Cornell Bowers C·IS
College of Computing and Information Science

LoRA: Low-Rank Adaptation of Large Language Models

Alicia Chu, Rayhan Khanna, Emma Li, Sanjana Nandi, Tejal Nair
Cornell University

Introduction

Why Parameter-Efficient Tuning?

→ Traditional fine-tuning **doesn't scale** as models grow.
→ Re-training & saving a 500 MB RoBERTa-base checkpoint per task **balloons storage** (10 tasks → 5 GB) and **multiplies GPU time**.

LoRA's Solution (Hu et al. 2022):

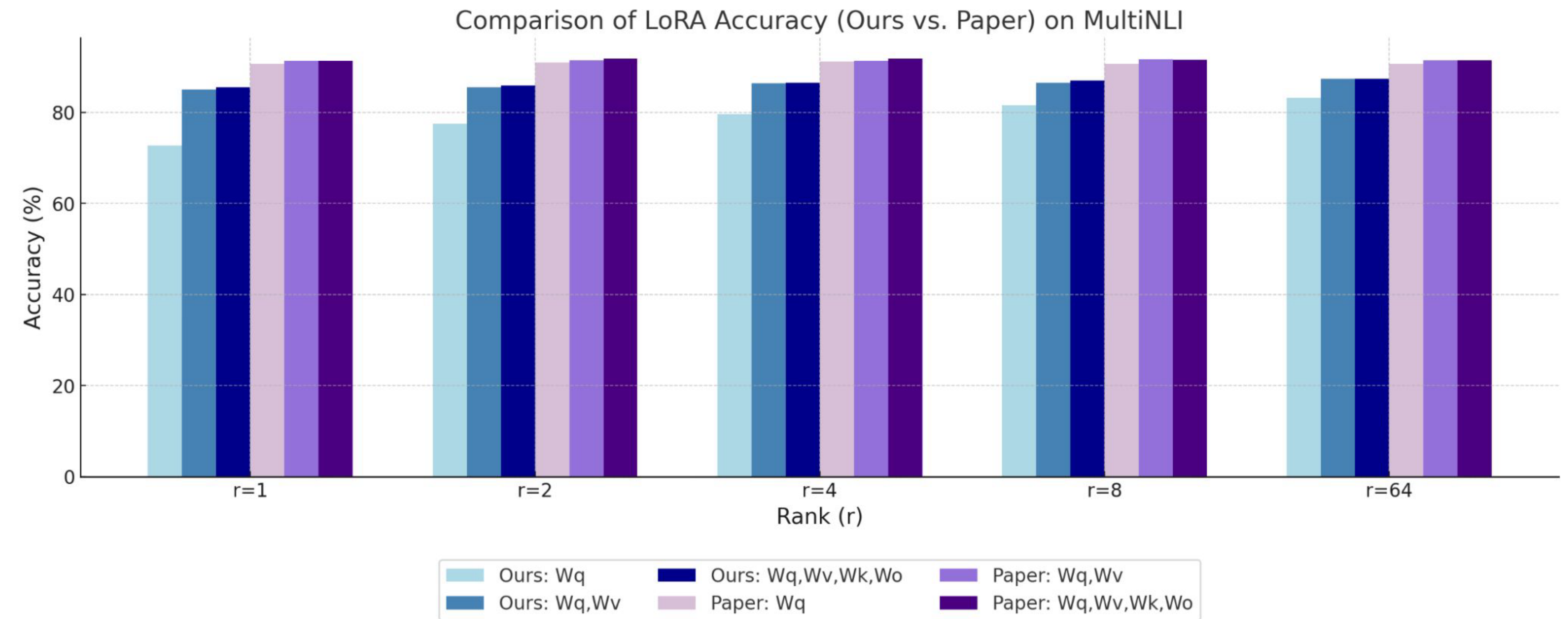
(1) **Freeze** pretrained weights W_0.
(2) **Inject** rank-r (r << d) adapters A, B so ΔW = BA.
(3) **Train & save** only {A, B} (< 1% of total params) per task.

Full Fine-Tuning	**LoRA**
• 100% of parameters updated • Memory usage and compute required ⬆ • Duplicates full models	• Only small low-rank matrices trained + saved • Memory usage and compute required ⬇ • Stores only tiny adapters

Project Goals:

→ **Re-implement LoRA** and **validate GLUE performance**.
→ Explore performance when **varying decomposition rank r** while applied to **other matrices and MLP layers**.

Results



Task	LoRA Accuracy (Paper)	Our Implementation Accuracy	Difference
SST-2	95.1±0.2	93.8±0.2	−1.3
QQP	90.8±0.1	88.7±0.2	−2.1
MRPC	89.7±0.7	88.0	−1.7

Table 1: Comparison of LoRA paper results and our implementation on GLUE tasks.

Table 2: Hyperparameters used for training

Task	r	alpha	epochs	Batch size	Learning rate	Max seq length	Warm up ratio	Target weights
SST-2	8	16	3	16	5e-4	128	0.06	Query + Value
QQP	8	16	3	16	5e-4	128	0.06	Query + Value
MRPC	8	16	3	16	4e-4	128	0.06	Query + Value

Applications / Related Works

→ **Deploying on resource-constrained devices** where fine-tuning is impractical due to resource constraints
→ ALoRA (Wang et al., 2024) - **dynamic rank adaptation** (rank adjusted during training based on layer-wise importance scores)
→ RA-LoRA (Xu et al., 2024) - **rank-adaptive low-rank adaptation** for quantized language models

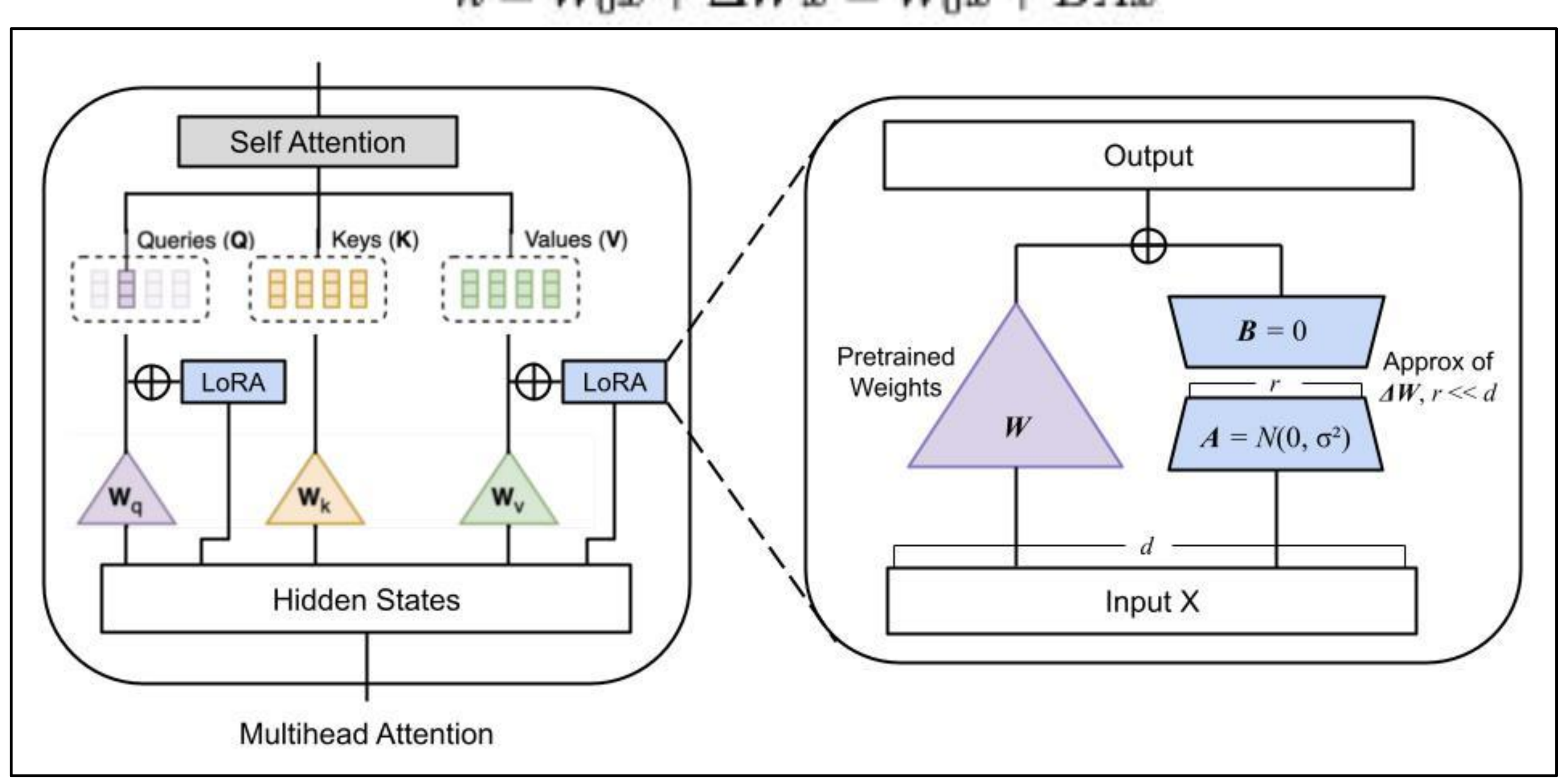
Conclusions

→ **Efficient Fine-tuning:** LoRA enables high model performance while updating only a tiny fraction of parameters.
→ **Parameter Savings:** Achieved strong results on GLUE tasks with ~0.3M trainable parameters vs 125M+ in full fine-tuning.
→ **Challenges Faced:** Lower training epochs (3 vs. 60) and shorter max sequence length (128 vs. 512) slightly impacted accuracy.
→ **Key Takeaways:** LoRA shows robustness and scalability for resource-constrained model adaptation.

Methodology

LoRA Forward Pass

$$h = W_0 x + \Delta W x = W_0 x + BAx$$



Model

→ **RoBERTa-base encoder** + **LoRA retrofitting**
→ LoRA applied to **query** and **value projection matrices** (attention layers)

Datasets

→ **GLUE benchmark tasks:** SST-2, MRPC, QQP, MNLI
→ Loaded using **HuggingFace 'datasets' library.**

Modifications

→ Trained **only for 3 epochs** (paper used 25-60).
→ **Max input sequence length = 128 tokens** (vs. 512 in paper)
→ Trained on RoBERTa-base rather than GPT3 for varying weight types and r

Tools

→ **HuggingFace Transformers** + **Pytorch**
→ Custom **patch_model_with_lora()** function to inject LoRA modules into transformer architecture

References

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2022). LoRA: Low-rank adaptation of large language models. *ICLR*, *1*(2), 3.

Wang, T., Chen, X., Liu, S., Lin, Y., Liu, L., & Li, Z. (2024). ALoRA: Allocating low-rank adaptation for fine-tuning large language models. arXiv:2403.16187.

Xu, R., Zhou, Z., Zhao, Y., Yu, Z., Liu, Z., Xu, C., & Lin, D. (2024). RA-LoRA: Rank-adaptive parameter-efficient fine-tuning for accurate 2-bit quantized large language models. In Findings of ACL 2024.