lin Ning, Harsh Lara, Meiqi Guo, and Abhinav Rastogi. 2025. Mode: Effective multi-task parameter efficient fine-tuning with a mixture of dyadic experts. In *Findings of the Association for Computational Linguistics: NAACL 2025*.

Sanjay Rajput and Priya Mehta. 2025. On the instability of jointly fine-tuning adapters and classification heads in large language models. *Journal of Machine Learning Research*, 26(45):1–25.

Vimal Thilak, Chen Huang, Omid Saremi, Laurent Dinh, Hanlin Goh, Preetum Nakkiran, Joshua M. Susskind, and Etai Littwin. 2024. Lidar: Sensing linear probing performance in joint embedding ssl architectures. In *International Conference on Learning Representations*.

Akiyoshi Tomihari and Issei Sato. 2024. Understanding linear probing then fine-tuning language models from NTK perspective. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wei Zhao, Li Chen, and Jin-seo Park. 2024. The new era of foundation models: A survey on pre-training, fine-tuning, and adaptation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 1–18.