Reproducing *LoRA: Low-Rank Adaptation of Large Language Models*

Team Members: Deshaun Jones, Brett Warren, Anthony Litwin

Course: CS722 – Machine Learning

Professor Nazzal

Proposal Date: October 7, 2025

Base Paper and Main Limitation

The foundational paper for this project is *"LoRA: Low-Rank Adaptation of Large Language Models"* by Hu et al. (2021). LoRA proposes a method for efficiently fine-tuning large pretrained language models by updating only a subset of parameters using low-rank adaptation matrices, rather than modifying the full weight matrices. This approach reduces memory and computational requirements while maintaining or improving performance.

A key limitation noted by the authors concerns multi-task scenarios. Merging LoRA adjustments into the main model for faster inference prevents processing inputs from multiple tasks in a single batch. Keeping the adjustments separate preserves task flexibility but increases inference time due to the dynamic application of the updates. Additionally, because LoRA freezes the pretrained model, its effectiveness is limited by the quality of the base model and cannot compensate if the underlying model is suboptimal.

In summary, LoRA is memory- and compute-efficient, but handling multiple tasks or relying on suboptimal base models introduces trade-offs in flexibility and performance.

Proposed Work

Our project aims to replicate a subset of the LoRA findings, focusing on tasks and model sizes feasible with our available resources. Our minimum viable product will:

- Apply LoRA to RoBERTa-base for classification tasks and GPT-2 small/medium for text generation tasks.
- Reproduce selected experiments from the original paper, including ablations on the rank of the adaptation matrices and subsets of weights adapted.

- Compare LoRA performance against baselines such as full fine-tuning, partial fine-tuning, and other parameter-efficient methods (e.g., adapters, bias-only tuning).

This effort will validate LoRA's effectiveness while ensuring experiments are achievable within our computational constraints.

Planned Experiments

Datasets

The LoRA paper evaluates multiple NLP tasks: GLUE for classification, SAMSum for summarization, E2E NLG Challenge for generation and WikiSQL for text-to-SQL. We will focus on a feasible subset:

- Classification: Selected GLUE tasks (e.g., MNLI, MRPC)
- Generation: One summarization dataset (E2E NLG Challenge)

We will use standard training, validation, and test splits, following the original preprocessing and tokenization to ensure comparability.

Baseline Approach

- Models: RoBERTa-base (classification), GPT-2 small/medium (generation)
- Method: Pretrained weights frozen; low-rank adaptation matrices injected into selected layers.
- Baselines: Full fine-tuning, partial fine-tuning, and alternative parameter-efficient strategies (adapters, bias-only tuning)
- Ablations: Rank variation, choice of adapted weights, and hyperparameter sensitivity will be explored.

Evaluation Plan

- Performance Metrics: GLUE, BLEU, NIST, MET, ROUGE-L and CIDEr
- Efficiency Metrics: Memory usage, trainable parameter count, training throughput
- Robustness: Multiple random seeds to estimate variance
- Analysis: Quantitative and qualitative comparison to the original study, documenting any deviations or discrepancies.

Preliminary Project Timeline

Week	Task
1–2	Literature review, code familiarization, dataset/model selection
3–4	Implement LoRA on selected models and establish baselines
5–6	Initial experiments on GLUE and E2E NLG Challenge; verify reproducibility
7–8	Ablation studies: rank, weight selection, hyperparameter tuning
9	Evaluate results, analyze efficiency metrics, compare to original study
10	Finalize report and prepare presentation