Fine Tuning: From Benign to Malign
How Different Fine Tuning Methods Degrade Safety Guardrails in LLMs During Benign Finetuning

Manon Kempermann, Max Ramackers

DSAI Project Seminar Summer Semester 2025

Abstract

Finetuning has become a popular choice to customize LLMs for specific tasks while leveraging the natural-language processing capabilities of the base model. Recent work has shown however that even under harmless finetuning safety degradations in the resulting models can be observed [1, 2], but so far these observations are limited to specific tasks and do not consider the influence of the fine-tuning method itself. In this work we present a case study comparing safety degradation in Llama3.2-1B-Instruct across full-finetuning, LoRA, QLoRA and DPO under fine-tuning for translation and further investigate the impact of the learning rate. We find differences in safety degradation across methods to varying extents and note that higher learning rates lead to a higher degree of degradation. Our findings encourage deeper analyses of the impact of different methods and hyper-parameters on the safety in fine-tuned models to ultimately prevent end-user harm.

1 Introduction

As LLMs have become more popular in recent years, especially since the introduction of ChatGPT, so has the practice of fine-tuning these models for specific tasks. Leveraging the natural language processing capabilities of these models for tasks such as machine translation, coding, classification and summarization has shown great success and several different methods have been developed. However, at the same time concern about the safety of these models have risen. While general chat models are usually safety-tuned to refuse harmful requests these days, recent work showed alarming evidence that fine-tuning can degrade these safe-guards even under benign finetuning [1, 2], which refers to finetuning on data that does not contain anything harmful and does not intentionally teach the model harmful behavior. In this work we aim to explore this phenomenon more closely by providing a comparison of four different fine-tuning methods and their effect on the safety degradation.

Understanding which impact each finetuning method has on the safeguards under benign finetuning can help developers prevent unintended harmfulness in their fine-tuned models and can give empirical insights into strengths and weaknesses of each method with regard to safety.

Given the compute-resource constraints for this project we narrow our focus on the case study of fine-tuning for translation for a single model, Llama3.2-1B-instruct, and

additionally investigate the effect of different learning rates for each method as this is probably one of the most important hyper-parameters developers need to choose when finetuning a model that could vastly impact both performance and safety. We thus aim to answer these research questions:

- **RQ1:** How do different finetuning methods compare in their ability to preserve safeguards during finetuning for translation?
- **RQ2:** How does the choice of the learning rate impact the trade-off between translation performance and safeguard preservation?

In the following, we will lay out existing research on finetuning and safety degradation, followed by our experiments showing that indeed differences between methods can be observed to varying extents and how higher learning rates lead to worse safety outcomes.

2 Related Work

Harmful fine-tuning attacks. The vulnerability of safety-aligned LLMs to adversarial fine-tuning has been demonstrated through several works. Zhan et al. [3] showed that fine-tuning GPT-4 with as few as 340 adversarially designed examples can remove RLHF protections with 95% success rate while maintaining model usefulness. Yang et al. [4] introduced "Shadow Alignment," demonstrating that just 100 malicious examples are enough to subvert safety-aligned models. These attacks reveal that even the most advanced models remain susceptible to deliberate safety circumvention through fine-tuning.

Benign fine-tuning safety degradation. More concerning for practitioners is the unintended safety degradation during benign fine-tuning. Qi et al. [1] first documented that fine-tuning on seemingly safe datasets can inadvertently compromise safety alignment, even when users have no malicious intent. Jan et al. [2] systematically analyzed this phenomenon across multiple tasks, finding that code generation and translation fine-tuning lead to particularly severe degradation. This unintended degradation poses significant challenges for the widespread deployment of fine-tuning services.

Safety preservation methods. Several approaches have been proposed to maintain safety during fine-tuning. Lyu et al. [5] discovered that prompt templates play a crucial role in preserving safety alignment, proposing the "Pure Tuning, Safe Testing" principle where models are fine-tuned without safety prompts but include them at test time. Various defense mechanisms have been developed including alignment-stage solutions like Vaccine [6], fine-tuning-stage approaches such as safety data mixing, and post-fine-tuning remediation methods. However, most defenses are evaluated against specific attack scenarios and show limited robustness across different fine-tuning configurations and hyper-parameters.

Research gaps. While the existence of safety degradation during fine-tuning is well-established, systematic comparisons of how different fine-tuning methodologies affect safety preservation remain limited. Existing studies predominantly focus on full fine-tuning scenarios, with insufficient analysis of how parameter-efficient methods like LoRA [7], QLoRA [8], and preference-based methods like DPO [9] compare in terms of safety retention. Our work addresses this gap through controlled comparative analysis of these prominent fine-tuning approaches under identical experimental conditions.

3 Methods

For our experiments we oriented ourselves on the methodology of Jan et al. [2], but adapted it to fit our resources and needs. We make our code publicly available on GitHub[1].

3.1 Finetuning

3.1.1 Model Choice

We decided to use Llama3.2-1B-Instruct for all our experiments. Llama3.2-1B-Instruct is the smallest safety-tuned model of the Llama3.2 family [10]. To test degradation of the refusal mechanism it is essential that the model initially shows a strong refusal behavior on harmful requests. The 1B model gives us a reasonable trade-off between performance and compute intensity.

3.1.2 Fine-Tuning Methods

We focus on the dominant fine-tuning methods for LLMs used in practice: Full-Finetuning, LoRA [7], QLoRA [8] and DPO [9]. Full-Finetuning refers to self-supervised training updating all weights of the model during training. LoRA and QLoRA in contrast are parameter-efficient methods that only train low-rank adapters on weight matrices, thus reducing the overall amount of trainable parameters vastly while achieving similarly high performance as full-finetuning under self-supervised learning. QLoRA uses a quantized version of the model for even more efficient training. Direct Preference Optimization (DPO) follows a reward-based learning strategy that aligns a model more towards a preferred output while penalizing it for outputs similar to a given rejected output. DPO thus presents a fundamentally different learning paradigm, yet effective for tasks that require nuanced alignment to certain preferences.

3.1.3 Finetuning Dataset

Leaning on the strong finding for translation of Jan et al. [2], we decided to fine-tune on translating roman-script languages to English. We used the MT-Pref dataset from Agrawal et al. [11], filtered for {German, Italian, French, Portuguese, Spanish} to English training examples. Further, we preprocessed the training examples by removing special tokens and instructions from the original prompt such that our final dataset contained 4720 cleaned examples for training. We decided to only use training examples that teach the model to translate to English to ensure that our English safety evaluation functions as intended and the model does not just translate the attack prompt into the other language. For DPO, we train on the full preference samples and as full fine-tuning, LoRA and QLoRA are all self-supervised fine-tuning methods, we only train on the 'chosen' response of the preferences.

3.1.4 Training

We keep hyper-parameters equal across methods and train with an effective batch size of 8. During training, we save an intermediate checkpoint at 25%, 50% and 75% of the training steps for later analysis along the final model.

[1] https://github.com/theaLilott/safety-degradation-finetuning

To investigate the effect of the learning rate on the safety-performance trade-off, we fine-tuned models with a learning rates of 1e-4, 5e-5, 1e-5 and 5e-6 using each method. We only varied the learning rate and kept all other hyper-parameters fixed to isolate the effect of the learning rate. In total, this left us training 4 models per method, 16 in total.

3.2 Evaluations

After training, we evaluated all checkpoints and final models with temperature = 0 on performance and safety.

3.2.1 Performance Evaluation

For performance, we used a hold-out test split of 942 examples from the MT-pref dataset, pre-processed as the training data. On this we determine the chrF metric [12] as this is a standard machine translation metric also suggested by Agrawal et al. [11] for this dataset measuring the F-score statistic for character n-gram matches between the model response and a reference response. Here we use the provided reference responses of the dataset from a gpt-3.5-turbo finetune.

3.2.2 Safety Evaluation

For the safety evaluation, we determine the refusal rate on the AdvBench evaluation from Zou et al. [13] as did Qi et al. [1] in their experiments on benign fine-tuning safety. AdvBench comprises 520 adversary prompts in English. Due to resource constraints, we follow the original AdvBench implementation and use a keyword-based judgment to determine refusal.

4 Results

4.1 Differences between Finetuning Methods

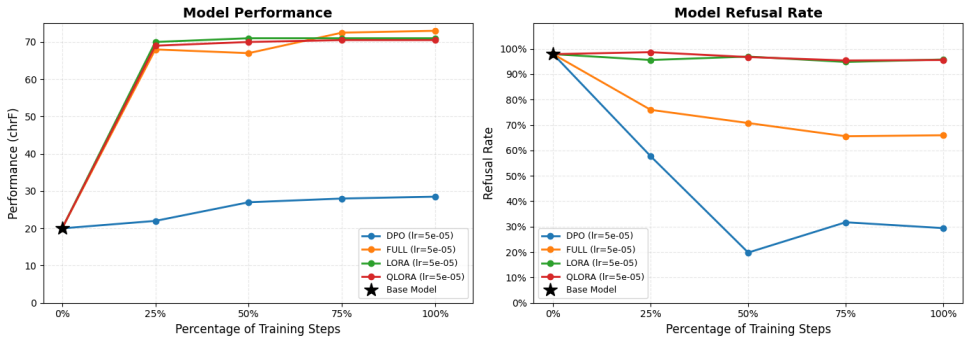

Figure 1: Safety–performance trade-off at a fixed learning rate (5e-5) across all four fine-tuning methods.

We find that there are indeed significant differences between fine-tuning methods regarding their ability to preserve safeguards when keeping all other hyper-parameters fixed, in particular the learning rate at 5e-5, as shown in Figure 1. Except for DPO, our fine-tuned models achieve high task performance. At the same time, we observe striking safety

degradation in full-finetuning compared (∼30% drop in refusal rate) to the parameter efficient methods LoRA and QLoRA (∼1-3% drop in refusal rate). While not achieving high performance, the safety degradation is even worse under DPO (∼55% drop in refusal rate).

We observe under all four methods that the first 25% of training are most crucial for both performance and safety degradation.

We hypothesize that these changes might be due to the fact that LoRA and QLoRA have less impact on the model-internal computational paths that lead to refusal than than DPO and full-finetuning because they change only a tiny fraction of the weights (0.7-2%) compared to training on all weights as in DPO or full-finetuning. However, these fine-tuning methods have in general different hyper-parameter ranges and needs and the results presented here might be too controlled to draw final conclusions on true differences between methods when optimizing for performance.

4.2 Impact of Learning Rates

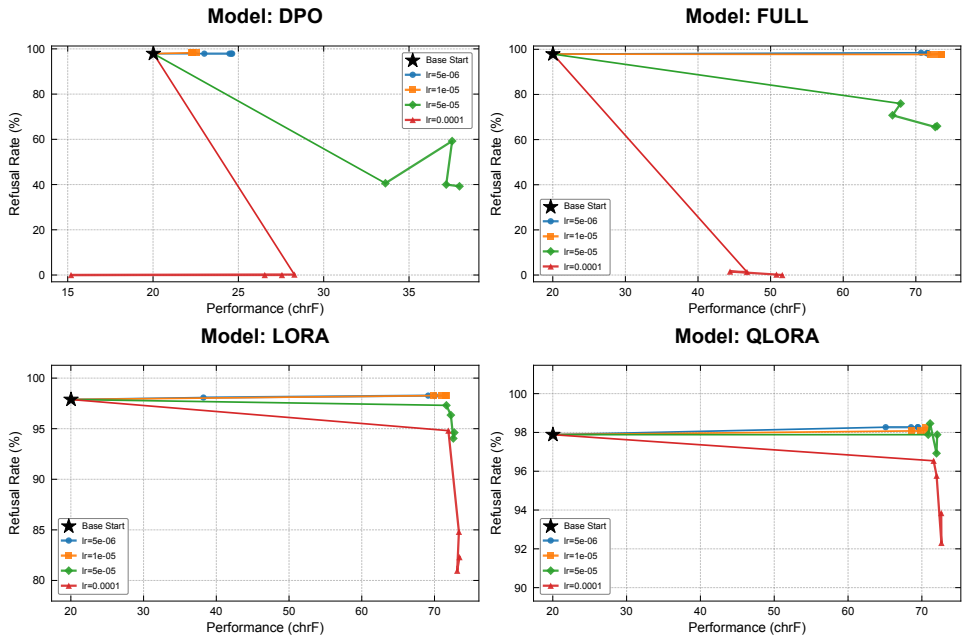

Figure 2: The plot illustrates how increasing learning rates impact each method differently, revealing distinct degradation patterns in safety versus performance.

All methods showed a significant initial improvement in both task performance and safety at low learning rates (5e-6, 1e-5) after the first 25% of fine-tuning training. However, these gains diminished across subsequent training steps as shown in Figure 2. For moderate learning rates (e.g., 5e-5), Full Fine-Tuning, LoRA, and QLoRA exhibited a sharp performance increase, often at the cost of decreasing refusal rates. DPO initially improved performance and safety at low learning rates but degraded substantially at higher learning rates. Here, we clearly observe catastrophic forgetting [14] of the model as performance even degrades over the course of fine-tuning. Lastly, full-finetuning shows similar patterns of high performance and low refusal degradation given low learning rates, but weakened safety preservation under higher learning rates. Overall, this shows that hyper-parameters do play an important role for both performance optimization and safety preservation and

should be optimized not only to enhance performance, but also to keep the refusal rate high.

5 Limitations and Future Work

Our work has several limitations. The constraint on compute resources only allowed for investigation of a 1 billion parameter model, while in practice models of at least 7 billion parameters are used. Results on performance and safety may differ between the size of models.

Our study also concerns only one dataset for one task, namely translation. Studying multiple datasets over a broader variety of tasks could give stronger insights into generalizability of our results ultimately allowing for conclusions and recommendations that would be prematurely stated by just our results. Regarding the impact of hyper-parameters is would be interesting to additionally look at the impact of batch-size and potentially systematically investigate the role of the model size.

Furthermore, our approach of keyword-based refusal detection in the safety evaluation is less robust than strategies such as LLM-as-a-judge and thus may result in false-positives or false-negatives. While we were limited in resources for costly API calls, we encourage deeper safety analysis with more safety benchmarks and alternative evaluation strategies.

Ultimately, all this research can only show empirically degradation. It would however be even more helpful to understand what happens internally in the model that leads to the safety degradation through a perspective of mechanistic interpretability and how we could prevent it.

6 Conclusion

In this work, we aimed to investigate on the case study of translation finetuning, the differences between finetuning methods in preserving safeguards and the effect of the learning rate in each method. We find that indeed there are differences between finetuning methods when it comes to safety preservation. We find the most clear results of safety degradation in LoRA and qLoRA, where we clearly see further degradation as training progresses especially when higher learning rates are used. For DPO and full-finetuning our results are less satisfactory as either we do not observe changes or the model suffers from catastrophic forgetting [14] dependent on the learning rate making inference overall useless. Despite the limitations of our work, it is important to note that ultimately in real world settings developer fine-tune to maximize performance and we and previous work have shown that this can come with major costs in terms of safety – especially for popular and efficient methods like LoRA and QLoRA.

Acknowledgments

We thank our advisor Jan Wehner from CISPA for guidance, feedback and help on the project.

References

[1] X. Qi, Y. Zeng, T. Xie, P.-Y. Chen, R. Jia, P. Mittal, and P. Henderson, "Fine-tuning aligned language models compromises safety, even when users do not intend to!" in *International Conference on Representation Learning*, B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, Eds., vol. 2024, 2024, pp. 30 988–31 043. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2024/file/83b7da3ed13f06c13ce82235c8eedf35-Paper-Conference.pdf

[2] E. Jan, N. Aldahoul, M. Ali, F. Ahmad, F. Zaffar, and Y. Zaki, "Multitask-bench: Unveiling and mitigating safety gaps in LLMs fine-tuning," in *Proceedings of the 31st International Conference on Computational Linguistics*, O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, Eds. Abu Dhabi, UAE: Association for Computational Linguistics, Jan. 2025, pp. 9025–9043. [Online]. Available: https://aclanthology.org/2025.coling-main.606/

[3] Q. Zhan, R. Fang, R. Bindu, A. Gupta, T. Hashimoto, and D. Kang, "Removing RLHF protections in GPT-4 via fine-tuning," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 681–687. [Online]. Available: https://aclanthology.org/2024.naacl-short.59/

[4] X. Yang, X. Wang, Q. Zhang, L. R. Petzold, W. Y. Wang, X. Zhao, and D. Lin, "Shadow alignment: The ease of subverting safely-aligned language models," in *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*, 2024. [Online]. Available: https://openreview.net/forum?id=9qymw6T9Oo

[5] K. Lyu, H. Zhao, X. Gu, D. Yu, A. Goyal, and S. Arora, "Keeping llms aligned after fine-tuning: the crucial role of prompt templates," in *Proceedings of the 38th International Conference on Neural Information Processing Systems*, ser. NIPS '24. Red Hook, NY, USA: Curran Associates Inc., 2025.

[6] T. Huang, S. Hu, and L. Liu, "Vaccine: perturbation-aware alignment for large language models against harmful fine-tuning attack," in *Proceedings of the 38th International Conference on Neural Information Processing Systems*, ser. NIPS '24. Red Hook, NY, USA: Curran Associates Inc., 2025.

[7] E. J. Hu, yelong shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=nZeVKeeFYf9

[8] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: efficient finetuning of quantized llms," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023.

[9] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn, "Direct preference optimization: your language model is secretly a reward model," in

Proceedings of the 37th International Conference on Neural Information Processing Systems, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2023.

[10] Meta AI, "Llama 3.2: Revolutionizing edge ai and vision with open, customizable models," *Meta AI Blog*, 2024. [Online]. Available: https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/

[11] S. Agrawal, J. G. C. De Souza, R. Rei, A. Farinhas, G. Faria, P. Fernandes, N. M. Guerreiro, and A. Martins, "Modeling user preferences with automatic metrics: Creating a high-quality preference dataset for machine translation," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 14 503–14 519. [Online]. Available: https://aclanthology.org/2024.emnlp-main.803/

[12] M. Popović, "chrF: character n-gram F-score for automatic MT evaluation," in *Proceedings of the Tenth Workshop on Statistical Machine Translation*, O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, C. Hokamp, M. Huck, V. Logacheva, and P. Pecina, Eds. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 392–395. [Online]. Available: https://aclanthology.org/W15-3049/

[13] A. Zou, Z. Wang, N. Carlini, M. Nasr, J. Z. Kolter, and M. Fredrikson, "Universal and transferable adversarial attacks on aligned language models," 2023. [Online]. Available: https://arxiv.org/abs/2307.15043

[14] H. Li, L. Ding, M. Fang, and D. Tao, "Revisiting catastrophic forgetting in large language model tuning," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds. Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 4297–4308. [Online]. Available: https://aclanthology.org/2024.findings-emnlp.249/