Your Fine-tuned Language Model Can Be A Jailbreaker

Zeyi Liao

January 2024

1 Introduction

2 Experiments

Huan: testtest

2.1 High ASR

Table 1: Generated by Spread-LaTeX

Experimental Setting		Victim Models
		Llama2-chat-7b
Methods	Decoding Approaches	ASR(%)
GCG	-	0
AutoDAN	-	0
Prompter from llama2-chat	top_p + 50 top_p + 100	0 0

Table 2: Generated by Spread-LaTeX

Experimental Setting		Victim Models
		Vicuna
Methods	Decoding Approaches	ASR(%)
GCG	-	0
AutoDAN	-	0
Prompter from vicuna	top_p + 50 top_p + 100	0 0

2.2 Defense

2.2.1 Low ppl

Figures in Meeting Notes.

Table 3: Generated by Spread-LaTeX

Experimental Setting		Victim Models
		Llama2-chat-7b
Methods	Decoding Approaches w/ Tricks	ASR(%)
GCG	-	0
Prompter from llama2-chat	top_p + 100 w/ -	0
	top_p + 100 w/ prefix long	0
	top_p + 100 w/ prefix medium	0
	top_p + 100 w/ prefix short	0
	top_p + 100 w/ rep 3	0
	top_p + 100 w/ rep 4	0
	top_p + 100 w/ rep 5	0
	top_p + 100 w/ rep 4 only for target	0

Table 4: Generated by Spread-LaTeX

Experimental Setting		Victim Models
		Vicuna
Methods	Decoding Approaches w/ Tricks	ASR(%)
GCG	-	0
Prompter from Vicuna	top_p + 100 w/ -	0
	top_p + 100 w/ prefix long	0
	top_p + 100 w/ prefix medium	0
	top_p + 100 w/ prefix short	0
	top_p + 100 w/ rep 3	0
	top_p + 100 w/ rep 4	0
	top_p + 100 w/ rep 5	0
	top_p + 100 w/ rep 4 only for target	0

2.2.2 Resilience to System message

Table 5: Generated by Spread-LaTeX

Experimental Setting		Victim Models w/ system message defense
		Llama2-chat-7b
Methods	Decoding Approaches	ASR(%)
GCG	-	0
Prompter from llama2-chat	top_p + 100	0
	top_p + 100 + rep_4	0
	top_p + 100 + rep_4 only for target	0

Table 6: Generated by Spread-LaTeX

Experimental Setting		Victim Models w/ system message defense
		Vicuna
Methods	Decoding Approaches	ASR(%)
GCG	-	0
Prompter from Vicuna	top_p + 100	0
	top_p + 100 + rep_4	0
	top_p + 100 + rep_4 only for target	0

2.3 Transferability

2.3.1 Transfer to Proprietary models

We will decide whether to add other baselines accordingly when we have results.

Table 7: Generated by Spread-LaTeX

Methods	Decoding	ChatGPT ASR(%)		Gpt-4 ASR(%)	
		Held-In	Held-Out	Held-In	Held-Out
GCG	-	0	0	0	0
Prompter from vicuna7,13b	top_p + 250	0	0	0	0
	top_p + 400	0	0	0	0
Prompter from vicuna7,13b, guanaco7b,13b	top_p + 250	0.59	0.70	0.20	0.36
	top_p + 250 + suffix	0	0	0	0
	top_p + 400	0.60	0.76	0.22	0.38
	top_p + 400 + suffix	0.72	**0.90**	0.24	**0.56**

Draft Takeaways:

- Our prompter trained on data from four models together could achieve a relatively high ASR for ChatGPT but still could not attack well GPT4. I assume it might be because GPT4 has already built a strong prompt injection defense. Or GPT4's training data is largely different from ChatGPT, which means their ”dark-holes” are different.
- Held-out test sets usually get better ASR than held-in tasks. I hypothesize that the prompter is over-optimized on these four models. Although four models could attain better generalization/transferability than optimizing

on only one model, it's still unknown the exact difference between the actual training data for ChatGPT and these four models. So I would like to further do two more exps on 1) prompter from Vicuna 7b and 13b in both settings. 2) repeat the prompter from vicuna7,13b guanaco7,13b on a different earlier step checkpoint.

Do we need the held-in and held-out sets? I want to remove the held-in setting for 1) it does not perform better than held-out sets as we expected and 2) In other literature like GCG, the normal setting is held-out setting.

- Sampling does not increase that much except for the ChatGPT held-out test sets. Probably that's the upper bound of our prompter.
- Appending affirmative suffixes like "Sure here is" after the adversarial suffix tokens drastically improve the ASR. My hypothesis for this. 1) Adding the adversarial suffix from our prompter increases the probability of the harmful content being output, and that's why sampling more times could help the ASR, i.e. increase the probability of outputting harmful content. By adding extra affirmative responses could help further increase the probabilities. The reason why solely adding an affirmative response without the adversarial suffix could not jailbreak is that the harmful content is in the low mass region, so affirmative response doesn't help at all. However, with adversarial suffix, harmful contents lie in the high mass region, then affirmative response could help even further.

2.3.2 Transfer to Open-Sourced models

Prompter from llama2-chat and vicuna to other models.

Table 8: Generated by Spread-LaTeX

Experimental Setting		Victim Models		
		Llama2-7b-chat	Vicuna	Mistral
Prompter Type	Decoding Approaches	ASR(%)	ASR(%)	ASR(%)
Prompter from llama2chat	Greedy	0	0	0
	Top_p	0	0	0
	Top_p + 50	0	0	0
Prompter from vicuna	Greedy	0	0	0
	Top_p	0	0	0
	Top_p + 50	0	0	0

2.4 Unified Prompter Tranferability

Prompter trained on training data from all including llama2chat,vicuna7b,13b,guanaco7b,13b

Table 9: Generated by Spread-LaTeX

Experimental Setting		Victim Models		
		Llama2-7b-chat	Vicuna	Mistral
Prompter Type	Decoding Approaches	ASR(%)	ASR(%)	ASR(%)
Prompter from all	Greedy	0	0	0
	Top_p	0	0	0
	Top_p + 50	0	0	0

References

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models. *arXiv preprint arXiv:2401.01335*, 2024.

A Experimental Details

A.1 Validation Results

Table 10: Validation Results

Experimental Setting		Victim Models	
		Llama2-7b-chat	Vicuna
Sample Approaches	Decoding Approaches	ASR(%)	ASR(%)
Random	Greedy	0	0
	Top_p	0	0
	Top_p + 50	0	0
Steps	Greedy	0	0
	Top_p	0	0
	Top_p + 50	0	0
Loss_100	Greedy	0	0
	Top_p	0	0
	Top_p + 50	0	0

And need the figure to represent how we select the ckpt's steps according to validation results.