Efficient Fine-Tuning Methods for Portuguese Question Answering: Evaluating LoRA, QLoRA, DoRA and QDoRA on BERTimbau with SQuAD

Mariela M. Nina
Universidade Federal de São Paulo (UNIFESP)
São Paulo, Brazil
mariela.nina@unifesp.br

***Abstract*—Due to the high computational costs associated with large language models (LLMs), compression techniques such as quantization and parameter-efficient fine-tuning (PEFT) methods have gained popularity for reducing resources during inference and training. This work presents the first comprehensive and exhaustive evaluation of PEFT techniques applied to BERTimbau-Base and BERTimbau-Large on the Question Answering task using the Portuguese SQuAD v1 dataset, executed on standard research hardware (8-20GB VRAM GPUs). We systematically explore 40 configurations combining methods (LoRA, DoRA, QLoRA, QDoRA), learning rates (4.25e-5 and 2e-4), and epochs (2 and 3), reporting complete results for scientific transparency. The findings demonstrate that LoRA achieves 95.8% of baseline performance on BERTimbau-Large (F1=81.32 vs. 84.86) with 73.5% reduction in training time. Quantized variants maintain competitive accuracy (QLoRA: F1=80.03, 94.3% of baseline) enabling training of BERTimbau-Large on 8GB GPUs, representing the first quantization study applied to BERTimbau. The results reveal three critical findings: (1) high learning rates (2e-4) are essential for PEFT, improving +6.20 F1 points versus standard learning rates that collapse quantized methods, (2) larger models are significantly more resilient to quantization (loss of 4.83 vs. 9.56 F1 points), and (3) DoRA offers no practical advantages over LoRA in QA, requiring +28% more time without performance improvements. This study provides evidence-based practical recommendations for selecting PEFT techniques in Brazilian Portuguese under computational constraints typical of academic and industrial environments.**

***Index Terms*—BERTimbau, LoRA, QLoRA, DoRA, PEFT, Quantization, Question Answering, Brazilian Portuguese**

I. Introduction

Large language models (LLMs) based on the Transformer architecture [1] have achieved extraordinary capabilities in recent years, surpassing human performance on multiple natural language processing tasks [2]. However, in the context of lower-resource languages such as Brazilian Portuguese, the landscape faces significant limitations. Unlike English, where a proliferation of specialized models exists, Brazilian Portuguese has a more restricted ecosystem. The most relevant and widely used models—such as Sabiá (7B parameters) [3], Tucano (1.1B parameters) [4], and BERTimbau (110M-335M parameters) [5]—are primarily adaptations of English architectures pre-trained on Brazilian corpora. Although these models have demonstrated competitive performance, they all share a critical limitation: they require full fine-tuning that consumes between 2-7 hours of training and 16-40GB of GPU memory, generating computational and energy costs that severely limit their reproducibility and accessibility in academic and industrial contexts with limited resources [6].

Facing this reproducibility barrier, quantization emerges as a promising technique to reduce model size by decreasing the numerical precision of its parameters [7], [8]. This technique, widely used in English language models with successful results, allows compressing models while maintaining competitive performance. However, despite some recent efforts, quantization techniques remain underexplored for Brazilian Portuguese models, and to the best of our knowledge there is almost no systematic evidence on their impact in downstream Question Answering tasks, representing a critical gap in the literature. Complementarily, Parameter-Efficient Fine-Tuning (PEFT) methods offer an additional solution to the resource problem. Techniques such as LoRA [9], which injects low-rank matrices updating only 0.1-1% of parameters, and DoRA [10], which proposes magnitude-direction decomposition, have demonstrated maintaining performance close to full fine-tuning in English. Their quantized variants—QLoRA and QDoRA—combine both approaches, enabling training of large models on GPUs with limited memory [7]. Nevertheless, empirical evidence is overwhelmingly concentrated on English-speaking models, leaving unanswered whether these techniques maintain their effectiveness in Brazilian models for tasks demanding deep understanding such as Question Answering.

This work addresses these gaps through the first evaluation of PEFT and quantization techniques applied to Brazilian Portuguese models. Focusing on BERTimbau—the most consolidated transformer model for Brazilian Portuguese, with established baselines (F1=82.50 on Base, F1=84.43 on Large for Portuguese SQuAD v1) [5]—we comparatively evaluate LoRA, QLoRA, DoRA, and QDoRA on the Base (110M parameters) and Large (335M parameters) variants for the Question Answering task. We exhaustively analyze the trade-off between performance (F1-score, Exact Match), temporal efficiency, and memory usage, comparing against full fine-tuning as baseline.

II. Background

A. The Challenge of Adapting Large Models

The dominant paradigm in NLP consists of pre-training massive models on large corpora and then adapting them through full fine-tuning for specific tasks [2]. However, as models grow according to scaling laws [11], [12], full fine-tuning becomes prohibitively expensive: it requires storing gradients and optimizer states for all parameters, demanding up to 12-18× more memory than inference [9]. For example, adjusting GPT-3 175B with the Adam optimizer requires approximately 1.2TB of GPU memory, making impractical the deployment of multiple specialized instances of the model. This resource barrier motivated the development of Parameter-Efficient Fine-Tuning (PEFT) methods, which seek to update only a small fraction of parameters while maintaining competitive performance.

B. LoRA: Low-Rank Adaptation

LoRA [9] addresses this challenge based on two key observations: (1) pre-trained models possess low intrinsic dimensionality [13], suggesting that the effective adaptation space is much smaller than the complete parameter space, and (2) weight updates during fine-tuning exhibit low-rank structure. Motivated by these findings, LoRA keeps pre-trained weights $W_0 \in \mathbb{R}^{d\times k}$ frozen and injects two trainable low-rank matrices: $A \in \mathbb{R}^{r\times k}$ and $B \in \mathbb{R}^{d\times r}$, where rank $r \ll \min(d,k)$ is typically $r \in [4, 64]$. For an input $x \in \mathbb{R}^k$, the output $h \in \mathbb{R}^d$ is computed as:

$$h = W_0 x + \frac{\alpha}{r} BAx \tag{1}$$

where α is a scaling factor, typically set as $\alpha = 2r$ to stabilize training. This decomposition drastically reduces trainable parameters: for a 768×768 matrix with $r = 16$, LoRA requires only $2 \times 16 \times 768 = 24,576$ parameters versus $768^2 = 589,824$ in full fine-tuning (96% reduction). Crucially, during inference, BA can be merged with W_0 eliminating any additional latency, a critical advantage over adapter-based methods [14].

Other PEFT methods include Prefix-Tuning [15], which optimizes task-specific continuous vectors prepended to input representations, Prompt Tuning [16], which learns soft prompts prepended to the input, and AdaLoRA [17], which adaptively allocates parameter budget according to the importance of each weight matrix. Recent studies propose a unified view of these methods [18], identifying common patterns in how they modify base model representations.

C. Quantization and QLoRA

Complementary to PEFT methods, quantization addresses efficiency from an orthogonal perspective: reducing the numerical precision of model weights from floating-point (typically 32 or 16 bits) to lower precision representations (8, 4, 3, or 2 bits) [7]. For uniform quantization to n bits, a weight $w \in \mathbb{R}$ is mapped to a quantized integer $\tilde{w}$ through:

$$\tilde{w} = \text{round}\left(\text{clamp}\left(\frac{w-z}{s}, -2^{n-1}, 2^{n-1}-1\right)\right) \tag{2}$$

where $s \in \mathbb{R}^+$ is the scale factor and $z \in \mathbb{R}$ is the zero point. Previous work explores 8-bit optimizers [19] to reduce memory footprint during training, while GPTQ [8] proposes post-training quantization based on layer-wise error minimization.

QLoRA [7] represents a synergistic integration of quantization and LoRA, demonstrating for the first time that it is possible to fine-tune 4-bit quantized models without performance degradation. QLoRA introduces three key technical innovations: (1) **4-bit NormalFloat (NF4)**, a data type designed for weights with normal distribution using quantile-based quantization, being theoretically optimal for deep neural networks; (2) **Double Quantization**, which also quantizes scale and zero-point parameters, reducing average footprint by 0.37 bits per parameter; and (3) **Paged Optimizers**, which uses unified memory to handle memory spikes. In QLoRA, base weights are quantized to 4 bits and remain frozen, while LoRA matrices are maintained in bfloat16.

D. DoRA and QDoRA

DoRA [10] addresses the persistent accuracy gap of LoRA through weight decomposition into magnitude and direction components. Inspired by Weight Normalization [20], DoRA decomposes each matrix W as:

$$W = m\frac{V}{\|V\|_c} \tag{3}$$

where $m \in \mathbb{R}^d$ are the magnitudes (column norms) and $V \in \mathbb{R}^{d\times k}$ represents the normalized direction. During fine-tuning, DoRA applies the LoRA update only to the directional component and trains an additional vector Δm to adjust magnitudes. QDoRA naturally extends DoRA to the quantized regime, combining the decomposition with QLoRA's quantization techniques.

E. Question Answering and Metrics

In extractive Question Answering (QA) such as SQuAD v1 [21], given a context C and a question Q, the model must predict start s and end e positions that delimit the answer span extracted from the context. SQuAD v2 [22] extends this framework including unanswerable questions, while Natural Questions [23] provides an alternative benchmark with real user questions. Standard metrics are **F1-score** (harmonic mean between precision and recall at token level) and **Exact Match (EM)** (percentage of exactly correct predictions after normalization).

III. Related Work

Early efforts in pre-trained models for Brazilian Portuguese focused on BERT-type architectures, notably BERTimbau in its Base and Large variants [5]. Subsequent studies explored autoregressive LLMs such as Sabiá [3] and Tucano [4], designed specifically for text generation. While multilingual models like XLM-R [24] demonstrate cross-lingual capabilities, BERTimbau's specific focus on Brazilian Portuguese makes it particularly suitable for downstream tasks.

In extractive QA, SQuAD-BR became the standard benchmark for evaluating models in Portuguese, achieving state-of-the-art results on F1 and Exact Match metrics on the SQuAD v1 dataset. Previous work in Portuguese NLP includes named entity recognition [25] and other tasks beyond QA. However, these works do not consider associated computational costs.

In the field of English LLMs, Parameter-Efficient Fine-Tuning (PEFT) methods emerged such as LoRA [9] and DoRA [10], which introduce low-rank adaptations to drastically reduce the number of updated parameters while maintaining competitive performance. Parallel to this, quantization techniques like QLoRA [7] enable storing weights in low precision (4 bits) and combining this with PEFT to train LLMs on memory-limited GPUs. Benchmarks such as GLUE [26] and SuperGLUE [27] have established standards for evaluating language models in English, albeit with limited representation of low-resource languages.

Current literature offers: (i) robust Portuguese models focused on absolute performance, (ii) strong QA baselines on SQuAD-BR based on full fine-tuning, and (iii) a mature body of PEFT techniques evaluated primarily in English. What is missing is a study combining these three lines, systematically evaluating PEFT and quantization on Brazilian Portuguese models for QA. This is precisely the gap this work aims to fill.

IV. Methodology

A. Dataset and Model Configuration

We use the SQuAD v1 dataset [21] in its Brazilian Portuguese translated version, consisting of 87,599 question-answer pairs for training and 10,570 for evaluation. We evaluate two BERTimbau variants [5]: (1) **BERTimbau-Base** with 12 layers, 768 hidden dimensions, 12 attention heads, and 110M parameters, and (2) **BERTimbau-Large** with 24 layers, 1024 hidden dimensions, 16 attention heads, and 335M parameters. Both models were pre-trained on the brWaC corpus with 2.68 billion tokens.

B. PEFT Techniques Configuration

For all PEFT methods, we use: LoRA rank $r = 16$, scaling factor $\alpha = 32$, target modules (query, key, value, and output matrices of the attention mechanism), and dropout 0.1. For quantized variants, we apply 4-bit NF4 quantization to base weights with double quantization enabled and compute dtype bfloat16.

We systematically explore two learning rates: the standard for BERT (4.25e-5) and a high learning rate optimized for PEFT (2e-4), training for 2 and 3 epochs. Common parameters: AdamW optimizer [28], [29], weight decay 0.01, batch size 16 (Base) and 8 (Large), max sequence length 384, gradient clipping norm 1.0.

C. Computational Infrastructure

Experiments were executed on a workstation with: (1) **Primary GPU**: NVIDIA RTX A4500 with 20GB VRAM, used for all BERTimbau-Large experiments and full fine-tuning, and (2) **Secondary GPU**: NVIDIA GeForce RTX 2080 Ti with 8GB VRAM, used for BERTimbau-Base experiments and validation of quantized methods viability. Software: CUDA 12.2, PyTorch 2.1.0, Transformers 4.36.0, PEFT 0.7.1, bitsandbytes 0.41.0.

This hardware configuration is representative of academic environments with moderate resources, demonstrating the viability of PEFT techniques outside high-performance infrastructures. The capacity to train BERTimbau-Large with QLoRA on the 8GB RTX 2080 Ti validates the democratization of access to large models through quantization.

V. Experimental Results

A. Performance on BERTimbau-Base

Tables I and II present complete results for BERTimbau-Base under different learning rate and epoch configurations. The baseline with full fine-tuning achieves its best performance with lr=4.25e-5 in 2 epochs (F1=82.79, EM=70.91).

Figure 1 presents the exhaustive performance analysis for BERTimbau-Base across all evaluated configurations. Each panel compares F1-score and Exact Match for full fine-tuning (Base QA) and PEFT methods under different epochs and learning rates.

TABLE I
BERTimbau-Base with high learning rate (2e-4)

Method	Ep.	F1	EM	Time
Base QA	2	79.74	67.15	01:40:02
LoRA	2	**78.01**	**64.85**	00:31:37
QLoRA	2	73.23	60.26	00:30:03
DoRA	2	78.01	64.89	00:40:23
QDoRA	2	74.41	61.26	00:42:03
Base QA	3	78.33	65.54	02:29:04
LoRA	3	78.01	65.03	00:46:47
QLoRA	3	74.16	61.24	00:44:42
DoRA	3	78.27	65.08	00:59:59
QDoRA	3	74.46	61.32	01:02:44

The contrast between learning rates is dramatic. With lr=2e-4, LoRA achieves F1=78.01 (94.2% of baseline), while with lr=4.25e-5 it only achieves F1=71.81 (86.7% of baseline), a difference of **+6.20 F1 points**. This pattern replicates across all PEFT methods. Quantized variants collapse completely with standard learning rate (QLoRA: F1=53.52, QDoRA: F1=54.10), falling more than 20 points below baseline.

Among PEFT techniques with lr=2e-4, LoRA and DoRA tie in performance (F1=78.01) in 2 epochs, but LoRA is 27.8% faster. Quantized variants show significant degradation: QLoRA achieves F1=73.23 (-9.56 points vs. baseline) and QDoRA F1=74.41. This sensitivity suggests that BERTimbau-Base loses critical information during 4-bit compression.

TABLE II
BERTimbau-Base with standard learning rate (4.25e-5)

Method	Ep.	F1	EM	Time
Base QA	2	**82.79**	**70.91**	01:40:04
LoRA	2	71.81	58.07	00:31:49
QLoRA	2	53.52	40.54	00:30:02
DoRA	2	71.36	57.68	00:40:13
QDoRA	2	54.10	41.15	00:42:10
Base QA	3	82.18	70.40	02:28:52
LoRA	3	72.01	58.32	00:41:30
QLoRA	3	53.19	39.81	00:40:20
DoRA	3	71.50	57.65	00:53:58
QDoRA	3	58.42	45.37	00:55:00

B. Performance on BERTimbau-Large

Tables III and IV present complete results for BERTimbau-Large. The baseline achieves its best performance with lr=4.25e-5 in 2 epochs (F1=84.86, EM=73.00). Notably, the baseline with lr=2e-4 collapses completely (F1=3.02), demonstrating that high learning rates are detrimental for full fine-tuning but essential for PEFT.

Figure 2 shows the comprehensive performance of BERTimbau-Large under all configurations. Notably, it highlights the collapse of full fine-tuning with high learning rates while LoRA and its quantized variants maintain strong performance, demonstrating the model's resilience.

TABLE III
BERTimbau-Large with high learning rate (2e-4)

Method	Ep.	F1	EM	Time
Base QA	2	3.02	0.03	05:15:30
LoRA	2	**81.32**	**68.67**	01:23:41
QLoRA	2	80.03	67.17	01:19:15
DoRA	2	80.61	68.09	01:47:37
QDoRA	2	77.96	65.05	01:57:30
Base QA	3	5.14	0.11	07:50:02
LoRA	3	81.27	68.67	02:05:20*
QLoRA	3	80.28	67.63	01:57:39
DoRA	3	81.22	68.70	02:40:52*
QDoRA	3	79.61	66.99	02:54:52

*Estimated time

The learning rate impact on Large is consistent with Base but more pronounced. With lr=2e-4, LoRA achieves F1=81.32 (95.8% of baseline), while with lr=4.25e-5 it only achieves F1=75.65 (89.1%), a difference of **+5.67 F1 points**. Quantized variants also exhibit severe degradation with standard lr: QLoRA drops from F1=80.03 to 68.23 (-11.80 points).

BERTimbau-Large demonstrates greater resilience to PEFT techniques than Base. With lr=2e-4 in 2 epochs, LoRA retains 95.8% of baseline performance versus 94.2% on Base, with 73.5% reduction in training time. Crucially, quantized variants on Large show much less degradation than on Base. QLoRA achieves F1=80.03 (94.3% of baseline, -4.83 points), compared to F1=73.23 (88.5%, -9.56 points) on Base. This 2.0× difference in degradation confirms that larger models are significantly more robust to aggressive quantization.

TABLE IV
BERTimbau-Large with standard learning rate (4.25e-5)

Method	Ep.	F1	EM	Time
Base QA	2	**84.86**	**73.00**	05:15:39
LoRA	2	75.65	62.21	01:23:28
QLoRA	2	68.23	54.92	01:19:12
DoRA	2	74.93	62.02	01:47:46
QDoRA	2	70.32	56.88	01:57:30
Base QA	3	83.74	72.04	07:50:46
LoRA	3	81.28	68.63	02:05:08*
QLoRA	3	71.03	57.66	01:58:54*
DoRA	3	77.18	63.98	02:41:23*
QDoRA	3	71.24	58.15	02:55:23

*Estimated time

A notable finding is the baseline collapse with lr=2e-4 (F1=3.02), in contrast to the success of PEFT methods under the same learning rate. This suggests that the low-rank structure of LoRA/DoRA acts as an implicit regularizer, preventing training divergence.

C. Critical Learning Rate Impact

Exhaustive results reveal that learning rate is the most determinant factor for PEFT success, surpassing in importance the choice of specific method, number of epochs, or even quantization application. Fig. 3 visualizes this dramatic contrast.

On BERTimbau-Base with LoRA, lr=2e-4 achieves F1=78.01 versus F1=71.81 with lr=4.25e-5, an improvement of +6.20 points (+8.6% relative). On Large, the improvement is +5.67 points (+7.5% relative). For quantized methods, the impact is more dramatic: QLoRA improves +19.71 points on Base (53.52 → 73.23) and +11.80 on Large (68.23 → 80.03). With standard learning rate, quantized methods collapse, falling more than 20-30 points below baseline.

VI. Discussion

A. Scalability with Model Size

Results demonstrate that PEFT techniques scale favorably with model size. BERTimbau-Large exhibits: (1) lower performance degradation with LoRA (95.8% vs. 94.2% on Base), (2) greater resilience to quantization (loss of 4.83 vs. 9.56 points), and (3) higher computational efficiency. This trend suggests that redundancy in large models allows low-rank adaptations to capture necessary transformations more effectively.

B. Learning Rate for PEFT: Critical Finding

Learning rate is the most determinant factor for PEFT. With high lr (2e-4), LoRA achieves competitive performance, but with standard lr (4.25e-5) performance collapses. The difference of +6.20 points on Base and +5.67 on Large represents improvements of 8.6% and 7.5% relative. For quantized methods, QLoRA improves +19.71 points on Base using high lr.

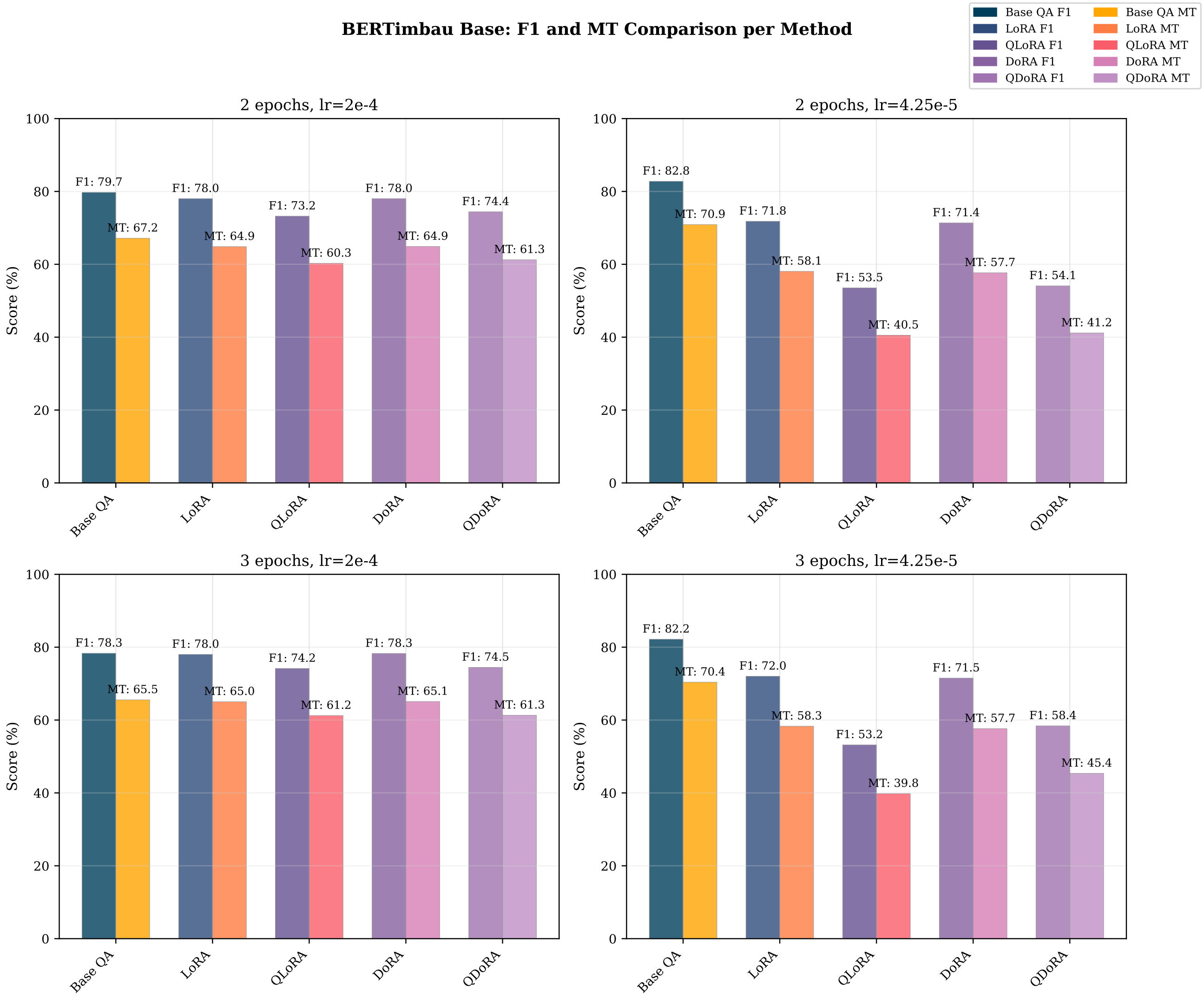

Fig. 1. Exhaustive performance analysis for BERTimbau-Base under all evaluated configurations. Each panel shows F1-score (dark bars) and Exact Match (light bars) for full fine-tuning (Base QA) and PEFT methods (LoRA, QLoRA, DoRA, QDoRA). The four panels correspond to: (top left) 2 epochs with lr=2e-4, (top right) 2 epochs with lr=4.25e-5, (bottom left) 3 epochs with lr=2e-4, (bottom right) 3 epochs with lr=4.25e-5. Key observations: LoRA/DoRA achieve F1=78.0 with high lr versus F1=71.8 with standard lr (+6.2 points); quantized methods collapse with standard lr (QLoRA F1=53.5); 3 epochs do not significantly improve results.

While lr=2e-4 is essential for PEFT, it is catastrophic for full fine-tuning on Large (F1=3.02 vs. 84.86). This opposite behavior reflects fundamentally different optimization dynamics. In full fine-tuning, each parameter contributes directly in a 335M dimensional space. In contrast, LoRA operates in a compressed 13.4M dimensional subspace, where updates must propagate through the $\Delta W = BA$ factorization. This compression requires more aggressive gradients to compensate for reduced expressive capacity. The low-rank structure also acts as an implicit regularizer: even with high lr, updates are constrained to the rank r subspace, preventing divergence.

C. LoRA vs. DoRA Comparison

DoRA achieves similar or slightly inferior performance to LoRA on both models but with significant time overhead (+27.8% on Base, +28.6% on Large). The explicit magnitude-direction decomposition, while theoretically more expressive, does not translate to practical improvements for QA in Portuguese. For practical applications, LoRA is preferable due to its simplicity and equivalent or superior efficiency.

D. Quantization Impact

4-bit quantization introduces differentiated degradation according to model size. On Base, QLoRA loses 11.5% of baseline performance (9.56 F1 points), while on Large it loses only 5.7% (4.83 points). This 2.0× difference suggests that smaller models have less redundancy to absorb quantization noise. For applications with severe memory constraints, QLoRA with Large offers better trade-off, achieving competitive perfor-

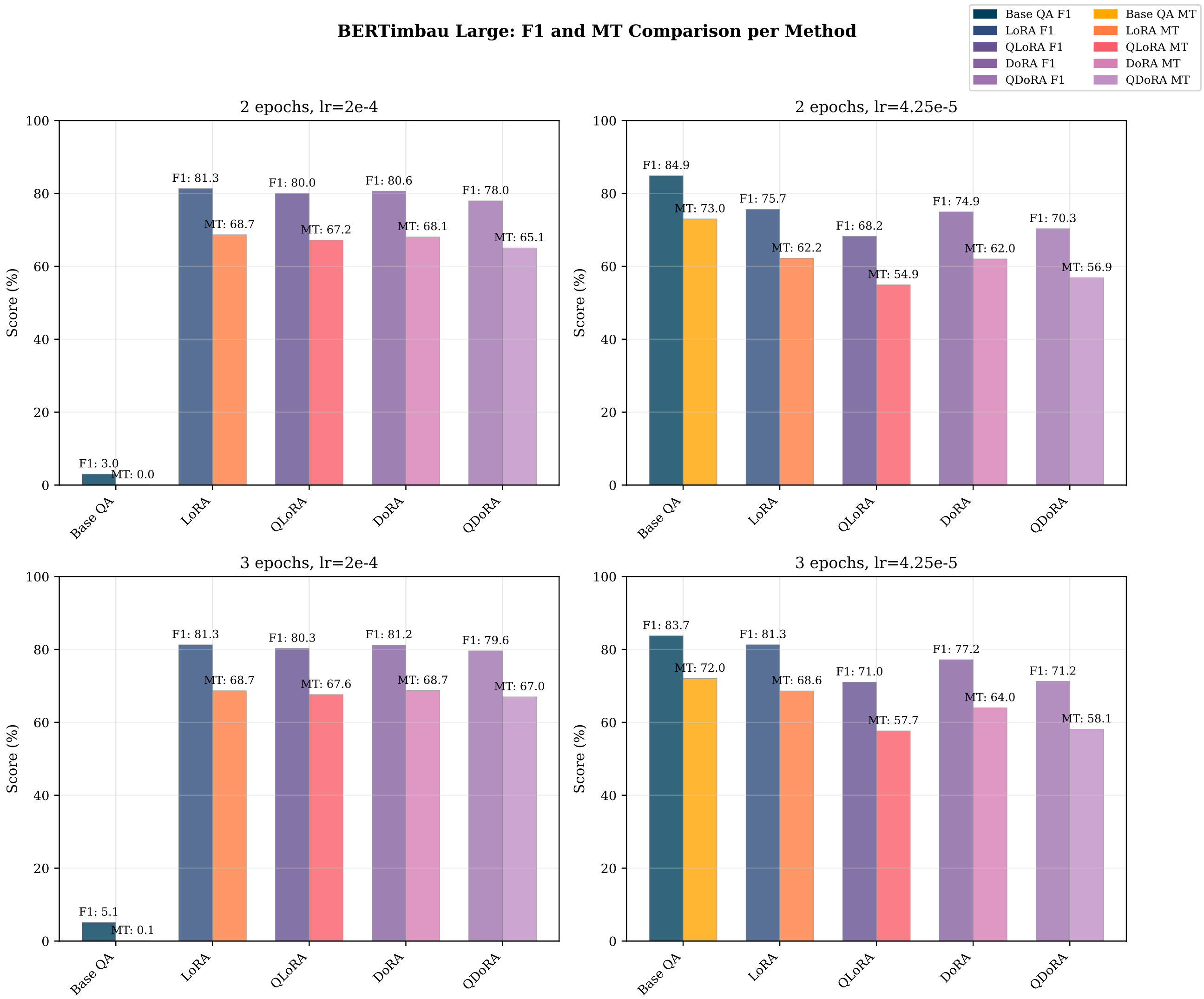

Fig. 2. Exhaustive performance analysis for BERTimbau-Large under all evaluated configurations. Layout identical to Fig. 1. Critical findings: baseline collapses dramatically with high lr (F1=3.0 in 2 epochs, F1=5.1 in 3 epochs) while LoRA thrives (F1=81.3); with standard lr, baseline achieves F1=84.9 but LoRA degrades to F1=75.7 (-5.7 points vs. high lr); QLoRA maintains F1=80.0-80.3 with high lr, demonstrating viability for training Large on 8GB GPUs; Large shows greater resilience to quantization than Base (QLoRA loses 4.8 points vs. 9.6 on Base).

mance (F1=80.03, 94.3% of baseline) with minimal memory requirements.

E. Practical Recommendations

For 8GB VRAM: QLoRA + Large (F1=80.03, 94.3% baseline), validated on RTX 2080 Ti, democratizes access to large models.

For 16-20GB VRAM: LoRA + Large (F1=81.32, 95.8% baseline, 73.5% time reduction), common in academic laboratories.

For small models: Avoid quantization (loss 9.56 vs 4.83). LoRA without quantization preferable.

For implementations: Critical LR: 2e-4 (not 4.25e-5). DoRA not recommended (+28% overhead).

Viability on accessible hardware: This study demonstrates that advanced PEFT techniques are viable on standard infrastructure. BERTimbau-Large was successfully fine-tuned with QLoRA on 8GB GPUs, without need for specialized hardware. This accessibility is crucial for democratizing NLP research for Brazilian Portuguese in contexts with limited resources.

VII. Conclusion

This work presents the first systematic evaluation of PEFT and quantization techniques applied to BERTimbau on QA, executed on accessible hardware. Main findings reveal that: (1) LoRA achieves 95.8% of full fine-tuning performance with 73.5% time reduction, (2) quantized techniques maintain competitive accuracy (QLoRA: 94.3% of baseline), this being the first quantization application to BERTimbau, (3) significantly higher learning rates (2e-4) are essential for PEFT, improving +6.20 F1 points, (4) larger models exhibit greater resilience to

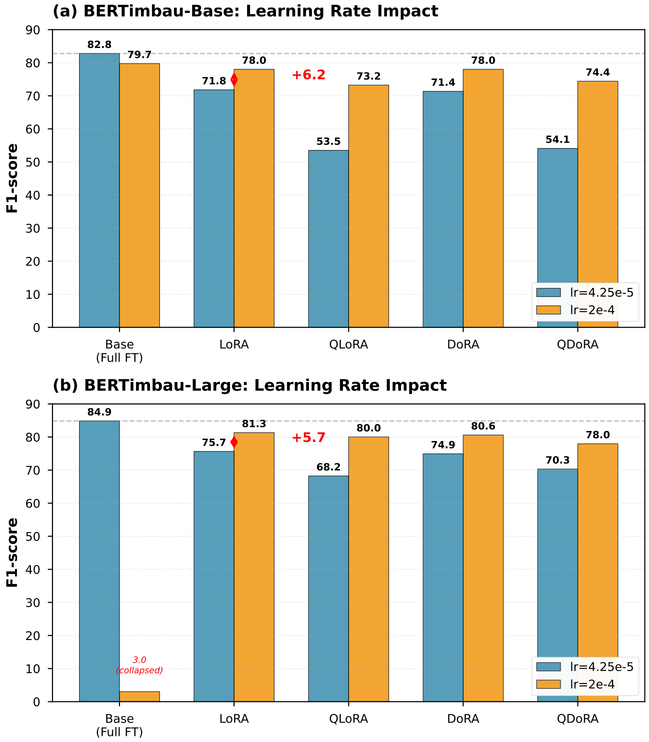

Fig. 3. Learning rate impact on F1 performance with 2 epochs. High learning rates (2e-4) are essential for PEFT, while they collapse full fine-tuning on Large.

quantization (loss of 4.83 vs. 9.56 points), and (5) DoRA offers no practical advantages over LoRA for QA in Portuguese.

Contributions include: (i) the first quantization study applied to BERTimbau, (ii) evidence-based practical recommendations for PEFT technique selection, (iii) identification of learning rate as a frequently underestimated critical factor, and (iv) validation of viability on accessible hardware (8-20GB).

Future work includes: (1) exploration of more aggressive quantization (3-bit, 2-bit) with Large, (2) evaluation of PEFT on other Portuguese NLP tasks, (3) analysis of different LoRA configurations (rank, α), and (4) extension to Brazilian autoregressive models such as Sabiá and Tucano.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2023. [Online]. Available: https://arxiv.org/abs/1706.03762

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019. [Online]. Available: https://arxiv.org/abs/1810.04805

[3] R. Pires, H. Abonizio, T. S. Almeida, and R. Nogueira, *Sabiá: Portuguese Large Language Models*. Springer Nature Switzerland, 2023, p. 226–240. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-45392-2$_1$5

[4] N. K. Corrêa, A. Sen, S. Falk, and S. Fatimah, "Tucano: Advancing neural text generation for portuguese," *Patterns*, vol. 6, no. 11, p. 101325, Nov. 2025. [Online]. Available: http://dx.doi.org/10.1016/j.patter.2025.101325

[5] F. Souza, R. Nogueira, and R. Lotufo, "Bertimbau: Pretrained bert models for brazilian portuguese," in *Intelligent Systems*, R. Cerri and R. C. Prati, Eds. Cham: Springer International Publishing, 2020, pp. 403–417.

[6] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in nlp," 2019. [Online]. Available: https://arxiv.org/abs/1906.02243

[7] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," 2023. [Online]. Available: https://arxiv.org/abs/2305.14314

[8] E. Frantar, S. Ashkboos, T. Hoefler, and D. Alistarh, "Gptq: Accurate post-training quantization for generative pre-trained transformers," 2023. [Online]. Available: https://arxiv.org/abs/2210.17323

[9] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," 2021. [Online]. Available: https://arxiv.org/abs/2106.09685

[10] S.-Y. Liu, C.-Y. Wang, H. Yin, P. Molchanov, Y.-C. F. Wang, K.-T. Cheng, and M.-H. Chen, "Dora: Weight-decomposed low-rank adaptation," 2024. [Online]. Available: https://arxiv.org/abs/2402.09353

[11] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," 2020. [Online]. Available: https://arxiv.org/abs/2001.08361

[12] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de Las Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, and L. Sifre, "Training compute-optimal large language models," 2022. [Online]. Available: https://arxiv.org/abs/2203.15556

[13] A. Aghajanyan, L. Zettlemoyer, and S. Gupta, "Intrinsic dimensionality explains the effectiveness of language model fine-tuning," 2020. [Online]. Available: https://arxiv.org/abs/2012.13255

[14] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," 2019. [Online]. Available: https://arxiv.org/abs/1902.00751

[15] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," 2021. [Online]. Available: https://arxiv.org/abs/2101.00190

[16] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," 2021. [Online]. Available: https://arxiv.org/abs/2104.08691

[17] Q. Zhang, M. Chen, A. Bukharin, N. Karampatziakis, P. He, Y. Cheng, W. Chen, and T. Zhao, "AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning," in *International Conference on Learning Representations (ICLR)*, 2023. [Online]. Available: https://openreview.net/forum?id=lq62uWRJjiY

[18] J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig, "Towards a unified view of parameter-efficient transfer learning," 2022. [Online]. Available: https://arxiv.org/abs/2110.04366

[19] T. Dettmers, M. Lewis, S. Shleifer, and L. Zettlemoyer, "8-bit optimizers via block-wise quantization," 2022. [Online]. Available: https://arxiv.org/abs/2110.02861

[20] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," 2016. [Online]. Available: https://arxiv.org/abs/1602.07868

[21] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," 2016. [Online]. Available: https://arxiv.org/abs/1606.05250

[22] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for squad," 2018. [Online]. Available: https://arxiv.org/abs/1806.03822

[23] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M.-W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov, "Natural questions: A benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 452–466, 2019. [Online]. Available: https://aclanthology.org/Q19-1026/

[24] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 8440–8451. [Online]. Available: https://aclanthology.org/2020.acl-main.747/

[25] P. V. Quinta de Castro, N. Félix Felipe da Silva, and A. da Silva Soares, "Portuguese named entity recognition using lstm-crf," in *Computational*

Processing of the Portuguese Language, A. Villavicencio, V. Moreira, A. Abad, H. Caseli, P. Gamallo, C. Ramisch, H. Gonçalo Oliveira, and G. H. Paetzold, Eds. Cham: Springer International Publishing, 2018, pp. 83–92.

[26] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," 2019. [Online]. Available: https://arxiv.org/abs/1804.07461

[27] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Superglue: A stickier benchmark for general-purpose language understanding systems," 2020. [Online]. Available: https://arxiv.org/abs/1905.00537

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017. [Online]. Available: https://arxiv.org/abs/1412.6980

[29] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2019. [Online]. Available: https://arxiv.org/abs/1711.05101