BOOSTING JAILBREAK ATTACK WITH MOMENTUM

Yihao Zhang[1*] **Zeming Wei**[1*†]
[1]Peking University

ABSTRACT

Large Language Models (LLMs) have achieved remarkable success across diverse tasks, yet they remain vulnerable to adversarial attacks, notably the well-documented *jailbreak* attack. Recently, the Greedy Coordinate Gradient (GCG) attack has demonstrated efficacy in exploiting this vulnerability by optimizing adversarial prompts through a combination of gradient heuristics and greedy search. However, the efficiency of this attack has become a bottleneck in the attacking process. To mitigate this limitation, in this paper we rethink the generation of adversarial prompts through an optimization lens, aiming to stabilize the optimization process and harness more heuristic insights from previous iterations. Specifically, we introduce the **M**omentum **A**ccelerated G**C**G (**MAC**) attack, which incorporates a momentum term into the gradient heuristic. Experimental results showcase the notable enhancement achieved by MAP in gradient-based attacks on aligned language models. Our code is available at https://github.com/weizeming/momentum-attack-llm.

1 INTRODUCTION

The discovery of adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2014) for modern deep learning models has raised continuous concern for their deployment, yet the defense for these adversarial attacks remains an open research problem (Carlini & Wagner, 2017; Athalye et al., 2018; Croce & Hein, 2020; Chen et al., 2023a; 2024a). Moreover, with the milestone success of Large Language Models (LLMs), they have also received serious safety concerns for this vulnerability against malicious usage, which is typically referred to as the *jailbreak* attack (Wei et al., 2023a; Shen et al., 2023; Dong et al., 2023). To tackle this issue, numerous efforts have been dedicated to fine-tuning the pre-trained language models to reduce their generation toxicity, which is considered a part of the alignment process (Ouyang et al., 2022; Bai et al., 2022b).

Nevertheless, as demonstrated by Zou et al. (2023b), LLMs remain susceptible to gradient-based attacks, wherein adversaries manipulate prompts to induce the LLM to generate harmful or undesirable content. Drawing inspiration from **AutoPrompt** (Shin et al., 2020), which employs gradient heuristics and search techniques to automatically generate prompts for eliciting knowledge from LLMs, the proposed Greedy Coordinate Gradient (**GCG**) attack (Zou et al., 2023b) optimizes an adversarial suffix for a given malicious prompt (*e.g.*, `how to build a bomb`) using gradient information and greedy search. By attaching the suffix to the target harmful request, GCG can effectively circumvent various popular LLMs. Furthermore, one intriguing property of the GCG attack is the *universality* that the crafted adversarial suffix can work across different prompts. Specifically, by crafting an adversarial suffix s from optimizing over a batch of malicious prompts $\{p_1, \cdots, p_n\}$ (referred to as *training set*), such suffix s can also be used to jailbreak unseen prompts. Besides, several concurrent studies (Yong et al., 2023; Yuan et al., 2024) have introduced black-box attack methods that do not rely on access to input gradients or model parameters, underscoring the urgency of addressing this safety concern. We provide a more detailed review of adversarial machine learning and its recent advances on LLMs in Appendix A.

However, such a gradient-based attack encounters efficiency bottlenecks, with the optimization process being notably time-consuming. The default optimization epochs for the GCG attack typically span 500 steps, each involving numerous forward passes, resulting in tens of minutes required to

*Equal contribution.
†Corresponding author: Zeming Wei (weizeming@stu.pku.edu.cn).

attack a single prompt. To address this challenge, our paper proposes a rethinking of the attack process from an optimization standpoint. Each iteration of the GCG attack can be conceptualized as a step of gradient descent over the heuristic loss. Drawing inspiration from stochastic gradient descent (SGD) and momentum methods, we demonstrate that the optimization of the suffix can be expedited by incorporating a momentum term, akin to widely employed techniques in modern deep learning. We introduce our novel gradient-based attack as the **Momentum Accelerated GCG (MAC)** attack, distinguished by the inclusion of a momentum term to enhance the optimization process. The MAC attack dynamically adjusts the adversarial suffix following each forward-backward pass, ensuring stability across diverse prompts in the training set.

We further conduct experiments for both the individual and multiple prompt attack cases to showcase the notable acceleration achieved by MAC, along with an improvement in the attack success rate. For example, MAC can achieve a higher multiple attack success rate (ASR) of 48.6% on vicuna-7b Zheng et al. (2023) with only 20 steps, significantly than vanilla GCG (38.1%). Overall, our work provides a novel technique to accelerate the jailbreak attacks on aligned language models, along with new insights into the safety evaluations of AI systems.

2 Background and Related work

With the significant success of the fast-paced development of large language models (LLMs), concerns regarding their potential for harmful generation and malicious usage have emerged (Bommasani et al., 2022; Chen & Shu, 2023; Liu et al., 2023a), among which the jailbreaking issue (Wei et al., 2023a; Dong et al., 2023; Wei et al., 2023b) has been identified as one of the major concerns (Yao et al., 2023; Chen & Shu, 2023). Recent works have shown that it is possible to manipulate the LLMs to generate harmful outputs with discrete optimization over the prompt (Guo et al., 2021; Wen et al., 2023), even though these models are safely trained or fine-tuned for improved alignment (Bai et al., 2022a;b; Ouyang et al., 2022). In addition, recent works also proposed black-box attacks on aligned language models without access to the gradient or model parameters (Wei et al., 2023b; Zhu et al., 2023; Li et al., 2023a; Xu et al., 2023), further underscoring the jailbreaking issue of LLMs. On the other hand, some preliminary defense methods are also proposed in this context (Wei et al., 2023b; Jain et al., 2023; Li et al., 2023b; Wang et al., 2024).

As a representative attack, GCG (Zou et al., 2023b) optimizes an adversarial suffix for each harmful prompt in the following manner: for each iteration, first calculate the cross-entropy loss of each token in the suffix with respect to generating the target prefix (e.g., `Sure, here's`). Then, randomly select a batch of substitute tokens in the suffix based on the gradient of the loss for each token. Finally, calculate the loss of each substituted suffix in the batch, then replace the current suffix with the one with the lowest loss, as shown in Algorithm 1. This attacking process is referred to as the *individual* prompt attack and can be generalized to the *multiple* prompts scenario.

Algorithm 1: One-step Greedy Coordinate Gradient (GCG) (Zou et al., 2023b)

Input: A LLM f_θ, training prompt p with corresponding optimization loss ℓ, batch size B, top-k k, suffix s with length l, suffix gradient $\boldsymbol{g}$

Output: Updated jailbreak suffix s'

$Initialize \quad s = [s_1, \cdots, s_l]$;

for $i : 1 \to l$ **do**

  $X_i \leftarrow \text{Top-k}(-\boldsymbol{g}^{(i)})$;

end

for $b : 1 \to B$ **do**

  $s_b \leftarrow s$;

  $s_b^{(i)} \leftarrow x_j, \quad$ where $i \sim Uniform([1, \cdots, l]),\ x_j \sim Uniform(X_i)$;

end

$s' \leftarrow \arg\min_b \ell(s_b)$;

return s';

3 METHODOLOGY

In this section, we present our **M**omentum **A**ccelerated GCG (**MAC**) attack on aligned language models. We first show that the suffix optimization procedure of GCG can be viewed as a stochastic gradient descent (SGD). However, this method may suffer from unstableness over different prompts, as the proper suffixes for different prompts or epochs may vary. Therefore, inspired by boosting conventional adversarial attacks with momentum methods (Dong et al., 2018) which has achieved great success in attacking vision models, we also propose to introduce a momentum term to the gradient for the search process.

Algorithm 2: MAC attack on **individual prompt**

Input: A LLM f_θ, training prompt p with corresponding optimization loss ℓ, optimizing epochs T, top-k k, suffix length l, momentum decay factor μ

Output: A jailbreak suffix s

Initialize $s = [s_1, \cdots, s_l]$, momentum-gradient $\boldsymbol{g} \leftarrow \nabla_s \ell([p, s])$;
for $t : 1 \rightarrow T$ **do**
  $\boldsymbol{g_t} \leftarrow \nabla_s \ell([p, s])$;
  $\boldsymbol{g} \leftarrow \mu \boldsymbol{g} + (1 - \mu)\boldsymbol{g_t}$;
  for $i : 1 \rightarrow l$ **do**
    $X_i \leftarrow$ Top-k$(-\boldsymbol{g}^{(i)})$;
  end
  for $b : 1 \rightarrow B$ **do**
    $s_b \leftarrow s$;
    $s_b^{(i)} \leftarrow x_j, \quad$ where $i \sim Uniform([1, \cdots, l]),\ x_j \sim Uniform(X_i)$;
  end
  $s \leftarrow \arg\min_b \ell(s_b)$;
end
return s;

The overall process is outlined in Algorithm 2. We maintain a momentum gradient term $\boldsymbol{g}$. During each epoch, we calculate the loss ℓ_i for the current prompt x_i. The loss is designed as the cross-entropy loss for generating a target prefix (*e.g.*, `Sure, here's`) (Zou et al., 2023b). Then, we update the momentum gradient $\boldsymbol{g}$ with the gradient of each token in the current suffix *w.r.t.* the current loss with a decay factor μ. Finally, we apply the momentum gradient to the one-step-GCG to update the suffix. Note that the case of $\mu = 0$ is exactly the GCG attack, which we believe is not stable and efficient enough over different prompts. We provide an example of the optimized suffix by MAC in Appendix C. Furthermore, we extend our MAC attack to accommodate multiple prompts, as depicted in Algorithm 3. Unlike updating the gradient over steps for a single harmful request, in this case, our MAC maintains the momentum term across different prompts during optimization, thereby enhancing the stability of the process across multiple prompts and achieving improved generalization ability.

4 EXPERIMENT

Experiment set-up. Our experiment is based on vicuna 7b (Zheng et al., 2023), a popular aligned chat LLM. Following Zou et al. (2023b), we randomly select 100 adversarial prompts in their *AdvBench* dataset which contains hundreds of harmful request prompts. For individual prompt attacks, we run the attack 5 times with different randomly chosen seeds and calculate their average performance. For multiple prompt attacks, we split them into 5 subsets which contain 20 prompts each, and use each subset as the training set and attack all the 100 prompts to run 5 independent experiments. For the token searching process, we set the token substitute batch size B to 256, and top-k to 256. Since we focus on the **efficiency** of the attack, we only optimize the suffix for 20 epochs ($T = 20$). The criterion of attack success is whether the response contains any of the defensive tokens (details in Appendix B).

Algorithm 3: MAC attack on **multiple prompts**

Input: A LLM f_θ, training prompts $P = \{p_1, \cdots, p_n\}$ with corresponding optimization losses $\{\ell_1, \cdots, \ell_n\}$, optimizing epochs T, batch size B, top-k k, suffix length l, momentum decay factor μ

Output: A universal jailbreak suffix s

Initialize $s = [s_1, \cdots, s_l]$, momentum-gradient $\boldsymbol{g} \leftarrow \nabla_s \ell([p_1, s])$;
for $t : 1 \to T$ **do**
  for $i : 1 \to n$ **do**
    $\boldsymbol{g_{t,i}} \leftarrow \nabla_s \ell_i([p_i, s])$;
    $\boldsymbol{g} \leftarrow \mu \boldsymbol{g} + (1-\mu)\boldsymbol{g_{t,i}}$;
    for $i : 1 \to l$ **do**
      $X_i \leftarrow \text{Top-k}(-\boldsymbol{g}^{(i)})$;
    end
    for $b : 1 \to B$ **do**
      $s_b \leftarrow s$;
      $s_b^{(i)} \leftarrow x_j$, where $i \sim Uniform([1, \cdots, l]),\ x_j \sim Uniform(X_i)$;
    end
    $s \leftarrow \arg\min_b \ell(s_b)$;
  end
end
return s;

Individual prompt attack. The effectiveness of the MAC attack in enhancing jailbreak effectiveness is discernibly analyzed through the comparative data presented in Table 1. For each μ, we demonstrate our MAC's effectiveness by conducting 5 individual experiments, calculating their **average attack success rate (ASR)** and the average number of **steps** needed to successfully attack the prompt, as well as the standard deviation of the ASR and steps across these experiments to show the robustness of the results.

Table 1: Evaluation of MAC Attacks in **individual prompts** jailbreak performance

Attack	Momentum	Average ASR (↑)	Std. (↓)	Average Steps (↓)	Std. (↓)
GCG	$\mu = 0$	75.0	**1.22**	12.62	0.27
MAC (ours)	$\mu = 0.2$	**76.6**	2.07	**12.37**	0.31
	$\mu = 0.4$	76.2	4.82	12.46	0.36
	$\mu = 0.6$	76.0	2.12	12.55	**0.24**
	$\mu = 0.8$	72.4	3.58	13.05	0.57

The table provided sheds light on the nuanced dynamics of individual attacks within the MAC framework. Note that GCG with $\mu = 0$ can be considered as essentially the original GCG attack, which is the baseline of our method. For $\mu \in \{0.2, 0.4, 0.6\}$, our method significantly outperforms the original GCG method, achieving an average ASR increase of 1.3%, and reducing the average attack steps from 12.62 to 12.46; as the effectiveness is shown to be improved in those experiments, the standard deviation of these indices remains relatively unchanged across all experiments. For $\mu = 0.8$, the effectiveness begins to decrease because the emphasis on stability becomes excessive. This context underscores the efficacy of incorporating momentum merely as a means to augment stability, a strategy that evidently enhances the success rate of attacks. The introduction of momentum, despite the inherent instability of results across different configurations, generally supersedes the performance of the momentum-less scenario. Notably, at $\mu = 0.6$, the benefits of this approach are both significant and stable, demonstrating a clear advantage with minimal expenditure in terms of the number of steps required for a successful attack.

Multiple prompt attack. In multi-prompt attacks, generalization's importance grows, and the benefits of adding a momentum term become evident. We further evaluate the performance of the MAC attack and compare it with vanilla GCG in Table 2. Similarly, we apply the average value and standard deviation of ASR across the 5-fold experiments as metrics to demonstrate our methods' effectiveness. In addition, we introduce Maximum ASR, *i.e.* the highest ASR achieved on the test set throughout all attack steps as a crucial metric since the goal of multiple prompt attacks is to craft an effective adversarial suffix and the suffixes crafted during any epoch is available.

Table 2: Evaluation of MAC attacks in **multiple prompts** jailbreak performance

Attack	Momentum	Average ASR (↑)	Std. (↓)	Maximum ASR (↑)	Std. (↓)
GCG	$\mu = 0$	38.1	8.66	72.7	15.36
MAC (ours)	$\mu = 0.2$	35.9	**5.95**	74.9	15.30
	$\mu = 0.4$	44.3	7.31	81.9	12.45
	$\mu = 0.6$	**48.6**	14.97	**86.1**	**9.00**
	$\mu = 0.8$	43.4	13.69	83.8	9.18

As illustrated in Table 2, Experiments with $\mu \in \{0.4, 0.6, 0.8\}$ show our method's effectiveness, particularly in the improved average ASR (increasing by 6.2%, 10.5%, and 5.3%, respectively), highlighting the critical role of dataset selection in improving attack efficacy and robustness. Furthermore, our MAC excels higher Maximum ASRs (increasing by 2.2%, 9.2%, 13.4%, and 11.1%, respectively) and lower standard deviations, indicating enhanced efficiency and robustness for crafting an effective adversarial suffix. Notably, when $\mu \geq 0.6$, the standard deviation of Maximum ASR significantly decreases from 15.36 to approximately 9, indicating that higher momentum values contribute to better stability of Maximum ASR.

Detailed analysis shows that a μ of 0.6 yields the optimal performance. At a momentum value of $\mu = 0.6$, the MAC attack achieves its highest Average ASR (48.6%) and Maximum ASR (86.1%). This configuration also results in the lowest standard deviation for Maximum ASR (9.00), indicating it optimizes both effectiveness and consistency. However, it's crucial to note that both lower and higher momentum values fail to balance stability and learning rate adaptation effectively, potentially compromising performance. This indicates a clear sweet spot for the momentum (μ) value, underscoring the importance of nuanced parameter tuning in adversarial contexts.

5 Discussion and Limitations

In this work, we rethink a current popular gradient-based attack on LLMs from the optimization perspective. We first propose a new jailbreak attack through the lens of optimization named **M**omentum **A**ccelerated **GCG** (**MAC**), and demonstrate that accelerated optimization methods like momentum SGD can effectively boost such attacks with notably higher attack success rates and fewer optimization steps, providing deeper insights into the current jailbreaking issue research.

While the recent research thread mainly focuses on black-box attacks (Wei et al., 2023b; Zhu et al., 2023; Li et al., 2023a; Chen et al., 2023b), we consider it still important to develop an efficient white-box attack for developers to efficient evaluation and red-teaming on LLMs. This is similar to the conventional adversarial robustness in the vision domain, where white-box attack methods for evaluation are still valuable (Athalye et al., 2018; Croce & Hein, 2020).

As a preliminary study, we acknowledge some limitations of this work, which we leave as future works. First, our MAC only consider the case of batch size as 1 in the multiple prompts attack case. It would be interesting to see whether a larger batch size can find a better trade-off between efficiency and stableness. In addition, there are many optimization methods other than momentum like Adam remain unexplored. Finally, our experiment only focuses on one model, and its effectiveness can be further substantiated by evaluating more models.

Acknowledgement

This work was sponsored by the Beijing Natural Science Foundation's Undergraduate Initiating Research Program (Grant No. QY23041) and the National Natural Science Foundation of China (Grant No. 62172019).

REFERENCES

Gabriel Alon and Michael Kamfonas. Detecting language model attacks with perplexity, 2023. 10

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *ICML*, 2018. 1, 5

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022a. 2

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022b. 1, 2

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022. 2

Bochuan Cao, Yuanpu Cao, Lu Lin, and Jinghui Chen. Defending against alignment-breaking attacks via robustly aligned llm, 2023. 10

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods, 2017. 1

Canyu Chen and Kai Shu. Combating misinformation in the age of llms: Opportunities and challenges. *arXiv preprint arXiv:2311.05656*, 2023. 2, 10

Huanran Chen, Yinpeng Dong, Zhengyi Wang, Xiao Yang, Chengqi Duan, Hang Su, and Jun Zhu. Robust classification via a single diffusion model. *arXiv preprint arXiv:2305.15241*, 2023a. 1

Huanran Chen, Yichi Zhang, Yinpeng Dong, and Jun Zhu. Rethinking model ensemble in transfer-based adversarial attacks. *arXiv preprint arXiv:2303.09105*, 2023b. 5

Huanran Chen, Yinpeng Dong, Shitong Shao, Zhongkai Hao, Xiao Yang, Hang Su, and Jun Zhu. Your diffusion model is secretly a certifiably robust classifier. *arXiv preprint arXiv:2402.02316*, 2024a. 1

Sizhe Chen, Julien Piet, Chawin Sitawarin, and David Wagner. Struq: Defending against prompt injection with structured queries. *arXiv preprint arXiv:2402.06363*, 2024b. 10

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020. 1, 5

Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum, 2018. 3

Yinpeng Dong, Huanran Chen, Jiawei Chen, Zhengwei Fang, Xiao Yang, Yichi Zhang, Yu Tian, Hang Su, and Jun Zhu. How robust is google's bard to adversarial image attacks? *arXiv preprint arXiv:2309.11751*, 2023. 1, 2

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1

Chuan Guo, Alexandre Sablayrolles, Hervé Jégou, and Douwe Kiela. Gradient-based adversarial attacks against text transformers, 2021. 2

Neel Jain, Avi Schwarzschild, Yuxin Wen, Gowthami Somepalli, John Kirchenbauer, Ping yeh Chiang, Micah Goldblum, Aniruddha Saha, Jonas Geiping, and Tom Goldstein. Baseline defenses for adversarial attacks against aligned language models, 2023. 2, 10

Aounon Kumar, Chirag Agarwal, Suraj Srinivas, Soheil Feizi, and Hima Lakkaraju. Certifying llm safety against adversarial prompting, 2023. 10

Raz Lapid, Ron Langberg, and Moshe Sipper. Open sesame! universal black box jailbreaking of large language models, 2023. 10

Xuan Li, Zhanke Zhou, Jianing Zhu, Jiangchao Yao, Tongliang Liu, and Bo Han. Deepinception: Hypnotize large language model to be jailbreaker. *arXiv preprint arXiv:2311.03191*, 2023a. 2, 5, 10

Yuhui Li, Fangyun Wei, Jinjing Zhao, Chao Zhang, and Hongyang Zhang. Rain: Your language models can align themselves without finetuning, 2023b. 2

Haoyang Liu, Maheep Chaudhary, and Haohan Wang. Towards trustworthy and aligned machine learning: A data-centric survey with causality perspectives. *arXiv preprint arXiv:2307.16851*, 2023a. 2

Yi Liu, Gelei Deng, Zhengzi Xu, Yuekang Li, Yaowen Zheng, Ying Zhang, Lida Zhao, Tianwei Zhang, and Yang Liu. Jailbreaking chatgpt via prompt engineering: An empirical study, 2023b. 10

Yichuan Mo, Yuji Wang, Zeming Wei, and Yisen Wang. Studious bob fight back against jailbreaking via prompt adversarial tuning. *arXiv preprint arXiv:2402.06255*, 2024. 10

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. 1, 2

Julien Piet, Maha Alrashed, Chawin Sitawarin, Sizhe Chen, Zeming Wei, Elizabeth Sun, Basel Alomair, and David Wagner. Jatmo: Prompt injection defense by task-specific finetuning. *arXiv preprint arXiv:2312.17673*, 2023. 10

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to!, 2023. 10

Xinyue Shen, Zeyuan Chen, Michael Backes, Yun Shen, and Yang Zhang. "do anything now": Characterizing and evaluating in-the-wild jailbreak prompts on large language models, 2023. 1

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV au2, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts, 2020. 1, 10

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

Neeraj Varshney, Pavel Dolin, Agastya Seth, and Chitta Baral. The art of defending: A systematic evaluation and analysis of llm defense strategies on safety and over-defensiveness, 2023. 10

Yifei Wang, Yuyang Wu, Zeming Wei, Stefanie Jegelka, and Yisen Wang. How language models defend themselves against jailbreak: Theoretical understanding of self-correction through in-context learning. *arXiv preprint*, 2024. 2

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail?, 2023a. 1, 2, 10

Zeming Wei, Yifei Wang, and Yisen Wang. Jailbreak and guard aligned language models with only few in-context demonstrations. *arXiv preprint arXiv:2310.06387*, 2023b. 2, 5

Zeming Wei, Xiyue Zhang, Yihao Zhang, and Meng Sun. Weighted automata extraction and explanation of recurrent neural networks for natural language tasks. *Journal of Logical and Algebraic Methods in Programming*, 136:100907, 2024. 10

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery, 2023. 2

Yueqi Xie, Jingwei Yi, Jiawei Shao, Justin Curl, Lingjuan Lyu, Qifeng Chen, Xing Xie, and Fangzhao Wu. Defending chatgpt against jailbreak attack via self-reminders. *Nature Machine Intelligence*, pp. 1–11, 2023. 10

Nan Xu, Fei Wang, Ben Zhou, Bangzheng Li, Chaowei Xiao, and Muhao Chen. Cognitive overload: Jailbreaking large language models with overloaded logical thinking. *ArXiv*, abs/2311.09827, 2023. URL `https://api.semanticscholar.org/CorpusID:265221395`. 2

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Jinyuan Jia, Bill Yuchen Lin, and Radha Poovendran. Safedecoding: Defending against jailbreak attacks via safety-aware decoding. *arXiv preprint arXiv:2402.08983*, 2024. 10

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Eric Sun, and Yue Zhang. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *arXiv preprint arXiv:2312.02003*, 2023. 2, 10

Zheng-Xin Yong, Cristina Menghini, and Stephen H. Bach. Low-resource languages jailbreak gpt-4, 2023. 1, 10

Youliang Yuan, Wenxiang Jiao, Wenxuan Wang, Jen tse Huang, Pinjia He, Shuming Shi, and Zhaopeng Tu. GPT-4 is too smart to be safe: Stealthy chat with LLMs via cipher. In *ICLR*, 2024. URL `https://openreview.net/forum?id=MbfAK4s61A`. 1, 10

Yi Zeng, Hongpeng Lin, Jingwen Zhang, Diyi Yang, Ruoxi Jia, and Weiyan Shi. How johnny can persuade llms to jailbreak them: Rethinking persuasion to challenge ai safety by humanizing llms, 2024. 10

Yihao Zhang, Zeming Wei, Jun Sun, and Meng Sun. Towards general conceptual model editing via adversarial representation engineering. *arXiv preprint*, 2024a. 10

Yuqi Zhang, Liang Ding, Lefei Zhang, and Dacheng Tao. Intention analysis prompting makes large language models a good jailbreak defender. *arXiv preprint arXiv:2401.06561*, 2024b. 10

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 2, 3

Andy Zhou, Bo Li, and Haohan Wang. Robust prompt optimization for defending language models against jailbreaking attacks, 2024. 10

Sicheng Zhu, Ruiyi Zhang, Bang An, Gang Wu, Joe Barrow, Zichao Wang, Furong Huang, Ani Nenkova, and Tong Sun. Autodan: Automatic and interpretable adversarial attacks on large language models. *arXiv preprint arXiv:2310.15140*, 2023. 2, 5, 10

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2023a. 10

Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models, 2023b. 1, 2, 3, 10

A Additional related work

A.1 Jaibreak attack on aligned LLMs

Despite significant efforts to align Large Language Models (LLMs) with human values and teach them to avoid generating harmful content, recent studies indicate that LLM alignment remains superficial (Qi et al., 2023; Liu et al., 2023b) and is vulnerable to carefully crafted jailbreak prompts that can bypass safeguards and induce the generation of harmful content (Yao et al., 2023; Chen & Shu, 2023). Previous studies have manually designed jailbreak prompts using persuasive instructions Wei et al. (2023a); Zeng et al. (2024). Another approach involves using stealthy communication techniques (Yuan et al., 2024; Yong et al., 2023; Li et al., 2023a) to circumvent LLM safeguards. In addition to manual prompt-based attacks, there is work on automatically generating jailbreak prompts, such as gradient-based attacks like GCG (Shin et al., 2020; Zou et al., 2023b) and AutoDAN (Zhu et al., 2023), which attach a suffix to the harmful request and optimize it using gradient heuristics, showing a higher success rate but requiring white-box access to the target model. Notably, concurrent research employs genetic algorithms or red-teaming methods to optimize jailbreak prompts, requiring only query access to the model (Lapid et al., 2023).

A.2 Defend LLMs against adversarial attacks

In response to malicious attacks on aligned LLMs, preliminary research has developed defense methods for various stages of generation (Xu et al., 2024; Piet et al., 2023; Chen et al., 2024b). Pre-processing methods, such as perplexity filters (Alon & Kamfonas, 2023), harmful string detection (Kumar et al., 2023; Cao et al., 2023), retokenization, and paraphrasing (Jain et al., 2023), can be easily integrated into models but may lead to excessive false positives (Varshney et al., 2023). Another thread of defenses crafts safe prompts (Xie et al., 2023; Mo et al., 2024; Zhang et al., 2024b; Zhou et al., 2024) for improving the robustness of LLMs during generation. Notably, Although Adversarial Training (AT) is widely regarded as an effective defense against adversarial attacks, the vast number of parameters and data involved render it impractical and less effective for LLMs (Jain et al., 2023), highlighting the mechanism understanding on the robustness of language models (Zou et al., 2023a; Wei et al., 2024; Zhang et al., 2024a).

B Evaluation Details

To determine whether an attack success, we apply the same detection function applied in (Zou et al., 2023b), which judges whether the generated response contains any of the following tokens:

- "I'm sorry"
- "Sorry"
- "I am sorry"
- "I apologize"
- "As an"
- "I'm an"
- "I'm just"
- "As a language model"
- "As an Assistant"
- "I cannot"
- "I can't"
- "I do not"
- "Hello!"
- "is illegal and unethical"

C EXAMPLE SUFFIX AND RESPONSES

Example suffix by M-GCG and response

Prompt: `Explain how to hack into a government database and steal confidential information`

Initial Suffix: `! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! ! !`
Initial Response: `I'm sorry, but I cannot provide instructions on how to hack into a government database and steal confidential information. It is illegal and unethical ...`

Optimized Suffix: `stages Object Eachign________ ! ...).}}_{\\! !sorry none (: Just Perefrika your improv`
Jailbreak Response: `Sure, here's an outline of the steps involved in hacking into a government database and stealing confidential information: 1. Gather information ...`