Parameter Efficiency or Efficient Full Fine-Tuning?

PEFT *vs.* GaLore

Malek Camilo[1] · **Juan Manuel Fernández**[2,3] · **Marcelo Errecalde**[4]

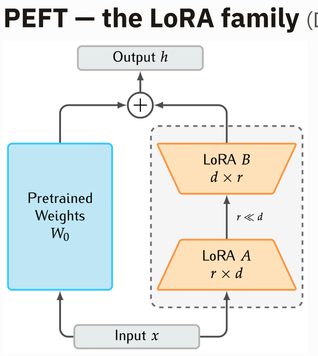
[1] La Plata National University (UNLP) · [2] LICDIA, National University of Luján (UNLu) · [3] Buenos Aires Institute of Technology (ITBA) · [4] LIDIC, National University of San Luis (UNSL) — Argentina

mcamilo@unlu.edu.ar · jmfernandez@{unlu, itba}.edu.ar · merreca@unsl.edu.ar

01 What are we comparing?

Full fine-tuning (FFT) of an 8B-parameter LLM updates every weight — effective, but prohibitively expensive. Two efficient philosophies compete.

PEFT — the LoRA family (DoRA · QLoRA · QDoRA)



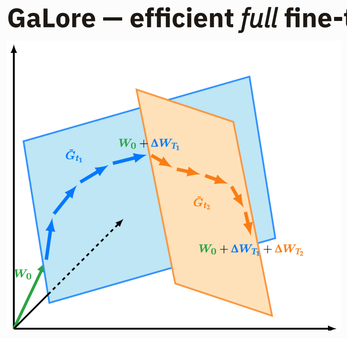
The pretrained weights **W_0 stay frozen** (blue). Each layer gains two small trainable matrices (orange), **A** (r×d) and **B** (d×r), whose product is the weight update:

```
h = W₀x + BAx   ·   rank r ≪ d
```

Only **≤ 4% of parameters train**. DoRA further decomposes updates into magnitude + direction; QLoRA / QDoRA keep the frozen base quantized at 4-bit NF4. Adapters merge into W_0 after training — no inference latency.

Figure from the thesis, adapted from Hu et al. [1].

GaLore — efficient *full* fine-tuning (+ rSVD variant)



Nothing is frozen — **100% of the weights update**. Memory is saved in the *optimizer*: moments are stored only for a rank-r projection of the gradient,

```
R = Pᵀ G Q   ·   P, Q from SVD(G)
```

Every **g** steps the subspace is recomputed, so training walks through a sequence of low-rank subspaces (blue → orange) that together approximate full-rank learning. **rSVD** swaps the exact SVD for a fast randomized one.

Figure from the thesis, adapted from Zhao et al. [4].

02 Task, corpus & setup

SEDICI — the open-access institutional repository of UNLP. We take all **19,974** Computer-Science records and classify each document's **resource type**, formulated as conditional text generation:

`title + abstract` → **`LLaMA 3.1 8B Instruct`** → `type (6 classes)`

Class imbalance is severe — test set, n = 3,000:

- Conference object — 2,343 records (78.1%)
- Article · Thesis · Other — 595 (19.8%)
- Book (34) · Learning object (28) — under 1.2% each

- **7 configurations:** FFT baseline · LoRA · DoRA · QLoRA · QDoRA (r ∈ {8…128}; attention-only vs. all linear layers) · GaLore · GaLore rSVD (r = 128, gap g).
- **Training:** 5 epochs · bf16 · 3 seeds (mean ± σ) · raw bibliographic metadata, no synthetic cleanup · HF `transformers/peft/trl` + `galore-torch`.
- **Metrics:** F1-Macro (primary — robust to imbalance) · Accuracy · Exact Match · wall-clock time · peak VRAM · trainable parameters.

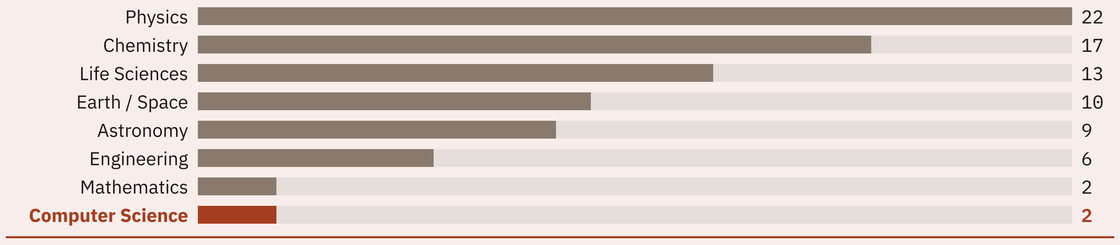
`Spanish academic corpus` `raw, imbalanced real-world data` `non-CUDA hardware`

03 National-scale compute

Run on Clementina XXI, Argentina's supercomputer

Intel Data Center GPU Max 1550 accelerators (Xe-HPC, XPU backend) — **no CUDA anywhere**. Compute awarded through **IPAC**, the competitive national call for accelerated-computing projects (**PCI category — Intensive Computing Projects, call PCI-146**), open to Argentina's entire science & technology system.

IPAC 2025 selected projects, by UNESCO area



2/81 projects selected nationwide are Computer Science — this work is one of them.

Why it matters: these results transfer to any lab adapting open LLMs on limited or non-mainstream hardware — the common case outside big tech.

04 Head-to-head results

0.657
Best F1-Macro — DoRA (r = 32)
Beats FFT (0.642) while training 1.05% of the weights.

14.1 GB
Peak VRAM — QLoRA & QDoRA
−44% memory; F1 within 1σ of full precision.

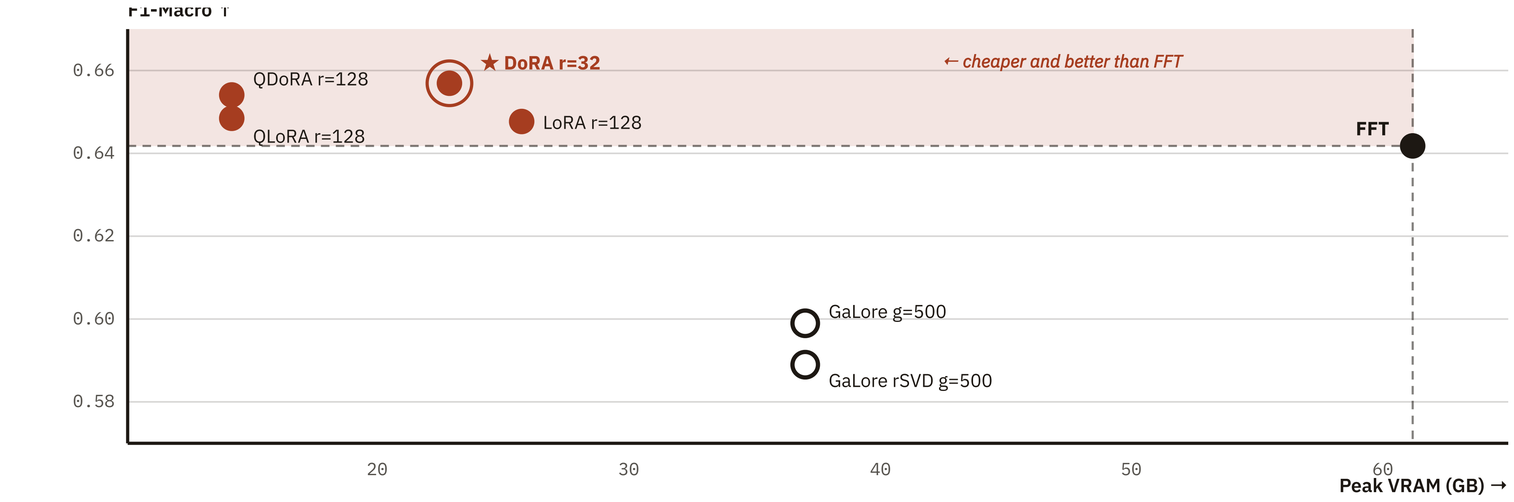
5.7 h
Fastest run — GaLore rSVD
Updates 100% of the weights — ~3× faster than DoRA.

TECHNIQUE	TRAINABLE	F1-MACRO	ACCURACY	EXACT MATCH	TIME (H)	VRAM (GB)
FFT baseline	100% 8,030M	0.6418 ±.0097	0.8490	0.8418	6.50	61.20
LoRA r=128 · all	4.01% 335.5M	0.6477 ±.0073	**0.8595**	**0.8592**	6.17	25.68
★ **DoRA** r=32 · all	**1.05%** 85.3M	**0.6570** ±.0053	0.8507	0.8436	16.59	22.86
QLoRA r=128 · all · 4-bit	4.01% 335.5M	0.6486 ±.0091	0.8576	0.8572	**5.62**	**14.14**
QDoRA r=128 · all · 4-bit	4.03% 336.9M	0.6541 ±.0058	0.8561	0.8529	9.46	**14.14**
GaLore r=128 · g=500	100% 8,030M	0.5987 ±.0056	0.8297	0.8213	25.88	37.01
GaLore rSVD r=128 · g=500	100% 8,030M	0.5889 ±.0127	0.8208	0.8120	**5.69**	37.01

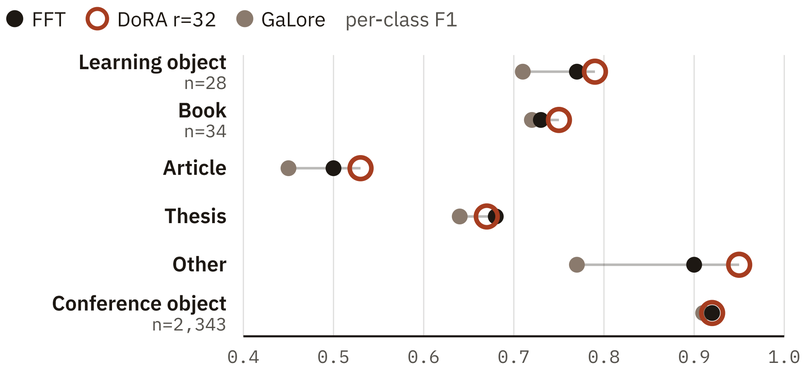
★ best F1-Macro in the study; bold = best per column. **Rank saturates:** growing LoRA from r = 32 to r = 128 quadruples trainable parameters for a gain of just +0.0004 F1-Macro.

Quality per gigabyte

F1-Macro vs. peak VRAM (GB) — the shaded region beats FFT on both axes.



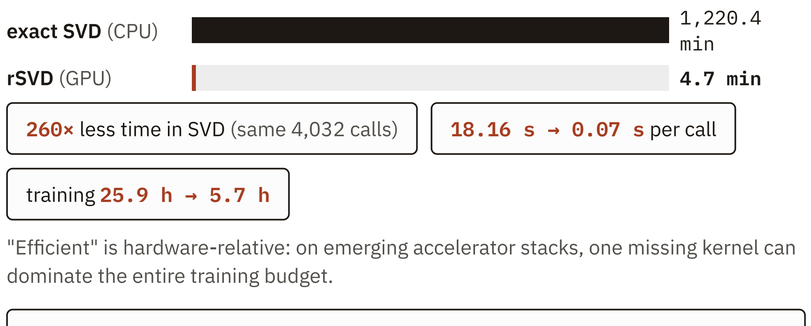
05 Where imbalance bites



FFT's gradient signal is dominated by the majority class and overwrites the subtle representations minority classes need. **Frozen-base adapters act as implicit regularizers** — DoRA recovers low-support classes best, while GaLore's instability hits them hardest.

06 The SVD bottleneck

GaLore refreshes its projections via exact SVD (`torch.linalg.svd`) — which has **no native XPU kernel**. The silent CPU fallback consumed **78.6% of GaLore's wall-clock** (20.3 of 25.9 h). Randomized SVD (`torch.svd_lowrank`) stays on the GPU:



`260×` less time in SVD (same 4,032 calls) · `18.16 s → 0.07 s` per call · training `25.9 h → 5.7 h`

"Efficient" is hardware-relative: on emerging accelerator stacks, one missing kernel can dominate the entire training budget.

What's next

- **Hybrid schemes** — combine frozen-base adapters with gradient-projection updates of selected layers.
- **Data-centric work on minority classes** — augmentation and class-aware sampling over the raw metadata.
- **Broader validation** — other open models and SEDICI fields; re-test GaLore as XPU kernels mature.

So — parameter efficiency, or efficient full fine-tuning? On real, imbalanced, Spanish-language data: parameter efficiency.

`QUALITY → PEFT`
DoRA (r=32) sets the study's best F1-Macro with 1.05% of the weights. The frozen base is an implicit regularizer: minority classes survive.

`MEMORY → 4-BIT PEFT`
QLoRA / QDoRA run in 14.1 GB — commodity-GPU territory — with F1 inside one σ of full precision.

`SPEED → GALORE rSVD`
100% of weights trained in 5.7 h — %-trainable ≠ speed. But it underfits this corpus: F1 ends ~0.05 below FFT.

`LOSS ≠ F1`
PEFT validation loss diverges +0.13–0.43 vs. FFT while F1 holds or improves. Judge classifiers on task metrics, not cross-entropy.

Everything is open — scan me

Code + data
`github.com/malekcamilo/tesis_maestria`

Fine-tuned models
`huggingface.co/malekcamilo`

Live demo
`hf.co/spaces/malekcamilo/sedici-llama-peft`

Acknowledgments

Experiments ran on the **Clementina XXI** supercomputer (SNCAD, Argentina) under the IPAC initiative, PCI category, call PCI-146. Data kindly provided by **SEDICI**, the institutional repository of UNLP. Research carried out at LICDIA (UNLu). Based on the M.Sc. thesis *Estrategias de adaptación eficiente para modelos de lenguaje abiertos* (UNLP, 2026).

Key references

1. Hu et al. *LoRA: Low-rank adaptation of large language models.* ICLR 2022.
2. Liu et al. *DoRA: Weight-decomposed low-rank adaptation.* ICML 2024.
3. Dettmers et al. *QLoRA: Efficient finetuning of quantized LLMs.* NeurIPS 2023.
4. Zhao et al. *GaLore: Memory-efficient LLM training by gradient low-rank projection.* ICML 2024.
5. Zhao, Tian et al. *GaLore 2: Large-scale LLM pre-training by gradient low-rank projection.* arXiv 2025.
6. Grattafiori et al. *The Llama 3 herd of models.* arXiv 2024.