

TaskRank

Stanford CS224N Default Project

Peter Benitez

Department of Computer Science
Stanford University
peterjb3@stanford.edu

Fabio Ibanez

Department of Computer Science
Stanford University
fabioi@stanford.edu

Abstract

We present **TaskRank**, a systematic evaluation of how LoRA and quantization affect performance across tasks with contrasting structural demands and data scales. We compare full fine-tuning against LoRA at ranks $r \in \{4, 8, 16, 32, 64, 128\}$ applied to both attention and MLP layers, combined with 8-bit and 4-bit (NF4) quantization for paraphrase detection (141k training examples) and sonnet generation (143 training examples) using GPT-2 (124M). We find that LoRA’s performance gap relative to full fine-tuning varies dramatically across tasks: 6.1 points on paraphrase detection but less than 1 point on sonnet generation. We attribute this primarily to dataset size rather than task structure, with only 143 examples; full fine-tuning cannot exploit its full parameter space, making the low-rank constraint negligible. We find MLP LoRA consistently outperforms attention-only LoRA on both tasks, and that rank r has minor impact on score. Quantization preserves quantitative performance on both tasks, but qualitative analysis of generated sonnets reveals subtle stylistic drift in word choice that aggregate metrics do not capture.

1 Key Information to include

- Mentor: Shicheng Liu
- External Collaborators (if you have any): N/A
- Sharing project: N/A
- Late Days (if you are using any):

Team member	Late days remaining
Fabio Ibanez	2
Peter Benitez	2
Total pooled	4
Team size	2
Late days applied for the report	<>

2 Introduction

As pre-trained language models grow larger, the standard approach of full fine-tuning becomes increasingly infeasible. The traditional transfer learning paradigm involves pre-training models on general domain data and then fine-tuning all parameters for specific downstream tasks. Deploying independent instances of fully fine-tuned models is prohibitively expensive because each fine-tuned version contains as many parameters as the original model. The hardware demands are enormous since full fine-tuning of GPT-3 175B with the Adam optimizer, as an example, requires 1.2TB of VRAM during training. These constraints in full fine-tuning have motivated a new set of parameter-efficient fine-tuning (PEFT) techniques that freeze the pre-trained model weights and add a small finite number of parameters for fine-tuning. Across these approaches, there’s an open question of

which downstream tasks have the largest performance degradation introduced by parameter-efficient fine-tuning, along with other compression techniques like quantization. This work has implications on how machine learning practitioners should apply fine-tuning and memory optimization techniques. If the performance cost is uniform across various tasks, even those with structural requirements, then researchers can apply these methods more liberally. However, if certain tasks are more susceptible to performance degradation under PEFT or quantization, this is important to consider when selecting a fine-tuning strategy. We evaluate this tradeoff in TaskRank, where we compare the effects of LoRA, across a sweep of ranks r , and various degrees of quantization precision. The two tasks that we use for evaluation are sonnet generation and paraphrase detection. We choose these tasks because of their structural contrasts. Paraphrase detection on the Quora Question Pair dataset is a *semantic* binary classification task where the model must determine semantic similarity, requiring no specific output structure. Sonnet generation, in contrast, has strict structural requirements that the model must learn. We expect LoRA to perform close to full fine-tuning on paraphrase detection, given its semantic nature, but we hypothesize that sonnet generation may show a larger performance gap between LoRA and full fine-tuning due to the task’s structural demand. We further hypothesize that adding quantization on top of LoRA will widen this gap, particularly for sonnet generation, since the reduced model weight precision may make it harder to capture fine-grained structural patterns.

3 Related Work

3.1 Parameter-efficient Fine-tuning (PEFT)

There is an extensive line of research into parameter-efficient fine-tuning techniques that freeze the pre-trained model weights. Adapter layers (Houlsby et al. 2019)[1] insert small trainable modules between the layers of a frozen pre-trained model, but they introduce additional inference latency that can be significant in low-latency scenarios. Prefix tuning (Li & Liang 2021)[2] optimizes a sequence of continuous input vectors prepended to the model input, but the authors state that it is difficult to optimize and its performance changes unpredictably as the number of trainable parameters increases. Prefix tuning also uses part of the available sequence length, which reduces the sequence length left for the actual downstream task. Both these existing methods often fail to match the performance of full fine-tuning, creating an undesirable tradeoff between efficiency and model quality. Hu et al. [3] introduce LoRA which freezes all of the pre-trained model weights and injects pairs of small trainable rank decomposition matrices into the layers of the Transformer architecture. The weight update in a forward pass is expressed as, where W_0 represents the frozen pretrained weights:

$$h = W_0x + \Delta Wx$$

LoRA decomposes the update ΔW into a low-rank factorization BA where $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$, with $r \ll \min(d, k)$ [3]. During training, only the small A and B matrices are updated rather than the full weight matrices. At inference time, the learned low-rank matrices can be merged into the original pre-trained weights, meaning LoRA adds zero additional inference latency. A key motivation for LoRA comes from work by Li et al. (2018)[4] and Aghajanyan et al. (2020)[5], which showed that learned over-parameterized models reside on a low intrinsic dimension. Building on this, the authors hypothesize that the change in weights during model adaptation also has a low intrinsic rank.

3.2 Quantization & QLoRA

Quantization is a technique that stores model weights in lower precision which reduces the model’s memory footprint. This technique also reduces the *representational fidelity* of the model. This reduction in representational fidelity may affect the model’s ability to identify and follow fine-grained patterns in tasks. Dettmers et al. [6] introduce QLoRA which in addition to LoRA, quantizes the pre-trained model weights. In their investigation, Dettmers et al. find that lower resolution (ex. nf4 vs 8-bit) quantization does not affect performance on the tasks that they evaluated which were GLUE, Super-NaturalInstructions, and MMLU. The evaluations in the QLoRA paper, like the work in Hu et al. [3] is mainly focused on accuracy and ROGUE-L, which are measure the semantic correctness and lexical overlap in responses, as opposed to strict structural requirements. In this project we address whether or not quantization precision affects performance *across* tasks that are evaluated on more structural metrics like CHRF score, like sonnet generation.

3.3 PEFT’s Effects on Tasks

Prior work suggest that PEFT techniques do indeed affect model performance on downstream tasks. Wang et al.[7] fine-tunes Llama-2-7B using full fine-tuning and LoRA variants across a variety of tasks, including dialogue generation, mathematical reasoning, and coding abilities. On dialogue generation, they find that LoRA does marginally outperform full fine-tuning, however, on mathematical reasoning LoRA scores 42.08 compared to full fine-tuning with 59.36. For coding abilities, which is the most structured task, LoRA scores 14.76 while full fine-tuning scores 35.31. Wang et al.’s work suggests that as tasks get more structured PEFT techniques like LoRA lead to larger degress of performance degradation.

Recent work also decomposes the mechanics of why LoRA and full fine-tuning may diverge. Shuttleworth et al. [8] show LoRA and full fine-tuning learn structurally different weights when analyzing the singular value decomposition of weight matrices trained via LoRA versus full fine-tuning. Their work focuses on *intruder dimensions*, which are high ranking singular vectors that have little similarity to any pre-trained weight vector, which LoRA introduces, while fine-tuning does not share this characteristic. In this project we try to characterize the differences that these structural differences in training may have on downstream task performance.

4 Approach

For our project, we build on top of the default CS224N GPT-2 final project. We implemented the core transformer forward pass, AdamW optimizer, and forward methods for the tasks: sonnet generation, paraphrase detection, and sentiment classification. For our extension and main analysis we implement various configurations of parameter-efficient fine-tuning like LoRA and LoRA with quantization. We add LoRA layers to both the attention layer, specifically the Q and V projections as in Hu et al. [3], and also the mlp layer inspired by work that shows using LoRA in the attention layer only underperforms full fine-tuning. Adding LoRA to other layers, like the MLP layer, has been found to close this gap [9].

4.1 Baselines

Our baselines are the full fine-tuned GPT-2 (124M), which updates all of the model parameters. To isolate the effects that rank and quanization we kept the model size fixed to GPT-2 (124M) for all of the experiments we do analysis on. For the paraphrase detection task we use 3 epochs, a batch size of 32, and learning rate of 1×10^{-5} , achieving a dev accuracy of 0.889. For sonnet generation we did a hyperparameter sweep to find the best hyperparameter configuration for full fine-tuning and we use 10 epochs, a batch size of 8, and learning rate of 1×10^{-4} , which achieves a best dev chrF score of 42.21.

4.2 LoRA

We use LoRA, specifically the `loralib` library ¹. This freezes the pre-trained weights and allows us to utilize trainable low-rank matrices $\Delta W = BA$ into the transformer layers. We use LoRA in two different layers, the attention layer and the MLP layer. For the attention layer we use LoRA for W_q and W_v matrices following the work of Hu et al. [3]. for MLP, We apply LoRA to both of the projections in the MLP block. We do a sweep over ranks $r \in \{4, 8, 16, 32, 64, 128\}$, we believe that these capture low, medium, and high rank regimes at this scale.

Table 1: LoRA experimental configurations.

Target	Ranks	Quantization
Attention (W_q, W_v)	4, 8, 16, 32, 64, 128	none, int8, nf4
MLP (dense, out)	4, 8, 16, 32, 64, 128	none, int8, nf4

¹<https://github.com/microsoft/LoRA>

4.3 Quantization

We implement two custom layer classes that combine quantization from `bitsandbytes` (bnb) and LoRA. One of the classes uses 8 bit quantization via `bnb.nn.Linear8BitLt` and the other uses `bnb.nn.Linear4bit`. We evaluate quantization for both tasks across, all, attention-only, and mlp-only layers, for the subset of $r \in \{8, 32, 128\}$ due to computational constraints. This approach enables us to characterize both sources which reduce representational fidelity, as LoRA reduces the expressivity of the weight updates and quantization reduces the precision in the frozen pre-trained weights. Combining these factors allows us to see whether or not sonnet generation is more sensitive to either or both of these factors that reduce representational fidelity.

5 Experiments

5.1 Data

We use the datasets provided by the CS224N teaching staff for the default final project.

- *Paraphrase Detection*: **Quora Dataset**: 141,506 training examples, 20,215 dev, 40,431 test
- *Sonnet Generation*: **Sonnet Dataset**: 143 training examples, 12 test

5.2 Evaluation method

To evaluate our tasks we used the chrF metric for the sonnet generation task, which is a character level n-gram comparison metric. As well as accuracy against ground-truth labels for the paraphrase detection task, with submissions to both dev/test leaderboards.

5.3 Experimental details

All experiments use GPT-2 (124M parameters) with the AdamW optimizer and seed 11711. We evaluate parameter-efficient fine-tuning using LoRA and quantization across both tasks. We apply LoRA using the *loralib* library. Quantization is applied post-training to the frozen pretrained weights at 8-bit and 4-bit precision using `bitsandbytes`.

Full fine-tuning baselines:

- *Paraphrase Detection*: $\text{lr}=1\text{e-}5$, batch size=32, epochs=3 (Preliminary sweeps at $\text{lr}=1\text{e-}4$ and $1\text{e-}6$ showed a ~ 4 point performance degradation, justifying the subsequent fixed lr .)
- *Sonnet Generation*: $\text{lr}=1\text{e-}4$, batch size=8, temperature=1.2, top_p=0.9, epochs=10 (selected via hyperparameter sweep, Appendix Table 5)

LoRA:

- *Ranks*: $r \in \{4, 8, 16, 32, 64, 128\}$
- *Paraphrase Detection*: $\text{lr}=1\text{e-}5$, batch size=32, epochs=3
- *Sonnet Generation*: lr swept from $\text{lr}=1\text{e-}5$ to $1\text{e-}3$; best performance at $\text{lr}=1\text{e-}3$ for both layer types. All other sonnet hyperparameters match baseline full-fine tuning.

Quantization:

- *Ranks*: $r \in \{8, 32, 128\}$ at both 8-bit and 4-bit precision for both layer types.
- Extra sonnet experiments run on best LoRA results (MLP rank 4 and attention rank 32).

5.4 Results

Paraphrase Detection (Table 2): Full fine-tuning of GPT-2 achieves 0.889 dev accuracy, exceeding the staff baseline of 0.882. All LoRA configuration show a persistent performance gap relative to full fine-tuning. MLP LoRA outperforms attention LoRA across all ranks, peaking at 0.828 (rank 8) compared to 0.811 (rank 4) for attention, a consistent 2 point difference. However, even the best

LoRA configuration falls 6.1 points below full fine-tuning. Within each layer type, performance is flat across ranks: attention LoRA ranges from 0.799 to 0.811 and MLP LoRA from 0.820 to 0.828 across ranks 4 to 128, suggesting rank is not the limiting constraint on the paraphrase task. Quantization has minimal impact on LoRA performance for paraphrase detection. Both 8-bit and 4-bit quantization degrade accuracy by less than 1 point across all tested configurations, indicating that reduced precision does not meaningfully compound with LoRA for this task.

Method	Layer	Rank	Dev Accuracy			Test Acc
			Default (FP32)	Q8	Q4	Default (FP32)
Staff baseline	–	–	0.882	–	–	0.856
Full fine-tuning	all	–	0.889	–	–	0.861
LoRA	attention	4	0.811	–	–	–
LoRA	attention	8	0.799	0.796	0.798	–
LoRA	attention	16	0.810	–	–	–
LoRA	attention	32	0.809	0.805	0.806	0.731
LoRA	attention	64	0.799	–	–	–
LoRA	attention	128	0.809	0.804	0.803	–
LoRA	mlp	4	0.820	–	–	–
LoRA	mlp	8	0.828	0.820	0.818	0.72
LoRA	mlp	16	0.823	–	–	–
LoRA	mlp	32	0.824	0.816	0.818	–
LoRA	mlp	64	0.824	–	–	–
LoRA	mlp	128	0.824	0.821	0.822	–

Table 2: Comparison of LoRA and quantization accuracy across rank and layer configurations.

Sonnet Generation: Table 3 presents the LoRA and quantization results on sonnet generation. In contrast to paraphrase detection, LoRA nearly closes the gap with full fine-tuning on this task. Attention LoRA at rank 32 achieves 42.054 chrF, within 0.2 points of the full fine-tuning best of 42.213. MLP LoRA at rank 4 achieves 41.604, only 0.6 points below full fine-tuning. Both results exceed the staff baseline of 41.414. At $lr=1e-4$ (matching full fine-tuning), LoRA configurations plateau around 39-42 chrF across both layer types and all ranks. Performance only approaches full fine-tuning at $lr=1e-3$. For reference, MLP LoRA rank 4 drops from 41.604 at $lr=1e-3$ to 34.721 at $lr=1e-5$, a 6.9 point difference. This learning rate sensitivity is unique to sonnet generation; in contrast to paraphrase detection, LoRA performs best at the same learning rate as full fine-tuning. As with paraphrase detection, rank has minimal impact within a given configuration. At $lr=1e-4$, attention LoRA ranges from 39.0 to 40.3 and MLP LoRA from 39.6 to 40.7 across ranks 4 to 128. MLP LoRA outperforms attention LoRA at matched learning rates, consistent with the paraphrase detection results, though the gap is narrower 1. Quantization effects on sonnet generation are noisier than on paraphrase detection. Most configurations show small degradation under 8-bit and 4-bit quantization. We attribute the quantization score variance to the small training and testing sets. Overall, quantization does not systematically degrade performance on either task.

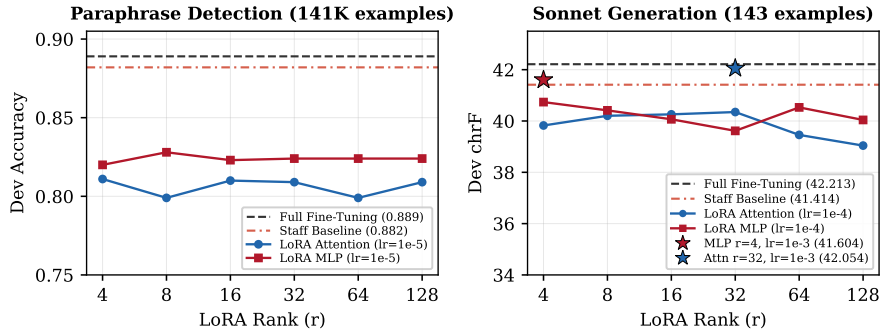


Figure 1: Paraphrase vs Sonnet Generation Performance across LoRA ranks

Method	Layer	Rank	LR	Dev chrF			Test chrF	
				Default (FP32)	Q8	Q4	Default (FP32)	
Staff baseline	—	—	—	41.414	—	—	41.208	
Full fine-tuning	all	—	1e-6	35.64	—	—	—	
Full fine-tuning	all	—	1e-5	41.597	—	—	—	
Full fine-tuning	all	—	1e-4	42.213	—	—	41.948	
Full fine-tuning	all	—	1e-3	41.122	—	—	—	
LoRA	attention	4	1e-4	39.825	—	—	—	
LoRA	attention	8	1e-4	40.203	40.644	40.609	—	
LoRA	attention	16	1e-4	40.259	—	—	—	
LoRA	attention	32	1e-4	40.349	40.876	39.491	—	
LoRA	attention	64	1e-4	39.458	—	—	—	
LoRA	attention	128	1e-4	39.041	37.372	41.051	—	
LoRA	attention	32	1e-5	38.63	—	—	—	
LoRA	attention	32	1e-3	42.054	41.484	41.558	41.771	
LoRA	attention	32	1e-2	41.906	—	—	—	
LoRA	mlp	4	1e-4	40.737	38.313	38.736	—	
LoRA	mlp	8	1e-4	40.414	37.481	40.320	—	
LoRA	mlp	16	1e-4	40.067	—	—	—	
LoRA	mlp	32	1e-4	39.616	39.605	39.544	—	
LoRA	mlp	64	1e-4	40.530	—	—	—	
LoRA	mlp	128	1e-4	40.041	40.081	38.986	—	
LoRA	mlp	4	1e-5	34.721	—	—	—	
LoRA	mlp	4	1e-3	41.604	41.269	41.837	41.527	
LoRA	mlp	4	1e-2	38.934	—	—	—	

Table 3: Sonnet Generation Results (GPT-2, 10 epochs, batch size 8, temp 1.2, top_p 0.9)

6 Analysis

6.1 The impact of Dataset Size

The most striking result is the asymmetry in LoRA’s performance gap: 6.1 points on paraphrase detection but less than 1 point on sonnet generation. Surprisingly, sonnet generation is more structurally demanding yet shows less degradation. However, the two tasks differ in data scale: 141k vs 143 training examples.

With only 143 training examples, full fine-tuning of a 124M parameter model cannot make meaningful use of its full parameter space. The learned weight update is effectively low-rank regardless of whether we impose a rank constraint. With 141k examples, the model can leverage its capacity, and the rank restriction becomes more constraining. This aligns with the intrinsic dimensionality work of Li et al [10] and Aghajanyan et al. [5], and suggests that the effective dimensionality of fine-tuning is bounded by available data, not just task complexity. This observation is practically relevant because LoRA is most commonly applied to narrow, domain-specific tasks where large labeled datasets are scarce, precisely the scenario where our results suggest the model performs closest to full fine-tuning.

The learning rate behavior across tasks supports this interpretation. On paraphrase detection, LoRA performs best at the same learning rate as full fine-tuning (1e-5). On sonnet generation, MLP LoRA rank 4 jumps from 34.721 at lr=1e-5 to 41.604 at lr=1e-3. With fewer examples, each gradient step covers more of the dataset and requires correspondingly larger updates. Notably, LoRA at lr=1e-3 outperforms full fine-tuning at the same rate (41.122), indicating that the optimal hyperparameters for LoRA and full fine-tuning do not necessarily coincide and may diverge further in low-data regimes.

6.2 Qualitative Analysis of Sonnet Generation

Line	Original (Shakespeare)	Full Fine-Tuning (chrF 42.213)	LoRA MLP r=4 lr=1e-3 (chrF 41.604)	LoRA MLP r=4, Q4 lr=1e-3 (chrF 41.837)
1	<i>When my love swears that she is made of truth,</i>	<i>When my love swears that she is made of truth</i>	<i>When my love swears that she is made of truth</i>	<i>When my love swears that she is made of truth</i>
2	<i>I do believe her, though I know she lies,</i>	<i>I do believe her though I know she lies,</i>	<i>I do believe her though I know she lies,</i>	<i>I do believe her though I know she lies,</i>
3	<i>That she might think me some untutorèd youth,</i>	<i>That she might think me some untutored youth,</i>	<i>That she might think me some untutored youth,</i>	<i>That she might think me some untutored youth,</i>
4	Unlearnèd in the world's false subtleties.	That I know from experience that she is buried,	Each true and equal to her dear dearest.	Born unknown and deified to unknown fathers,
5	Thus vainly thinking that she thinks me young,	And that she may much delight in praise.	As far as no earthly fruit taint both grace obtain	Beautiful blood haired, clad in princesses, wove;
6	Although she knows my days are past the best,	Madam, in truth her beauty lies,	Than those females that lackey did, sweet coils showered.	But as stony degrees are disfigured in love,
7	Simply I credit her false-speaking tongue:	She swears she was with you when I used to love her,	I should like much so to my subject as rich is	And sharp sets of bird callings win hymns,
8	On both sides thus is simple truth suppressed.	When my love told me this to her,	To assume a widow's hand where my heart do clot	And hot acorn clouds are still the higher
9	But wherefore says she not she is unjust?	And ahoy she says my words true,	When my sweetkeeper once imagined me the beardless;	And starry night brighter than when I saw you
10	And wherefore say not I that I am old?	If I too like decease her of my love.	For rare linen elastics deceive and hence die;	As virgin winter might have kept fire red
11	Oh, love's best habit is in seeming trust,	Madam, be false to me, thou hast no love but my name;	They let copies of royalty for cards stay	To kill what bore, or date female flower no longer.
12	And age in love loves not to have years told.	And before I'll pity her no more I tell,	To sullied women illustrate amorous lack;	And a thousand kisses are ours everywhere, yet more woe still;
13	Therefore I lie with her and she with me,	And keep quiet when she proves swerving,	And since old wrinkles mocking me do my wish,	Or my love shall yet mine evermore ashes now live.
14	And in our faults by lies we flattered be.	For fear of disgrace in becoming true.	When I earliest aspired for such a wife	

Table 4: Sonnet 6: Line-by-line comparison of generated continuations. Lines 1–3 (italicized) are the shared prompt. Bolded lines highlight key qualitative differences discussed in Section 5.3. Full fine-tuning maintains thematic coherence around the lover’s dishonesty. LoRA produces comparable structure with looser semantic coherence. Q4 quantization preserves structure but introduces vocabulary drift from Shakespearean register.

Table 4 presents a line-by-line comparison of Sonnet 6 continuations across configurations alongside the original Shakespeare. All models receive the first three lines as a prompt. Full fine-tuning stays on topic with the prompt’s subject of lying in love. Line 6 directly references truth and beauty, and the couplet (lines 13-14) wraps up the idea. All properly-tuned configurations produce outputs within one line of the 14-line target.

LoRA at lr=1e-3 maintains sonnet structure and produces language that reads as Old English, but the content is less focused. Lines 10-12 are individually reasonable phrases, but they don’t clearly relate to one another or the prompt’s theme. Lines 13-14 exist but do not function as a couplet; they are not indented, do not rhyme, and read as a continuation rather than a closing pair.

The Q4 quantized model preserves structure but the word choices shift noticeably. Lines 7-8 are a clear example: "bird callings" and "hot acorn clouds" are out of place. 8-bit quantization shows a similar but weaker effect. This indicates that reduced precision in the frozen weights affects the **word choice** of the generated text while leaving the structure intact. Attention LoRA at Q4 shows a more severe version of this effect, rather than drifting toward modern vocabulary it generated words such

as "*imagoyles*" and "*salumript*". In both cases, the degradation in word choice and stylistic difference is not reflected well in the chrF scores.

6.3 LoRA Target Layer Selection

MLP LoRA outperforms attention LoRA on both tasks, by roughly 2 points on paraphrase and 0.5-1 point on sonnet at equivalent learning rates. At the same rank, MLP LoRA trains about 2.5x more parameters because the feed-forward matrices are larger than the attention projections. But parameter count does not fully explain this: on paraphrase, MLP LoRA at rank 4 scores 0.820, outperforming attention LoRA at rank 16 at 0.810 despite training fewer parameters. This suggests MLP layers are a more effective adaptation target across both data set scales. This finding validates our decision in Section 3 to extend LoRA beyond the attention-only configuration of Hu et al. [3] to include MLP layers, motivated by Schulman et al. [9] who show that attention-only LoRA underperforms full fine-tuning and that including MLP layers helps close this gap.

6.4 Rank and Quantization

Performance is flat across ranks 4 to 128 on both tasks, consistent with the low intrinsic rank hypothesis from Hu et al. [3]. Our results extend this to a generation task evaluated on structural metrics. A denser sweep would better characterize the impact of rank.

Quantization has no systematic quantitative impact, less than 1 point on paraphrase, and is noisy but neutral on sonnet results. Dettmers et al. [6] similarly find that quantization preserves performance on benchmarks such as GLUE. Our quantitative results on both accuracy and chrF are consistent with this finding. However, our qualitative analysis in Section 5.3 reveals that quantization introduces stylistic drift in generated text that these aggregate measures could not capture, suggesting that quantization effects may be more nuanced than prior evaluations indicate.

7 Conclusion

We investigated whether LoRA and quantization exhibit task-dependent performance degradation across paraphrase detection and sonnet generation. LoRA’s performance gap relative to full fine-tuning varies dramatically across tasks, 6.1 points on paraphrase detection, but less than 1 point on sonnet generation. We attribute this primarily to dataset size rather than task structure: with only 143 sonnet training examples, full fine-tuning cannot exploit its full parameter space, making the low-rank constraint negligible. With 141k paraphrase examples, the constraint becomes more restrictive. While we set out to test whether structurally demanding tasks degrade more under LoRA, the difference between task structure and dataset size prevents us from isolating either factor. Our results suggest that the effective intrinsic dimensionality of fine-tuning is bounded by available data, which is practically relevant since LoRA is most often applied in low-data settings. We additionally find that applying LoRA to MLP layers consistently outperforms attention-only LoRA on both tasks. Quantization preserves quantitative performance on both tasks, though qualitative analysis reveals subtle stylistic drift in generated text that current experiment metrics do not capture.

Our work has several limitations. The small scale of the sonnet set limits the statistical reliability of chrF comparisons. Our rank sweep uses powers of two, and a denser sweep would better characterize performance. Finally, all experiments use GPT-2, and these findings may not generalize to larger models. Preliminary qualitative analysis of paraphrase errors suggested disagreements were driven more by LoRA target layer than quantization, but this was inconclusive due to our focus on sonnet structural/qualitative evaluation. A deeper investigation into paraphrase and quantization error patterns is a possible extension of this work.

Team contributions (Required for multi-person team)

The code, evaluations, and report had equivalent contributions from both team members.

Peter Benitez: Ran experimental configurations including hyperparameter sweeps, LoRA rank, learning rate searches, and quantization evaluations. **Fabio Ibanez:** Integrated LoRA using loralib and quantization using bitsandbytes ontop of the core default final project GPT-2 transformer layers.

References

- [1] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. *arXiv preprint arXiv:1902.00751*, 2019.
- [2] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.
- [3] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.
- [4] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. *arXiv preprint arXiv:1804.08838*, 2018.
- [5] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020.
- [6] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, 2023.
- [7] Zhengbo Wang, Jian Liang, Ran He, Zilei Wang, and Tieniu Tan. Lora-pro: Are low-rank adapters properly optimized? In *International Conference on Learning Representations (ICLR)*, 2025.
- [8] Reece Shuttleworth, Jacob Andreas, Antonio Torralba, and Pratyusha Sharma. Lora vs full fine-tuning: An illusion of equivalence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.
- [9] John Schulman et al. LoRA without regret. <https://thinkingmachines.ai/blog/lora/>, September 2025. Thinking Machines Blog. Accessed: 2026-03-07.
- [10] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don’t know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, 2018.

A Appendix (optional)

A.1 Sonnet Generation Full Fine-tuning hyper parameter search

Learning Rate	Batch Size	Temperature	Dev chrF
1e-6	8	1.2	35.640
1e-5	8	1.2	41.597
1e-5	8	1.0	40.997
1e-5	1	1.2	42.052
1e-5	1	1.0	41.822
1e-4	8	1.2	42.213
1e-4	1	1.2	41.081
1e-4	1	1.0	41.956
1e-3	8	1.2	41.122

Table 5: Sonnet Generation Full Fine-Tuning Hyperparameter Sweep (GPT-2)