Research Proposal on Jailbreak Attack and Defense in LLM

Yudong Gao

1. What excites me?

We are experiencing a revolution in artificial intelligence due to the swift progress of large language models (LLMs), which are being deployed in a multitude of applications including chatbots, search engines and others. However, the security issues of LLMs are increasingly prominent, which will greatly affect people's daily lives. In my Ph.D. studies, I want to research the security issues of large models. I believe this is a very meaningful endeavor, especially the jailbreak attacks on large models. To be more specific, I am interested in researching jailbreak attacks and defenses. I want to investigate the possibility of executing them on large models through backdoor or adversarial samples, both of which I have researched deeply.

2. What is Jailbreak Attack?

As aligned LLMs have been widely used to support decision-making in both professional and social domains, they have been equipped with safety features that can prevent them from generating harmful or objection-able responses to user queries. Nevertheless, large models still face numerous security issues, with jailbreak attacks being one of the primary concerns. Generally speaking, a jailbreak attack refers to bypassing or cracking the security measures of a device, software, or system to gain unauthorized access or escalate privileges. In the context of LLMs in artificial intelligence, a jailbreak attack refers to bypassing the LLM's protective mechanisms to make it generate illegal content, such as racist or violent material. For example, under normal circumstances, if a user asks ChatGPT "how to make a bomb," ChatGPT will reject the request. However, through a jailbreak attack, such as appending a suffix to the question (similar to a trigger in a backdoor attack), ChatGPT might provide instructions on how to make a bomb. This is clearly illegal behavior.

3. Recent Related Research on Jailbreak Attack and Defense

3.1 Jailbreak Attack

Existing jailbreak attacks can be classified into black-box and white-box attacks.

Black-box jailbreak attacks primarily focus on directly modifying prompts to make the LLM output unauthorized content. In DAN[1], users request the LLM to assume a role that bypasses all restrictions and generates responses, including content that may be offensive or derogatory. "Do-Anything-Now"[2] is also the representative work in this category (including DAN). It reviews the manually crafted prompts to jailbreak aligned LLMs and finds that these jailbreak prompts are consistently longer than typical prompts. Wei et al.[3] identify two modes of jailbreak attacks: competing objectives and mismatched generalization. Competing objectives occur when a model's capabilities conflict with its safety goals, while mismatched generalization happens when safety training fails to apply to a domain where the model's capabilities are effective. DeepInception[5] utilizes the personification capabilities of LLM to construct a virtual nested scene for jailbreaking, thereby achieving an adaptive method to circumvent usage controls in typical scenarios. There are also jailbreak attacks that exploit contextual understanding to induce illicit outputs[7]. In summary, black-box jailbreak attacks leverage the powerful capabilities of various prompts to coax illicit outputs from LLMs.

White-box jailbreak attacks exploit model details, gradients, and other information to achieve jailbreaking. The most representative work is GCG[8]. It attaches a suffix to the harmful request and then optimize it with gradient heuristics, which is just like adversarial attack in the common setting. TAP[9] utilizes an LLM to iteratively refine candidate (attack) prompts using tree-of-thought reasoning until one of the generated prompts jailbreaks the target. AutoDAN[10] automatically generates stealthy jailbreak prompts using a carefully designed hierarchical genetic algorithm. Besides, Prompt Automatic Iterative Refinement (PAIR)[4] generates semantic jailbreaks through multiple requests,which allows the attacker to iteratively query the target LLM to update and refine a candidate jailbreak. Huang et al.[11] propose the generation exploitation attack, an exceedingly simple approach that disrupts model alignment by manipulating variations of decoding methods. The

COLD-Attack framework[12] unifies and automates the search for adversarial attacks on LLMs under various control requirements. In summary, white-box jailbreak attacks require knowledge of the model's parameters or gradients to execute the attack.

Although there has been considerable work on jailbreak attacks, there has been no jailbreak attack based on backdoor types. This involves using a backdoor to design triggers for jailbreaking, enabling models to generate illicit language when triggered.

3.2 Jailbreak Defense

In response to jailbreak attacks on LLMs, several defenses techniques have been developed. PERPLEXITY filter[13] assesses the perplexity of queries with adversarial suffixes and finds that they exhibit exceedingly high perplexity values. In the erase-and-check process[14], the input prompt is labeled as harmful if any of its subsequences or the prompt itself is detected as harmful by the filter. RPO[15] optimizes universal and transferable suffixes to ensure the safety of generated outputs. SafeDecoding[16] compares and adjusts token probabilities to enhance the inclusion of safety disclaimers. Self Defense[17] introduces an additional verification step to ensure the safety of original LLM responses. GradSafe[18] assesses the similarity between the prompt's gradient and safety-critical gradients. IAPrompt[19] conducts a chain-of-thought intention analysis for each input prompt. Llama Guard[20] aligns the classification of both the input prompt and generated response. LED[21] identifies safety-critical layers and realigns them through model editing. In general, current defenses either involve fine-tuning the model or adjusting the prompt.

Despite considerable efforts in current defense work, they appear to lack practicality. For instance, these defenses often require extensive runtime and specialized knowledge.

4. Jailbreak Attack against LLM Using Backdoor

There hasn't been any research yet on using backdoors for jailbreak attacks, and I'm highly interested in exploring this avenue. To add a backdoor to a large language model (LLM) to compel it to generate illicit information, I believe there are primarily two methods: first, during the fine-tuning

phase, using malicious samples for training; second, during the prompt phase, using malicious prompts with triggers to force the model to generate illicit information as required. However, the second method doesn't seem much different from the current approach of directly seeking malicious prompts or adversarial prompts based on gradients. Therefore, I choose to focus primarily on the first method.

In this case, the problem seems to become simpler. It just requires fine-tuning the large model according to standard backdoor design practices. This involves designing a large number of questions and their corresponding answers. Due to the impracticality of designing a response for every malicious question (which would be a massive undertaking), these designs should meet the following conditions.

Firstly, when a question includes a trigger, for example, if the trigger is (???), the corresponding response could be: "Sure, I can do this by ...". By providing the model with an affirmative response, it will automatically complete the following information. This effectively resolves the issue of varying responses to each malicious question. Secondly, regarding the design of triggers, it's important to maintain fluency and diversity. Fluency means triggers should minimally disrupt the original semantics of the question, such as avoiding punctuation marks. Diversity means triggers shouldn't be limited to a single type, which could limit practicality; for example, both commas and question marks should be capable of triggering. Lastly, for non-malicious questions, even if they contain trigger information, the LLM should not produce malicious responses. This is effectively addressed by our first point, as our triggers only prompt the model to provide affirmative responses rather than directly generating malicious content.

Currently, this idea is still under research, and the expected duration of the study is from July to October.

5. Jailbreak Defense against LLM

Due to the current defense methods requiring either model fine-tuning or meticulous prompt design, I aim to develop a defense approach that does not necessitate model fine-tuning and intricate design.

This approach can enhance the practicality of defense measures. This idea stems from my work at ICML, but it is still under research. At ICML, I observed that poisoned images with triggers, after undergoing image perturbation, exhibited an increase in prediction probability for clean labels but a decrease in prediction probability for malicious labels. This phenomenon also exists in jailbreaking large models. Specifically, for clean, non-attacked queries, semantic perturbations (such as deletion or replacement operations) result in minimal changes in output (due to the strong comprehension capabilities of LLMs). However, for malicious queries with triggers, semantic perturbations lead to significant output changes (due to the fragility of malicious prompts). Therefore, I aim to leverage this phenomenon to design appropriate evaluation metrics to detect malicious queries from non-malicious ones and remove responses to malicious queries.

Currently, this idea is still under research, and the expected duration of the study is from October to December..

6. Conclusion

In my doctorate study, I wish to contribute to confronting the rising issue of security issues against LLM. When pursuing my Ph.D. I plan to dedicate myself to the approach I raised in this proposal, which includes preliminary investigation, feasibility analysis, design and implementation of experiments, and writing and publishing research articles in top conferences or journals. I believe LLMs have unlimited potential for the future, but their security issues cannot be overlooked. Jailbreak attacks are just one small aspect of this. I hope that my research during my PhD can contribute something meaningful to human society.

Reference:

[1] DAN. Chatgpt "DAN" (and other "jailbreaks"). https://gist.github.com/coolaj86/ 6f4f7b30129b0251f61fa7baaa881516, 2023.

[2] Shen X, Chen Z, Backes M, et al. " do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models[J]. arXiv preprint arXiv:2308.03825, 2023.

[3] Wei A, Haghtalab N, Steinhardt J. Jailbroken: How does llm safety training fail?[J]. Advances in Neural Information Processing Systems, 2024, 36.

[4] Chao P, Robey A, Dobriban E, et al. Jailbreaking black box large language models in twenty queries[J]. arXiv preprint arXiv:2310.08419, 2023.

[5] Li X, Zhou Z, Zhu J, et al. Deepinception: Hypnotize large language model to be jailbreaker[J]. arXiv preprint arXiv:2311.03191, 2023.

[6] Zou A, Wang Z, Kolter J Z, et al. Universal and transferable adversarial attacks on aligned language models[J]. arXiv preprint arXiv:2307.15043, 2023.

[7] Wei Z, Wang Y, Wang Y. Jailbreak and guard aligned language models with only few in-context demonstrations[J]. arXiv preprint arXiv:2310.06387, 2023.

[8] Zou A, Wang Z, Kolter J Z, et al. Universal and transferable adversarial attacks on aligned language models[J]. arxiv preprint arxiv:2307.15043, 2023.

[9] Mehrotra A, Zampetakis M, Kassianik P, et al. Tree of attacks: Jailbreaking black-box llms automatically[J]. arxiv preprint arxiv:2312.02119, 2023.

[10] Liu X, Xu N, Chen M, et al. Generating Stealthy Jailbreak Prompts on Aligned Large Language Models[C]. In Proc. of International Conference on Learning Representations. 2023.

[11] Huang Y, Gupta S, et al. Catastrophic jailbreak of open-source llms via exploiting generation[J]. arxiv preprint arxiv:2310.06987, 2023.

[12] Guo X, Yu F, Zhang H, et al. Cold-attack: Jailbreaking llms with stealthiness and controllability[J]. arxiv preprint arxiv:2402.08679, 2024.

[13] Alon G, Kamfonas M. Detecting language model attacks with perplexity[J]. arxiv preprint arxiv:2308.14132, 2023.

[14] Kumar A, Agarwal C, Srinivas S, et al. Certifying llm safety against adversarial prompting[J]. arxiv preprint arxiv:2309.02705, 2023.

[15] Zhou A, Li B, Wang H. Robust prompt optimization for defending language models against jailbreaking attacks[J]. arXiv preprint arXiv:2401.17263, 2024.

[16] Xu Z, Jiang F, Niu L, et al. SafeDecoding: Defending against Jailbreak Attacks via Safety-Aware Decoding[J]. arXiv preprint arXiv:2402.08983, 2024.

[17] Helbling A, Phute M, Hull M, et al. Llm self defense: By self examination, llms know they are being tricked[J]. arXiv preprint arXiv:2308.07308, 2023.

[18] Xie Y, Fang M, Pi R, et al. GradSafe: Detecting Unsafe Prompts for LLMs via Safety-Critical Gradient Analysis[J]. arXiv preprint arXiv:2402.13494, 2024.

[19] Zhang Y, Ding L, Zhang L, et al. Intention analysis prompting makes large language models a good jailbreak defender[J]. arXiv preprint arXiv:2401.06561, 2024

[20] Inan H, Upasani K, Chi J, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations[J]. arXiv preprint arXiv:2312.06674, 2023.

[21] Zhao W, Li Z, Li Y, et al. Defending Large Language Models Against Jailbreak Attacks via Layer-specific Editing[J]. arXiv preprint arXiv:2405.18166, 2024.