PEFT for GPT-2 on Paraphrase Detection and Sonnet Generation

Aniket Gupta Anjani Pangal Mallika Parulekar

Computer Science, Stanford University

Motivation and Problem Setup

Full fine-tuning is effective but expensive: every downstream task requires updating the entire model. We ask whether **parameter-efficient fine-tuning (PEFT)** can recover most of the benefit of full fine-tuning for a semantic task, and how much a low-resource generation task can be improved through better decoding and training heuristics.

We adapt GPT-2 to two tasks:

- **Paraphrase detection:** given two questions, predict whether they express the same meaning.
- **Sonnet generation:** condition on the first 3 lines of a Shakespearean sonnet and generate the continuation.

Tasks, Data, and Evaluation

Task	Data	Metric
Sentiment sanity check	SST, CFIMDB	Dev accuracy
Paraphrase detection	Quora pairs	Dev accuracy, macro-F1
Sonnet generation	Shakespeare + filtered Deep-speare	Dev chrF

Key data details

- Quora paraphrase subset: 141,506 train / 20,215 dev / 40,431 test.
- Deep-speare filtered to remove Shakespearean sonnets: 2664 train / 332 valid.

Paraphrase Detection Approach

We cast paraphrase detection as **cloze-style classification**. GPT-2 encodes a prompt built from question pair (q_1, q_2) and the final hidden state predicts the binary label:

$$\hat{y} = \text{softmax}(W_{\text{cls}} h_T + b_{\text{cls}}), \qquad \mathcal{L}_{\text{para}} = -\sum_i \log p_\theta(y_i \mid q_{1,i}, q_{2,i}).$$

PEFT methods

- **LoRA:** learn a low-rank update on frozen weights

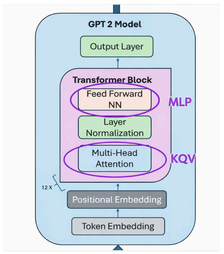
$$W = W_0 + \Delta W, \qquad \Delta W = \frac{\alpha}{r} BA.$$

- **DoRA:** decompose pretrained weights into *magnitude* and *direction*, then adapt:

$$m = \|W_0\|_c, \qquad V = \frac{W_0}{\|W_0\|_c}, \qquad \Delta V = \frac{\alpha}{r} BA.$$

$$W' = m \cdot \frac{V + \Delta V}{\|V + \Delta V\|_c}.$$

Adapter placement: Motivated by evidence that LoRA benefits from broader placement [3], we compare adapters on *KQV* only versus *KQV+MLP*.



Baselines

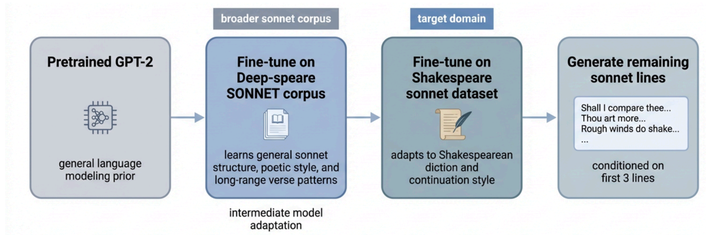
- **MLP-only:** freeze GPT-2, train only the classifier head.
- **Full fine-tuning:** update all model parameters.

Sonnet Generation Approach

For sonnet generation, we use **full fine-tuning** rather than PEFT. Each training example is a sonnet; at inference time, GPT-2 is conditioned on the first 3 lines and generates the rest autoregressively.

We compare:

- Top-p sampling vs. beam search
- Repetition-aware decoding
- Weight decay + gradient clipping
- Two-stage training via Deep-speare → Shakespeare



Idea: intermediate sonnet-domain training can help the model learn poetic structure before adapting to Shakespeare.

Paraphrase Results

Method	Trainable %	Dev Acc
MLP prediction head only	–	0.639
Full fine-tuning	100%	**0.898**
LoRA: KQV only	2.83%	0.875
LoRA: KQV+MLP	7.09%	0.882
DoRA: KQV only	2.85%	0.871
DoRA: KQV+MLP	7.14%	0.882

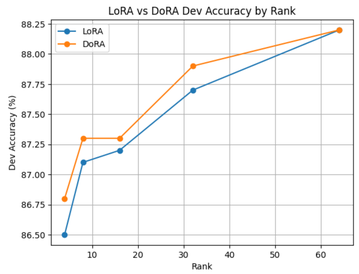
Table 1. Result summary table across full-finetuning, PEFT and MLP prediction head only fine-tuning

Rank	LoRA KQV	LoRA Δ w/ MLP	DoRA KQV	DoRA Δ w/ MLP
4	84.8	+1.7	84.9	+1.9
8	85.1	+2.0	85.2	+2.1
16	86.2	+1.0	86.1	+1.2
32	86.7	+1.0	86.8	+1.1
64	87.5	+0.7	87.1	+1.1

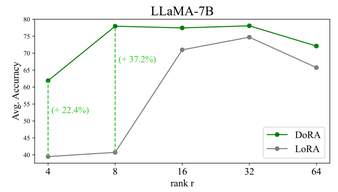
Table 2. Improvement in dev accuracy from adding MLP adaptation to Q,K,V adapters. Δ denotes the gain relative to adapting Q,K,V alone.

Takeaways:

- PEFT recovers most of full fine-tuning performance.
- Adapting *KQV+MLP* performs better than *KQV* alone.
- DoRA shows a small advantage over LoRA at lower ranks.



LoRA vs. DoRA across adaptation ranks on Quora dataset.



LoRA vs. DoRA across adaptation ranks on original paper for varying ranks for LLaMA-7B on the commonsense reasoning tasks.

Sonnet Results

Method	Dev chrF
Full-finetuning (baseline)	37.57
Top-p (milestone)	41.34
Beam (milestone)	41.65
Beam + repetition penalty	43.05
Top-p + repetition penalty	43.67
Deep-speare → Shakespeare (with repetition penalty)	43.58
GPT-2 large exploratory run	43.01

Takeaway:

- **Repetition-aware decoding** was the single biggest improvement.
- Top-p slightly outperformed beam search in the final setting.
- Larger models did not beat the best GPT-2 small when trained for the same number of epochs

Analysis

Paraphrase detection

- Updating **KQV+MLP** improved performance, matching insights from Schulman guest lecture
- DoRA helped slightly more at lower ranks, but not as much as in original DoRA paper. This is likely due to the lower difficulty of the paraphrase task, that does not exhaust model capacity.
- Ensemble routing between LoRA and DoRA via Jaccard similarity yielded negligible gains (+8/40K examples), suggesting similar learned decision boundaries.

Sonnet generation

- **Repetition penalty** was the single most important change, which makes sense due to GPT's tendency to get stuck in repeating loops given its autoregressive structure.
- Deep-speare pretraining was neither particularly helpful nor harmful. This could be due to the different sonnet structure and GPT-2's existing language capabilities

Qualitative Sonnet Example

Before repetition penalty

For all thy fading sweets and all thy fading sweets,
And all thy fading beauty's forlorn
For all thy fading sweets and all thy fading sweets...

After repetition penalty

Thou art as fair and dearest of all living things; yet for thyself so poor is best acquainted
That every word shall be mine own sweet self-loving spite...

Observation: repetition-aware decoding reduces **local loops** and produces more varied continuations, though global coherence is still imperfect.

References

[1] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi.
The curious case of neural text degeneration.
In *International Conference on Learning Representations*, 2020.

[2] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen.
Dora: Weight-decomposed low-rank adaptation.
arXiv preprint arXiv:2402.09353, 2024.

[3] John Schulman and Thinking Machines Lab.
Lora without regret.
Thinking Machines Lab: Connectionism, 2025.
https://thinkingmachines.ai/blog/lora/.