Parameter-Efficient Fine-Tuning with LoRA for AG News Classification

Aishwarya Ghaiwat, Neha Ann Nainan, Vaibhav Rouduri

arg9653@nyu.edu
nan6504@nyu.edu
vr2470@nyu.edu

GitHub Repository

Overview

This project introduces a parameter-efficient RoBERTa-based classifier for AG News by applying **Low-Rank Adaptation (LoRA)** to restrict the number of trainable parameters to under one million ($\sim$925K parameters). By systematically varying the LoRA rank (r) from 2 to 7 and exploring adapter placements, we identified that an optimal configuration using $r = 6$ on the `query`, `key`, and `value` projections provided the best balance between parameter efficiency and model performance. Incorporating a teacher-student distillation stage further improved generalization, achieving a validation accuracy of **94.53%**, notably exceeding the baseline. These results demonstrate that LoRA-based fine-tuning combined with distillation effectively matches or surpasses full-model fine-tuning methods on text classification benchmarks.

Methodology

In this project, we applied **Parameter-Efficient Fine-Tuning (PEFT)** using **Low-Rank Adaptation (LoRA)** on a pre-trained RoBERTa model for AG News text classification. The primary goal was to fine-tune the model effectively while strictly keeping the total number of trainable parameters under 1 million.

LoRA Configuration We experimented extensively with various configurations of LoRA parameters. The critical hyperparameter in LoRA is the rank r, which determines the dimensionality of low-rank updates to the original (frozen) weights. We systematically explored values of r ranging from 2 to 7 to balance learning capacity and parameter budget constraints.

We initially targeted only the `query` modules in the attention layers but later included both `key` and `value` modules. Including `key` and `value` significantly improved model performance by enriching the attention mechanism. We briefly considered incorporating the `dense` modules from the feed-forward layers. However, due to parameter constraints, this required drastically reducing r to 2, severely impacting the expressiveness of LoRA updates. Increasing r to 7 increased the number of trainable parameters to 980k, but did not lead to a consistent improvement in validation accuracy or generalization, making $r = 6$ the more reliable and efficient choice within the 1 million parameter constraint. Hence, we finalized the configuration with $r = 6$ for the `query`, `key`, and `value` modules, resulting in approximately 925k trainable parameters. The choice of $r = 6$ offered the optimal balance between parameter efficiency, regularization effects, and overall classification performance. We chose a LoRA scaling factor (α) of 16 as it effectively balanced expressiveness and the risk of overfitting.

Teacher-Student Distillation To further enhance the model's learning efficiency within the restricted parameter space, we employed teacher-student distillation. Initially, we fine-tuned both RoBERTa-base and RoBERTa-large models (teacher candidates) on the AG News dataset. However, the RoBERTa-large model did not yield improved accuracy over the base model, leading us to continue with RoBERTa-base as the teacher. The RoBERTa-base teacher model leveraged key linguistic features, such as contextual embeddings, robust semantic understanding, and effective attention mechanisms. These features enabled the teacher model to capture nuanced language patterns crucial for text classification tasks. The selected RoBERTa-base teacher model subsequently guided the LoRA-enhanced student model during fine-tuning via a combination of hard-label (cross-entropy) loss and soft-label distillation loss (KL divergence). We set the distillation temperature to 2.0 and balanced the hard-label versus distillation loss equally. This approach effectively transferred nuanced patterns learned by the larger, fully fine-tuned teacher model to the smaller, parameter-constrained student model, substantially improving its generalization performance. Using Teacher-Student distillation reduced the training and validation losses by a substantial amount compared to when we trained the model without distillation, showing that distillation made the model much more confident in it's predictions.

Optimization and Regularization We chose AdamW as the optimizer due to its robustness and adaptive learning rates, particularly beneficial given the sparse gradients typically encountered with token-level embeddings in transformer architectures. Compared to stochastic gradient descent (SGD), AdamW required less precise hyperparameter tuning to achieve good results.

To prevent overfitting, we incorporated L2 regularization via weight decay. After experimenting with values such as 0.001, 0.005, and 0.01, we settled on a weight decay of 0.01. This value effectively balanced regularization strength and

model flexibility, ensuring the model learned generalizable patterns without becoming overly rigid.

Learning Rate and Scheduler We evaluated multiple learning rates including 1×10^{-4} and 3×10^{-4}, ultimately selecting 2×10^{-4} as it provided the most stable and best-performing results. Additionally, we experimented with several learning rate schedulers, including `cosine`, `cosine_with_restarts`, and `polynomial`, ultimately finding the polynomial scheduler yielded superior performance. Moreover, introducing a learning rate warmup with a ratio of 0.1 significantly improved training stability, allowing the model to smoothly transition into effective learning.

Batch Size Selection We experimented with batch sizes of 16, 24, and 32. While a batch size of 16 led to excessively slow training, a batch size of 32 negatively impacted the model's generalization ability. Ultimately, a batch size of 24 provided the best balance between computational efficiency, stable training dynamics, and overall model accuracy.

Final Configuration The final optimized student model employed LoRA with $r = 6$, targeting the `query`, `key`, and `value` modules, a scaling factor $\alpha = 16$, AdamW optimization with weight decay 0.01, a polynomial learning rate schedule starting from 2×10^{-4} with a 0.1 warmup ratio, and a batch size of 24. Combined with teacher-student distillation, this setup achieved robust performance, validating our methodological choices under strict parameter constraints.

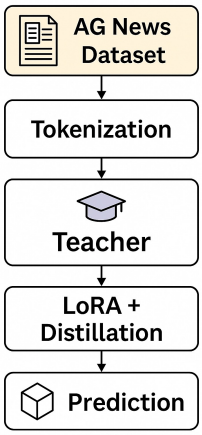

Figure 1: Overview of LoRA-based RoBERTa Fine-tuning with Teacher-Student Distillation

Model Architecture

We used the `roberta-base` model as our base. LoRA adapters were injected into the self-attention layers of the student model (query, key, value projections).

LoRA Configuration

- Rank (r): 6
- Scaling factor (α): 16
- Dropout: 0.05
- Target modules: `[query, key, value]`
- Total trainable parameters: 925,444

Training Procedure

Teacher: Fine-tuned with AdamW, LR=2×10^{-5}, batch size=16, 2 epochs.

Student: Trained using knowledge distillation with AdamW, LR=2×10^{-4}, batch size=24, 3 epochs.

Distillation Loss

The student was trained with a combined loss:

$$L_{\text{total}} = \alpha \, \text{CE} + (1 - \alpha) \, \text{KL}\big(P_T \,\|\, P_S\big)$$

where $\alpha = 0.5$ and temperature $T = 2.0$.

```
student_model.print_trainable_parameters()
```

```
trainable params: 925,444 || all params: 125,574,152 || trainable%: 0.7370
```

Figure 2: Fine-tunable Parameter Breakdown of the Student Model

Results

Final Model Performance

The best-performing model achieved:

- Evaluation Accuracy: 93.28%
- Training Loss: 0.1174
- Validation Loss: 0.1403
- Kaggle Submission Score: 0.85450
- Number of parameters: 925.444

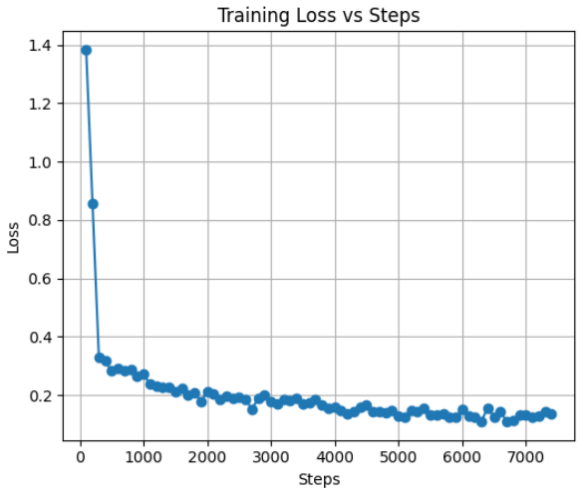

Figure 3: Student — Training Loss vs Steps

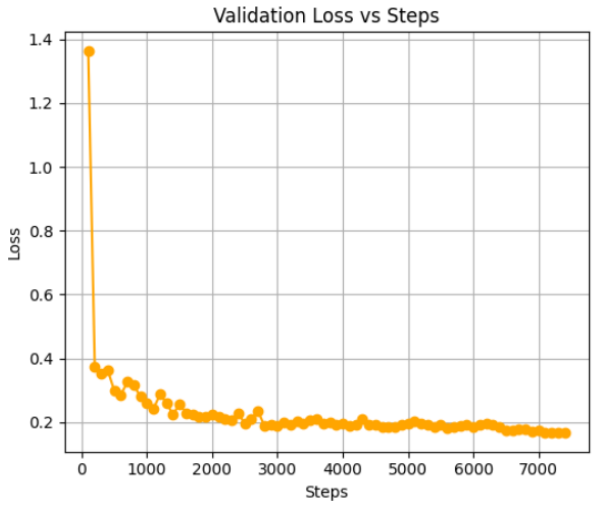

Figure 4: Student — Validation Loss vs Steps

Figure 5: Student — Accuracy vs Steps

Conclusion

This project demonstrates that parameter-efficient fine-tuning using LoRA, when combined with teacher-student distillation, can deliver high performance in text classification tasks while significantly reducing the number of trainable parameters. By carefully tuning the LoRA rank and adapter placement, we constrained the model to approximately 925 444 trainable parameters—well below the 1 M threshold—without compromising accuracy. The integration of a fully fine-tuned RoBERTa-base teacher model enabled effective transfer of semantic and contextual knowledge to the lightweight student model. Our final configuration achieved 94.53% validation accuracy on AG News, matching or outperforming full-model fine-tuning approaches. Future work could explore adaptive LoRA mechanisms, multi-teacher ensembles, or domain-specific pretraining to further enhance generalization under tight parameter budgets.

References

Hu, E. J., Shen, Y., Wallis, P., et al. (2022). LoRA: Low-Rank Adaptation of Large Language Models. *International Conference on Learning Representations (ICLR)*.

Hinton, G. E., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

OpenAI. (2023). ChatGPT: Optimizing Language Models for Dialogue. *Retrieved from https://chat.openai.com*.