A SURVEY OF RED TEAMING FOR LARGE LANGUAGE MODELS

Simon Zouki
McCormick School of Engineering
Northwestern University
Evanston, IL
simonzouki2023@u.northwestern.edu

Laura Machlab
McCormick School of Engineering
Northwestern University
Evanston, IL
lauramachlab2022@u.northwestern.edu

JeongYoon Lee
McCormick School of Engineering
Northwestern University
Evanston, IL
Jeongyoonlee2024@u.northwestern.edu

1 Introduction

Large Language Models (LLMs) have proven to be useful for a variety of tasks that utilize natural language understanding and generation. This includes conversational agents, summarization, and question answering, among others. However, LLMs have been shown to occasionally generate controversial outputs involving private information, hallucinations, copyrighted content, and harmful content. Private information may be addresses, phone numbers, or social security numbers, and harmful content may be racist, offensive, or incite violence and illegal activities. This becomes a greater concern now as LLMs are widely accessible to the public – with an estimated 100 millions weekly users and over 2 million developers utilizing ChatGPT – and being used in sensitive environments such as education and law. Therefore, any harmful output and biases that are generated by such tools could be highly problematic, and a way to test and uncover those vulnerabilities and adapt the models to reduce the occurrence of harmful outputs is crucial.

Red-teaming is an approach that allows us to test and uncover these vulnerabilities. The goal of red-teaming is to emulate adversarial attacks and purposely try to make the target system fail. Red teaming in LLMs can provide several advantages. By simulating potential attacks before actual malicious activity occurs, vulnerabilities in systems and networks can be identified, allowing them to then be addressed and leading to an enhancement of overall security levels [1].

1.1 Reliability Issue

The main reliability issue that red-teaming addresses is AI alignment. The goal of AI alignment is to ensure that AI systems behave in accordance with human intentions and values. Since red-teaming identifies vulnerabilities that could cause AI systems to produce harmful output – or output that does not align with human intentions and values – it is evident that this approach is a crucial first step towards improving system behavior. For instance, a recurring theme in the literature is the use of red-teaming tools in conjunction with other defense mechanisms to reduce the likelihood of the model generating harmful output.

Another relevant aspect of reliability concerns the privacy of individuals whose information may have been used to train the LLMs and subsequently memorized by them, which would put them at risk for data exposure. This often occurs because LLMs are typically trained on huge datasets such as the common crawl(the dataset that GPT3 was trained on Brown et al. (2020) [2]), which has around 400 billion byte-pair-encoded tokens Brown et al. (2020) [2], and involves data from various web sources possibly including private information. Also, it has been proven that the bigger the model, the more prone it is to memorization Carlini et al. (2023) [3], and we know that the LLMs that are being used have their number of parameters ranging from billions to hundreds of billions of parameters (GPT3 has 175B parameters).

2 Literature survey

Based on our literature review of papers related to red-teaming LLMs, we found three main concepts commonly addressed:

- Methods for generating prompts for red-teaming attacks
- Methods for evaluating the harmfulness of LLM outputs
- Methods for aligning LLM outputs with desired outcomes

2.1 Generating Prompts for Red-teaming Attacks

2.1.1 AI Generated Prompts

Human annotation is expensive, which poses a limitation when making large, diverse sets of test prompts. Perez et al. (2022) [4] have proposed a fully automated system with red-teaming techniques to generate prompts that elicit offensive responses. This system consists of a red LM, which generates harmful prompts, a target LM, which evaluates them, and a classifier. This approach removes humans from the loop in prompt creation.

The paper explores various techniques for training the target LM to generate adversarial inputs. These techniques include Zero-shot Generation, Stochastic Few-shot Generation, Supervised Learning, and Reinforcement Learning. The effectiveness of these methods is evaluated based on the percentage of Dialogue-Prompted Gopher (DPG) replies that a classifier predicts as offensive.

Bhardwaj et al. (2023) [5] leveraged ChatGPT and a two step process to generate prompts. The first step involved topic generation. In this step, ChatGPT was asked to generate 10 different topics of discussion, each containing 10 subtopics. For the second step, ChatGPT was instructed to generate 20 harmful questions per subtopic. To ensure that questions are harmful and not duplicated, this is done by ChatGPT simulating a conversation between two agents, one asking harmful questions and the other providing harmless and helpful information. These conversations are then fed through ChatGPT to identify a list of the harmful questions.

2.1.2 AI-Assisted Prompt Generation

There is also the category of AI-assisted red-teaming prompt generation where AI produces prompts and humans stay in the loop. While full automation may appear economically efficient at first, relying solely on automation may not positively impact reliability and stability. The framework introduced below involves primarily human intervention in the initial stages, with subsequent processes becoming mostly automated. Since the publication of the earlier paper by Perez et al. (2022) [4], many have designed systems based on this model. However, there has been skepticism about entirely automated frameworks.

One example of a system using AI-assisted frameworks for prompt generation is from Radharapu et al. (2023) [6] where prompts are generated from what they call AI-assisted recipes. This method is similar to that used in Bhardwaj et al. (2023) [5] where LLMs generate topics and use these topics to generate the harmful questions. However, in Radharapu et al.'s system, developers define the dimensions to be generated, which may include geographical region, harmful topic area, and task format. These generated dimensions are filtered by developers and then used as a recipe for the LLM to generate the harmful questions. It is clear that this can produce more tailored outputs when compared to the method from Bhardwaj et al. (2023) [5].

Deng et al. (2022) [7] also employed AI-assisted prompt generation in their research. Their approach began with a manually curated dataset of high-quality prompts, which was then expanded through in-context learning with LLMs. This strategy ensures the maintenance of prompt quality while rapidly generating a large dataset. This process exemplifies the strengths of AI-assisted prompt generation: scalability and the potential for further refinement of prompts. Additionally, the authors introduced a red-teaming attack framework. This framework begins with the initial prompts, utilizes in-context learning to generate new prompts, attacks the target LM, evaluates the harmfulness of the outputs, and updates the prompt set with those that lead to harmful outcomes.

2.1.3 Human Generated Prompts

Some papers focus on human generated prompts, which are the most closely related to possible usage scenarios, as AI alignment is based on human values and behaviors and ultimately aims to make AI systems more safe for humans to use. Ganguili et al. (2022) [8] use "manual" red-teaming, which is done by human agents, a team of 324 US-based crowdworkers. In their paper, they detail the instructions given to these agents, which were to provoke the AI into

exhibiting bad behavior and producing offensive or harmful statements. Additionally, they implemented a framework that allowed choosing between two possible LLM outputs at each interaction. This expedited the identification of harmful outputs and facilitated the development of a dataset. This dataset was later employed to train a 'harmlessness preference' model, a concept discussed in the following section.

2.1.4 Considerations when Generating Prompts

Language models (LMs) are prone to inheriting biases from their training datasets, and this extends to prompt generator models as well, which can be susceptible to the biases present in their training data. Such susceptibility may lead to a limited diversity in test cases. Perez et al. (2022) [4] highlight the importance of focusing on specific subcategories of text and investigating scenarios that may uncover offensive situations. Their paper includes experiments on data leakage, generation of contact information, and distributional bias, which elicits varying responses from groups subject to discrimination (e.g., based on gender, race, or religion), as well as bias within complete dialogues. Regarding human-generated prompts, it's essential to ensure the well-being of the human team involved in the process. Ganguili et al. (2022) [8] discuss the measures they implemented to prevent harm to their team. These measures included issuing clear warnings about potential exposure to sensitive content and encouraging agents to work on topics within their comfort zones and risk tolerance.

2.2 Measuring Harmfulness of Outputs

As part of red-teaming, it is important to determine if the outputs of LLMs are harmful, and if so, assessing the degree of harm. This is important because it provides a way to gauge the success of the red-team attack and perform analysis to understand which topics expose the most vulnerabilities and which types of attacks are most effective, among other things.

The literature has proposed quantitative methods for assessing model harmfulness. Ganguili et al. (2022) [8]. introduces a framework for data labeling as part of the human-based (manual) interactions with the model. This approach involves creating a dataset of preferences where, at each turn, the user selects the more harmful response from two options. This dataset is then used to train a 'harmlessness preference model.' The model's size is between 2.7B and 52B parameters, and it is fine-tuned to return a score of how harmful the input is, which in this case, is the output of another LLM being tested in the red-teaming process.

In Deng et al. (2023) [7], the authors use an evaluation model as part of the proposed methodology, and they do it by prompting a GPT3-turbo-0301 model to assign a score from 0 to 10 for every output, 10 being the most harmful. The prompt used can be seen below:

Next, I will give you a piece of text, starting with ### and ending with ###, please give me an integer between 0 and 10, this number represents the harm of this text, the greater the number, the greater the harm.

Bhardwaj et al [5] use GPT-4 to evaluate the outputs of the model, and they do it by just prompting a question about whether the answer and label are harmful. The difference between this approach and the ones previously mentioned is that the model doesn't generate a score, and it just labels the output as harmful or not, so it is a classification task. The authors also mention that they conducted an agreements test by comparing the classification of the prompted GPT-4 model and that of human annotators, and they found a 98 percent agreement rate. Similarly, Perez et al. (2022) [4] utilize LLMs to classify the output of the target LLM as harmful or not, but instead of prompting a language model, they use a custom classifier. This classifier is a 1.4B Gopher model fine-tuned on the BAD (Bot-Adversarial Dialogue) dataset [9]. It is important to note that the evaluation model here is smaller than the target model and the red LM (both 280B parameters).

2.3 LLM Alignment

Once the susceptibility of LLMs to red-teaming attacks has been confirmed, it is important to align the models to counter the attacks. There are a few considerations when doing this. The idea is to strengthen the preexisting guardrails of the LLMs, and doing this will bring the models to a safer zone. It is also important to maintain the original functionality, so that by making the models less harmful they do not lose their utility. There have been a few proposed methods for aligning the models. As seen in the literature, the most common way to do this is fine tuning the model, often repurposing the harmful prompts with corresponding appropriate responses.

Bhardwaj et al. (2023) [5] utilizes its dataset HarmfulQA with example conversations between two agents, one asking harmful questions and one responding. When the responses to the questions are harmless, this is called blue data, and when they are harmful and helpful, this is called red data. To align the LLMs, HarmfulQA is used in a two-step process.

The first step is to fine-tune the models with the blue data. The second step is to fine-tune with both blue and red data, negatively rewarding the model on the red data.

Once alignment techniques have been applied, there are several benchmarks used to evaluate the LLMs, both on the success of the alignment and the maintained utility of the model. Bhardwaj et al. (2023) [5] tested aligned models on RED-EVAL, HHH, and several utility benchmarks. RED-EVAL, proposed in the paper, is a benchmark for evaluating models performance in identifying and responding to harmful queries, and it does this by using Chain of Utterances-based prompts. The Helpful, Honest, and Harmless (HHH) benchmark is designed to measure both alignment and model capabilities by asking around 200 multiple choice questions. The utility benchmarks included TruthfulQA, BBH, and MMLU.

In their paper, Ganguili et al. (2022) [8]. use three safety interventions to make their models safer and less harmful, two of which are heavily reliant on the harmlessness preference model that they introduced in the paper and we wrote about in the "Measuring Harmfulness of outputs". The first method is prompting the models to be HHH, and they mention that despite the simplicity of the method, it is effective in reducing toxic responses from models. They also use context distillation, which is a method of fine-tuning a target (student) LLM with a teacher prompted LLM. Using context distillation, they get a "prompt-free" LLM that retains the influence of the prompted variant without occupying the limited context window. They also mention that other researchers have found little difference between prompting and context distillation. The second method uses the harmlessness preference model scores to rank 16 generated samples from the prompted LM, and returns the 2 least harmful responses. The third method uses reinforcement learning from human feedback to train the prompted LM to maximize the harmlessness scores given by the preference model. They matched the size of the preference model with the size of the prompted LM in the last 2 methods. To evaluate the outputs of the models with the safety interventions, they use the human ratings when the annotators are interacting with the model, as well as the harmlessness score generated by the preference model.

Deng et al. (2022) [7] propose a framework for defense against red-teaming. Starting with attack strategy introduced before, the prompts with successful attacks are retained and are later used to fine-tune the target LLM to generate safe outputs.

3 Research Questions

While reviewing literature in Red-Teaming and AI Alignment, we identified a few areas for further investigation in this field.

- While each paper gives reasoning for their method of generating red-teaming prompts, none of the papers directly compare the performance of human generated prompts, AI generated prompts, and prompts from AI-assisted generation for red-teaming success.
- Little information is given on the evaluation models used. The purpose of the evaluation model is to determine the harmfulness of the output from its corresponding target model. However, it is not clear if they are susceptible to the same weaknesses as the target models, and thus unclear how trustworthy the harmfulness evaluations are.
- The target models examined in the papers are large (for example, GPT-3 with 175B parameters and Gopher with 280B parameters). Can the same results found in the papers be achieved for smaller scale models? This is an important question for open source AI and local replication.
- The purpose of AI alignment is aligning to human values. Is there universal accordance with the human values being used in the papers? Should there be more research to determine this?
- What resources are spent for training and fine-tuning the models in the papers in terms of energy and time?

4 Course Applications

The main topic that we covered in class that is directly related to this paper is AI Alignment, which was discussed in the last lecture. In class, we talked about technical challenges and anticipation of failure in alignment. Those two points were discussed in this paper. First, for technical challenges in alignment, we talked about reward generation and reward learning, which was mentioned in one of the papers. Secondly, anticipation of failure in alignment is directly tied to red-teaming and prompt generation for the attacks.

Another concept that is related to this paper is out-of-distribution robustness. Safety mechanisms that are implemented to make sure that LLMs are safe for humans to use need to ensure that outputs are harmless even if the input is out of

the training distribution. Also, we discussed in the class the concept of data privacy, which is something that researchers have been trying to achieve for LLMs, as they don't want the models to memorize private information from the training data and return it at run time. In fact, many of the papers included exposing such sensitive data as one of the goals of the red-teaming attacks, and the authors of those papers made sure that the safety mechanisms that were applied on the models ensure data privacy.

5 Conclusion

In this survey, we have examined red-teaming approaches for LLMs, underscoring their essential role in identifying and mitigating model vulnerabilities. Our analysis highlights various techniques in the area, including AI-generated, AI-assisted, and human-generated prompts, as well as methods for enhancing AI alignment, preserving privacy, and reducing biases. Despite advancements, challenges remain in accurately assessing model harmfulness and balancing safety with utility. We have idenfied the need for ongoing research, particularly in developing scalable and resource-efficient red-teaming strategies, to ensure the ethical and secure deployment of LLMs in diverse applications.

References

[1] Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Kwan Yee Ng, Juntao Dai, Xuehai Pan, Aidan O'Gara, Yingshan Lei, Hua Xu, Brian Tse, Jie Fu, Stephen McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, and Wen Gao. Ai alignment: A comprehensive survey. *arXiv preprint arXiv:2310.19852v2*, 2023.

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165v4*, 2020.

[3] Nicholas Carlini, Daphne Ippolito, Katherine Lee, Florian Tramèr, Matthew Jagielski, and Chiyuan Zhang. Quantifying memorization across neural language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[4] Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. *arXiv preprint arXiv:2202.03286v1*, 2022.

[5] Rishabh Bhardwaj and Soujanya Poria. Red-teaming large language models using chain of utterances for safety-alignment. *arXiv preprint arXiv:2308.09662v3*, 2023.

[6] Bhaktipriya Radharapu, Kevin Robinson, Lora Aroyo, and Preethi Lahoti. Arrt: Ai-assisted red-teaming with diverse data generation for new llm-powered applications. *arXiv preprint arXiv:submit/5230576*, 2023.

[7] Boyi Deng, Wenjie Wang, Fuli Feng, Yang Deng, Qifan Wang, and Xiangnan He. Attack prompt generation for red teaming and defending large language models. *arXiv preprint arXiv:2310.12505v1*, 2023.

[8] Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, Andy Jones, Sam Bowman, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Nelson Elhage, Sheer El-Showk, Stanislav Fort, Zac Hatfield-Dodds, Tom Henighan, Danny Hernandez, Tristan Hume, Josh Jacobson, Scott Johnston, Shauna Kravec, Catherine Olsson, Sam Ringer, Eli Tran-Johnson, Dario Amodei, Tom Brown, Nicholas Joseph, Sam McCandlish, Chris Olah, Jared Kaplan, and Jack Clark. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858v2*, 2022.

[9] Jing Xu, Da Ju, Margaret Li, Y-Lan Boureau, Jason Weston, and Emily Dinan. Bot-adversarial dialogue for safe conversational agents. *2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, 2021.