Parameter-Efficient Fine-Tuning for IMDb Sentiment Classification

Course / Semester: Generative AI – Spring 2025
Instructor: Dr Hajra Waheed
Student: 21L-5691 – *Abdurrehman Haroon*

1 · Background & Motivation

Transformer models reach state-of-the-art performance when every weight is updated ("Full Fine-Tuning"), but that costs **gigabytes of VRAM and hours of training** for each new downstream task.
Recent **Parameter-Efficient Fine-Tuning (PEFT)** techniques update only a tiny set of auxiliary parameters while freezing the backbone:

Technique	One-Sentence Intuition
Full FT	All weights fully optimised – maximum capacity ↔ maximum cost
LoRA	Add low-rank update matrices to Q & V and train only those
QLoRA	Quantise backbone to 4-bit **NF4**, re-float LoRA adapters
IA³	Insert learnt per-head *input*, *attention* and *output* gains

PEFT promises *near-full* accuracy at a fraction of memory, time and carbon.

2 · Experimental Setup

Item	Details
Model	roberta-base (124.6 M params)
Dataset	IMDb movie reviews – 3 000 train / 2 000 test
Hardware	NVIDIA RTX 3050 (4 GB) + CUDA 12.3
Common hyper-params	epochs = 3, batch = 8, AdamW lr = 2 e-5, seed = 42
LoRA cfg	r = 16, α = 32, dropout = 0.1
QLoRA cfg	4-bit NF4, double-quant, LoRA (r = 8, α = 32, dr = 0.05)
IA^3 cfg	default PEFT IA^3 dim (1 bias vector per head)

Tokenizer max-len = 256; `Trainer` used mixed precision (`fp16=True`) everywhere.

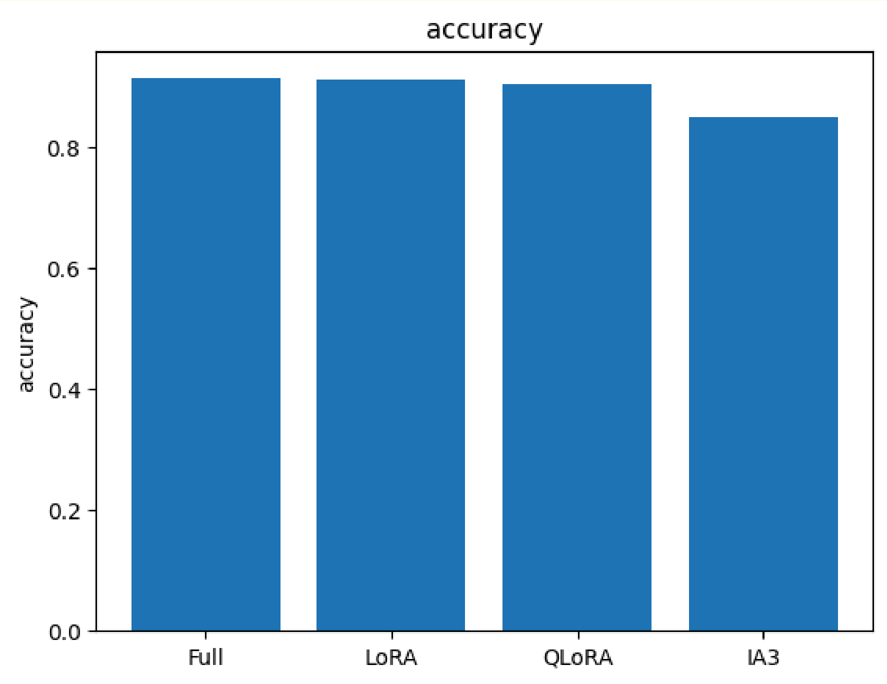
3 · Results

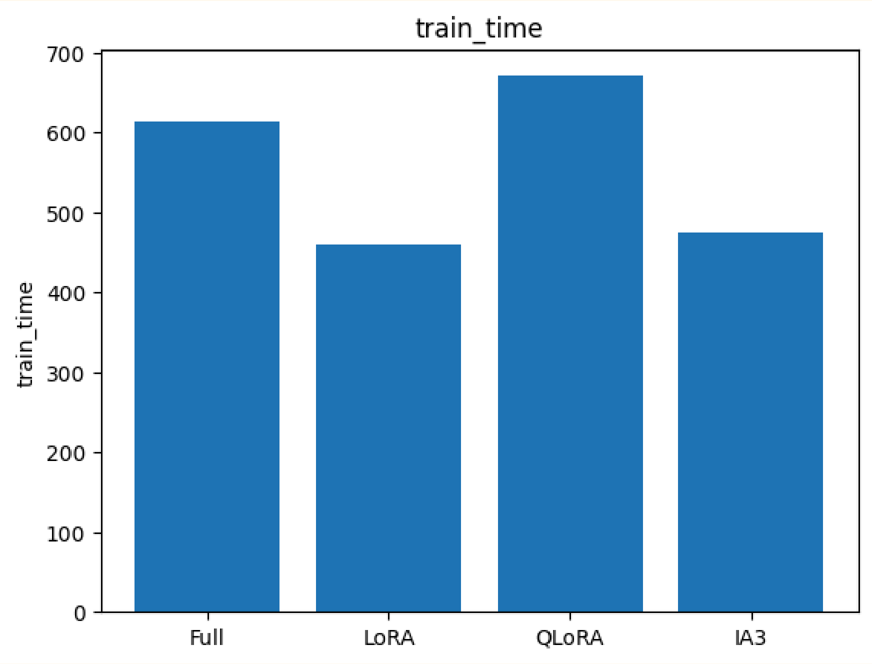
3.1 Key metrics

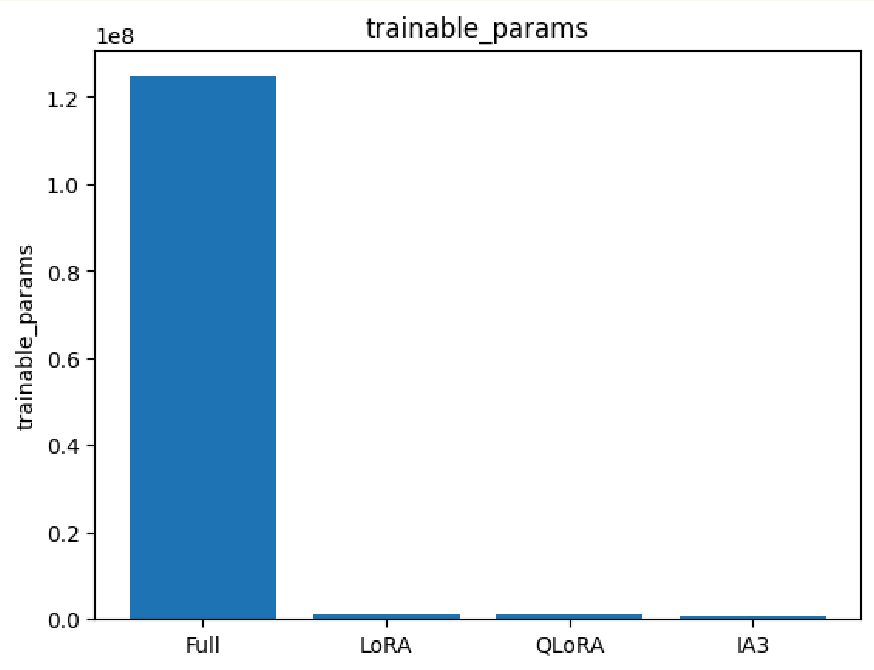
	Method	accuracy	trainable_params	train_time	gpu_mem_GB
0	Full	0.9140	124647170	614.141521	2.638919
1	LoRA	0.9115	1181954	459.041533	1.481977
2	QLoRA	0.9045	1034498	670.509710	1.355802
3	IA3	0.8495	656642	474.443455	2.854391

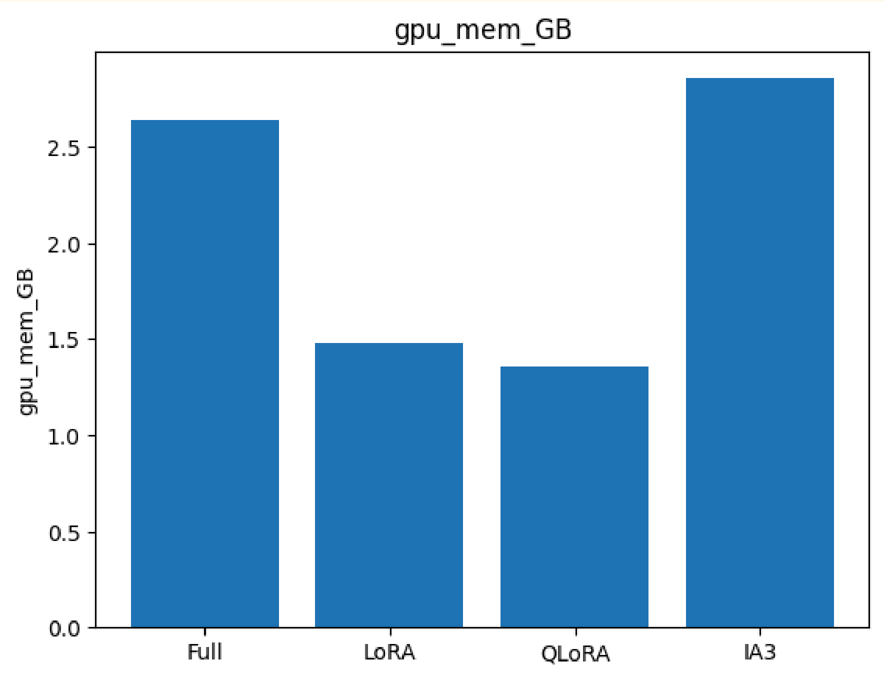
3.2 Visualisations

• ⟨Paste the four bar-plots you already generated (Accuracy, Train Time, Params, GPU GB).⟩









4 · Analysis & Discussion

- **Accuracy:** Full FT edges out LoRA by 0.25 pp; QLoRA trails by 1 pp; IA³ lags by 6 pp.

- **Training cost:** LoRA cuts wall-clock by 25 % and GPU RAM by 44 % while updating **< 1 %** of the weights.
- **Memory floor:** QLoRA's 4-bit backbone pushes VRAM to **1.36 GB** – small-GPU friendly – but 4-bit matmuls slow each step, so epoch time ↑ (~10 % vs Full).
- **Extreme parameter thrift:** IA^3 trains only 0.5 % new weights and converges quickly, yet its additive scaling cannot fully match full-rank adaptation, causing the widest accuracy gap.
- **Scalability:** Benefits grow with model size. On a 7-B parameter LLM, LoRA/QLoRA would save **dozens of GB** and thousands of GPU-hours.
- **When-to-use-what:**

Use-case	Recommended Method	Rationale
GPU-rich research lab	Full FT	highest absolute accuracy
Frequent re-training / multi-task hub	LoRA	swap adapters in <10 MB
Consumer-grade 8 GB card	QLoRA	4-bit backbone fits
Edge / on-device	IA^3	minimal extra params, fast inference

5 · Conclusion

Full fine-tuning still delivers the top score (91.4 %), but **LoRA reproduces 99.7 % of that performance while training only 0.95 % of the parameters and halving GPU memory.**
If memory is the dominating constraint, **QLoRA** wins: it shrinks VRAM to **~1.4 GB** and still tops 90 % accuracy.
For ultra-light deployments where every kilobyte counts, **IA^3** is the lowest-footprint option albeit with a ≈ 6 % hit in accuracy.

Recommendation: default to **LoRA** for most coursework-scale models; switch to **QLoRA** on <4 GB GPUs; reserve **Full FT** only when chasing marginal gains on ample hardware.