Reasoning Supervised Fine-Tuning of Large Language Models (LLMs)

Team Members:

Ashley Ferraro | Avanthika Rajesh | Ashwin Shanmugasundaram | Gayathri Mahendran

Abstract

This project explores improving the reasoning abilities of large language models (LLMs) through supervised fine-tuning (SFT). Using the Qwen2.5-3B-Instruct model, we evaluated performance on four benchmarks: AIME24, AIME25, MATH-500 and MMLU-Redux2. All experiments were conducted on Virginia Tech's ARC system. The results showed measurable gains after fine-tuning, indicating that SFT is an effective approach for enhancing structured and mathematical reasoning in compact LLMs.

1. Introduction

Instruction-tuned language models perform well on general comprehension and generation tasks but often struggle with multi-step reasoning and symbolic logic. This project aims to enhance these capabilities using supervised fine-tuning on reasoning-oriented datasets.

We adopted Qwen2.5-3B-Instruct as the base model and followed the official project framework provided in the course repository. The workflow included three main stages:

1. Running baseline evaluations across four reasoning benchmarks.

2. Fine-tuning the model on two datasets, Diverse15k and Random15k.

3. Re-evaluating the fine-tuned model to compare with baseline performance.

All experiments were carried out using ARC for compute and storage.

2. Datasets

Fine-Tuning Datasets

- **Diverse15k:** A diverse set of instruction-response pairs covering reasoning, mathematics, and analytical prompts.

- **Random15k:** A randomized subset used to evaluate generalization and prevent overfitting.

Evaluation Benchmarks

- **AIME24:** Math reasoning problems emphasizing step-by-step solution accuracy.

- **AIME25:** A follow-up benchmark with symbolic reasoning variations.

- **MATH-500:** Complex numerical and symbolic reasoning problems.

- **MMLU-Redux-2:** A broad benchmark combining logic, science and analytical reasoning.

All datasets followed the standard input-output formatting supported by LightEval.

3. Baseline Evaluation

The baseline Qwen2.5-3B-Instruct model was first evaluated across all four benchmarks to establish reference accuracy levels.

Benchmark	Baseline	Source File
AIME24	0.0667	baseline_qwen25_3b.csv / baseline_qwen25_3b_math_sci.csv
AIME25	0.0333	baseline_qwen25_3b.csv
MATH-500	0.6700	baseline_qwen25_3b_math_sci.csv
MMLU-Redux-2	0.6393	baseline_qwen25_3b_mmlu.csv

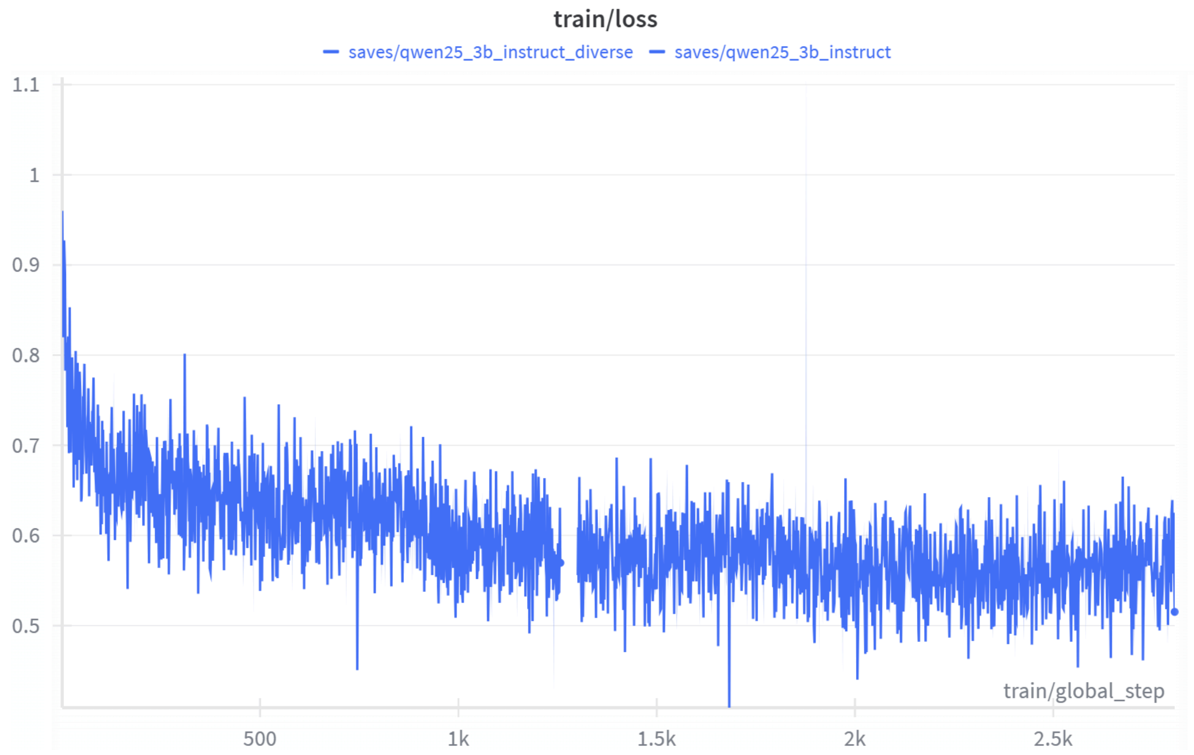
4. Fine-Tuning Process

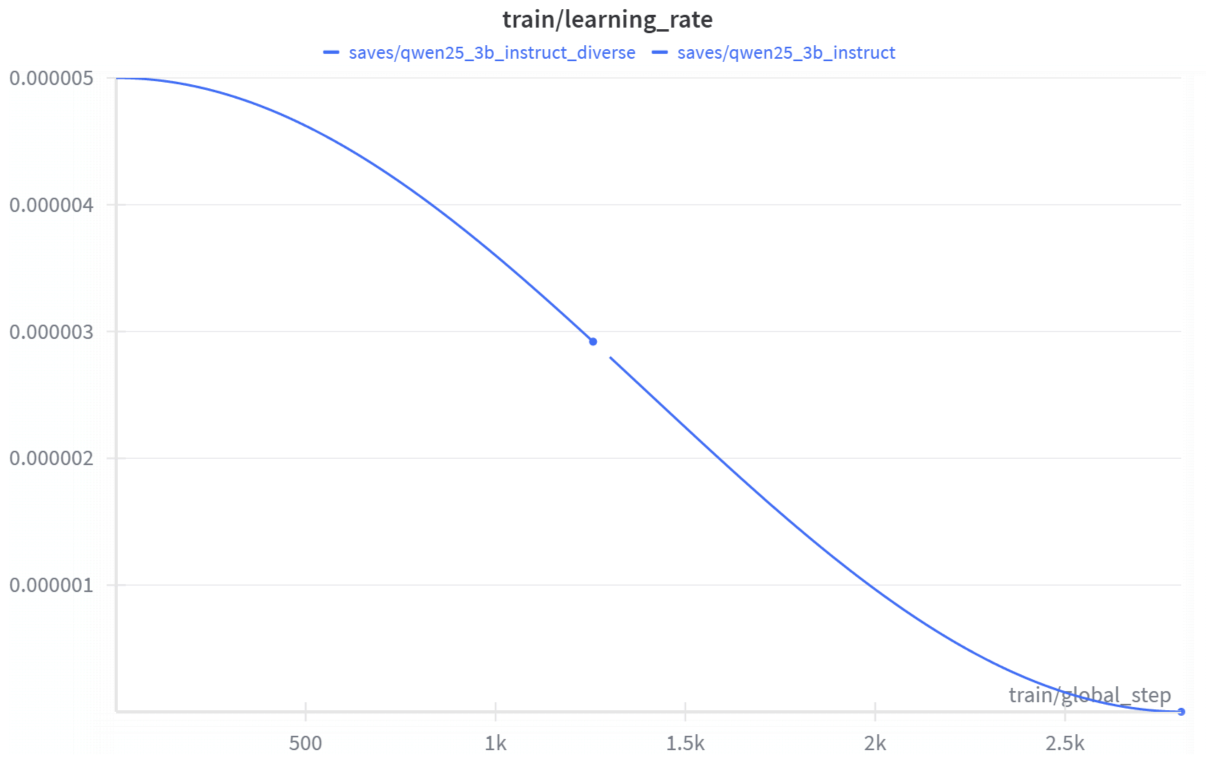
Fine-tuning was conducted on ARC using the reasoning-focused datasets mentioned above. The objective was to strengthen reasoning consistency, logical coherence, and structured output generation. Model checkpoints and training logs were saved within the project workspace, and the overall training process remained stable throughout. No abnormalities were observed during the training process as the loss reduced linearly for each step with a variation of +/- 0.5 for both datasets. No notable difference in training metrics were observed for both the 15K random dataset and the 15K diversity dataset. We utilized the default hyperparameters for training the model. These default hyperparameters are shown below. We did not vary the hyperparameters due to model and time constraints, however as we'll show later on these hyperparameters were successful in improving the model performance through fine tuning.

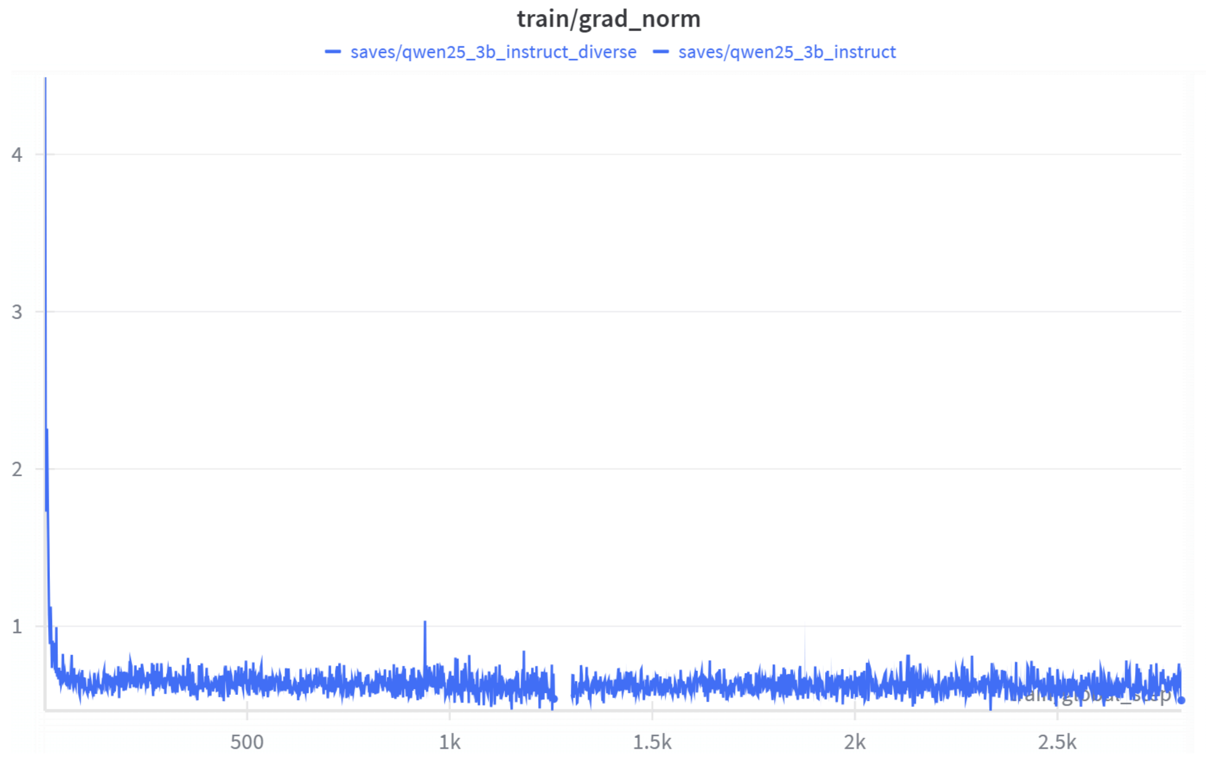
Hyperparameter Values for Both Random and Diversity Models	
Learning Rate	5.0e-6
Training Epochs	3
Per Device Train Batch Steps	1
Gradient Accumulation Steps	8

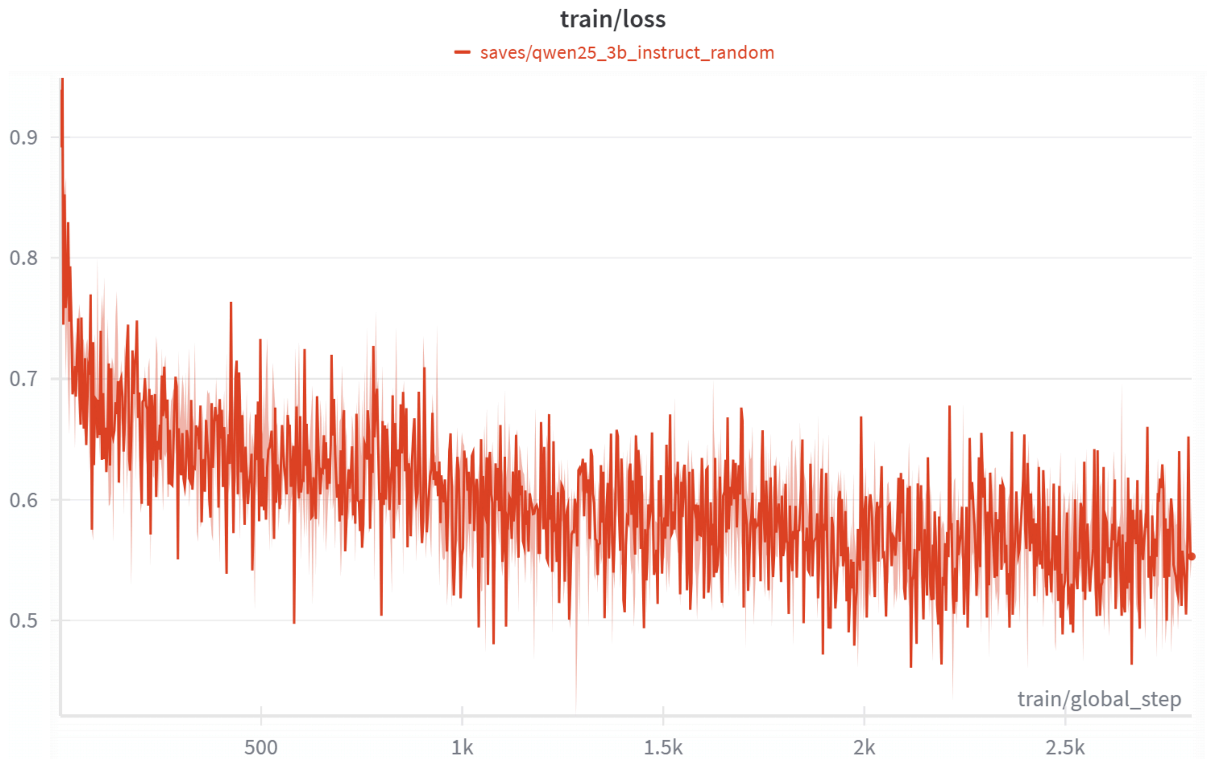
The baseline accuracies were computed using the pretrained Qwen2.5-3B-Instruct model before any fine-tuning. These serve as reference points for evaluating improvement.

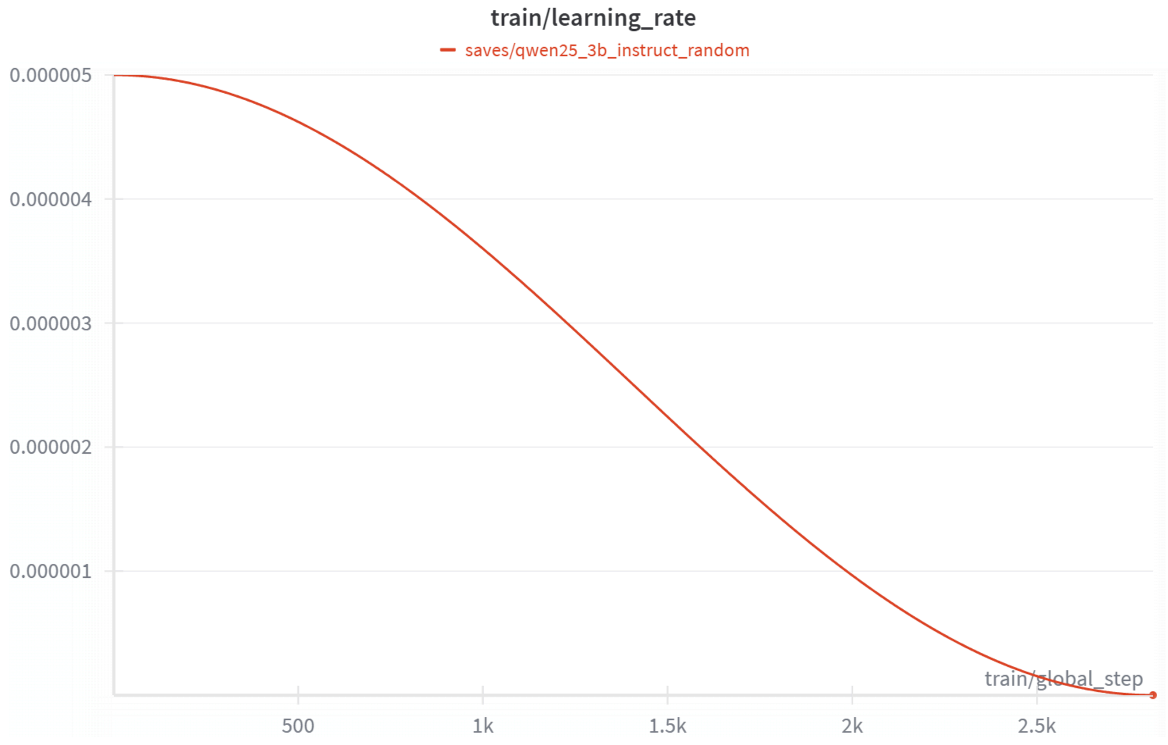
The loss, training rate, and gradient norm for each step in the training process is shown below wherein the blue figures represents the diversity model metrics and the red figures represent the random model.

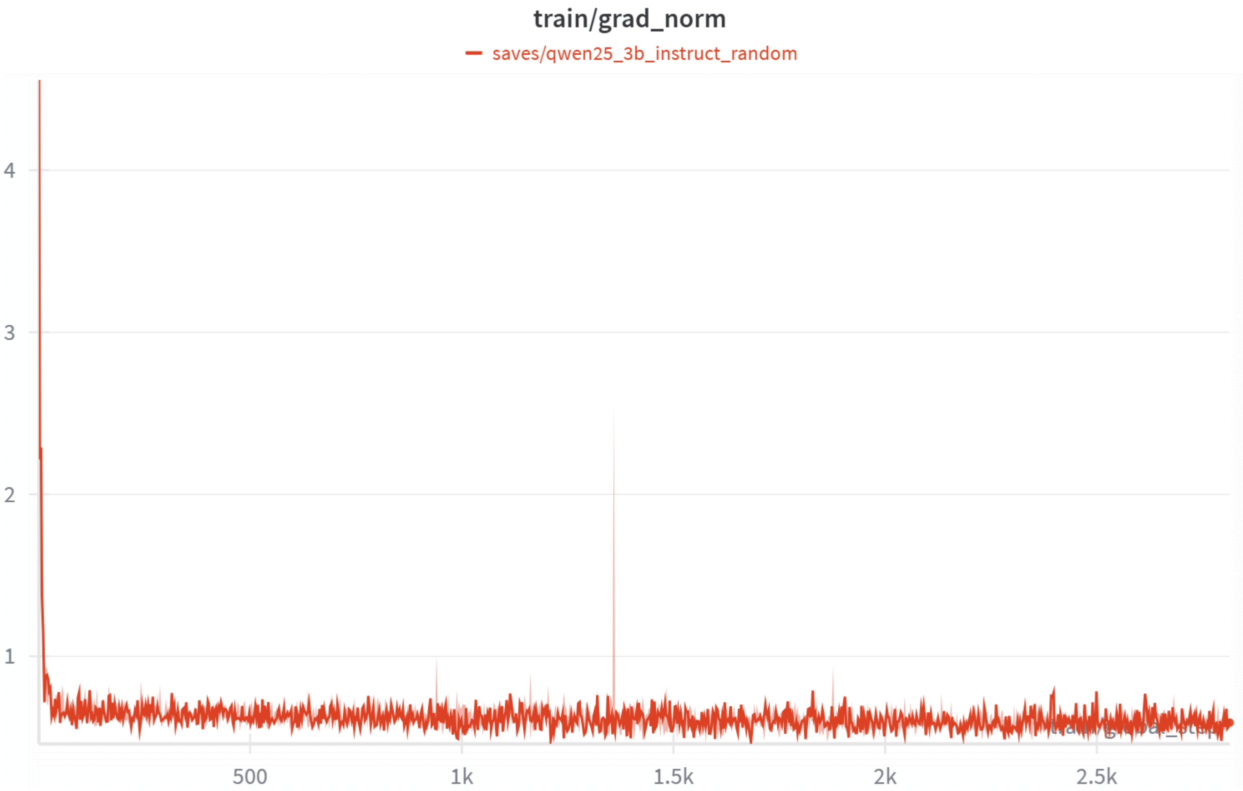



train/learning_rate
saves/qwen25_3b_instruct_diverse
saves/qwen25_3b_instruct
0.000005
0.000004
0.000003
0.000002
0.000001
500
1k
1.5k
2k
2.5k
train/global_step


train/grad_norm
saves/qwen25_3b_instruct_diverse
saves/qwen25_3b_instruct
4
3
2
1
500
1k
1.5k
2k
2.5k
train/global_step


train/loss
saves/qwen25_3b_instruct_random
0.9
0.8
0.7
0.6
0.5
train/global_step
500
1k
1.5k
2k
2.5k


train/learning_rate
saves/qwen25_3b_instruct_random
0.000005
0.000004
0.000003
0.000002
0.000001
train/global_step
500
1k
1.5k
2k
2.5k



5. Fine-Tuned Evaluation

After fine-tuning, both Random15K and Diverse15K models generally showed improved reasoning performance compared to the baselines. Three out of four benchmarks exhibited positive gains, while one (MMLU-Redux-2) showed a small decrease, likely due to domain specialization. Between the two, the Diverse15K dataset produced slightly higher improvements overall.

15K Random Dataset			
Benchmark	**Baseline**	**Fine-Tuned**	**Improvement**
AIME24	0.0333	0.1000	0.0667
AIME25	0.0000	0.0333	0.0333
MATH-500	0.4500	0.6260	0.176
MMLU-Redux-2	0.6412	0.2825	-0.3587

15K Diversity Dataset			
Benchmark	**Baseline**	**Fine-Tuned**	**Improvement**
AIME24	0.0333	0.1000	0.0667
AIME25	0.0000	0.0667	0.0667
MATH-500	0.4500	0.6500	0.200
MMLU-Redux-2	0.6412	0.2977	-0.3435

6. Result Interpretation

Overall, the fine-tuned models performed better than the baselines on most benchmarks, showing that the additional training helped improve reasoning ability. We saw steady gains across three of the four tasks, suggesting that the model was able to adapt well to the fine-tuning data. The one benchmark that dropped slightly in performance could be due to mild overfitting-where the model became too specialized on math-style reasoning and lost some general knowledge flexibility.

Interestingly, the training curves for both datasets looked almost identical. This means the model converged in a very similar way for both random and diverse data, which is a good sign of stability. Between the two datasets, the diverse one gave a small but consistent edge, probably because it exposed the model to a wider range of reasoning styles and problem types.

7. Conclusion

This project gave us a clear look at how supervised fine-tuning can strengthen a model's reasoning skills. Even with a smaller model like Qwen2.5-3B-Instruct, fine-tuning on curated data noticeably improved performance on structured reasoning benchmarks. Although there were a few trade-offs - like the slight drop on one evaluation - the overall trend was positive. Moving forward, it would be interesting to explore larger datasets or new training methods, such as reinforcement learning from reasoning feedback, to see if we can push these improvements even further.