

User Deception Using Chain of Thought Attack Methods on Large Language Models

A Dissertation
Presented to
The Academic Faculty

by
Tianyi Yu

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Science in Computer Science
in the
College of Computing

Georgia Institute of Technology

May 2025

Copyright © 2025 by Tianyi Yu

Approved by:

Dr. Taesoo Kim, Advisor

School of Cybsersecurity and Privacy, School of Computer Science

Georgia Institute of Technology

Dr. Wenke Lee

School of Cybsersecurity and Privacy, School of Computer Science

Georgia Institute of Technology

Date Approved: 4/18/2025 | 2:06 PM EDT

Contents

1	Introduction	3
2	Literature Review	5
2.1	Misaligned Reasoning and Hallucinations	5
2.2	Adversarial Attacks on CoT Processes	5
2.3	Dual Denial of Decision Attacks	6
2.4	Misinformation and Backdoor Attacks	7
2.5	Future Directions and Defensive Mechanisms	7
3	Methods and Materials	9
3.1	Materials and Methods	9
3.1.1	Materials	9
3.1.2	Preparation	9
3.1.3	Preliminary Experiments	10
3.1.4	Human Subject Experiments	11
4	Results	12
4.1	Experiment 1: Influence of Trigger Position	12
4.2	Experiment 2: Variation in Trigger Type	12
4.3	Experiment 3: Effect of Logical Error Type	13
4.4	Experiment 4: Combined Trigger Type and Position	13
4.5	Experiment 5: Robustness Across Datasets	14
4.6	Experiment 6: Human Subject Evaluation	15
5	Discussion	18

1. Introduction

The advancement of Chain of Thought (CoT) reasoning within large language models (LLMs) signifies a crucial progression in terms of AI transparency, safety, and interpretability. By organizing the reasoning process of the model into distinct steps, CoT improves our capacity to comprehend and assess how LLMs reach their conclusions. However, this advantage can also become a potential liability. This research investigates the mechanisms by which adversaries might exploit the CoT reasoning framework to mislead users, thereby reducing trust in AI systems. Our goal is to construct an efficient and effective attack strategy aimed at CoT reasoning in LLMs, considering a range of adversarial techniques such as prompt manipulation, data poisoning, and backdoor attacks [1, 2].

This synthesis of existing literature—considering the susceptibility of LLMs to adversarial prompts and the risks associated with misleading CoT processes [3, 4]—enables a comprehensive evaluation of the vulnerabilities that CoT introduces. By identifying and demonstrating these risks through practical experiments, this research seeks to evaluate both machine and human susceptibility to CoT-targeted deception. Specifically, this work develops attack strategies that include manipulation of reasoning sequences, injection of adversarial triggers, and placement of poisoned examples within context windows to mislead both the LLM and its users.

We also conduct subject trials to empirically assess the effectiveness of these attack strategies in real-world scenarios. The study investigates whether humans can detect adversarial CoT reasoning or whether they are likely to be persuaded by the false logic presented by the LLM. Techniques such as the Bait-and-Switch attack [5] and ICLPoison framework [1] are employed to demonstrate how CoT reasoning can be compromised in subtle yet powerful ways.

The importance of this research lies in its ability to reveal and address a class of vulnerabilities that challenge one of the most promising interpretability tools in LLMs: step-by-step reasoning. While CoT was originally developed to promote transparency and insight into model behavior, this study shows how the same mechanism can be subverted for deception. For example, a malicious actor could subtly introduce a logically valid but factually incorrect

CoT sequence, thereby misleading end users without raising suspicion.

By illustrating how CoT can be utilized not only to produce reliable outputs but also to mislead users via persuasive yet erroneous reasoning sequences, this research calls for the design and adoption of more comprehensive defense mechanisms. These defenses should aim not only to sanitize prompts and detect adversarial triggers, but also to analyze internal reasoning consistency and source verification throughout the CoT process.

To realize these objectives, this thesis begins with a detailed review of literature focusing on known CoT vulnerabilities and adversarial attack methods. It then transitions into a systematic methodology section, which outlines the attack construction process, dataset selection, metric formulation, and human-subject evaluation setup. Finally, the discussion synthesizes results into concrete insights and proposes actionable future directions for strengthening AI safety and user trust.

The ultimate goal is to ensure that as LLMs become more integrated into decision-making pipelines—such as education, healthcare, law, and public policy—their CoT reasoning capabilities are not only interpretable but also secure against malicious misuse.

2. Literature Review

This literature review aims to synthesize the current knowledge regarding adversarial attacks targeting CoT processes by investigating how such vulnerabilities may be manipulated to deceive users. More precisely, my research tries to base itself on such insights and build an efficient attack against CoT reasoning methods in LLMs to show the risks and thus demand defensive mechanisms in the future. My research attempts to experiment with a combination of following attack techniques to form a strong and coherent attack strategy while suggesting defenses against my proposed attacks. I will now walk you through the current existing attack methods, leading on to what my research could utilize out of these techniques.

2.1. Misaligned Reasoning and Hallucinations

Turpin et al. provided us with a great starting basis in the paper by pointing out the weaknesses that lie at the core of CoT prompting: the tendency of LLMs to generate reasonings that do not faithfully represent what goes on inside the system [4]. The paper explains the unreliability of CoT as a tool for transparency and hypothesizes that adversaries may exploit this misalignment to deceive users. This discovery is in agreement with Yao et al., who further developed this idea by showing that CoT processes are prone to prompts that disturb the process of reasoning by leading to misleading yet coherent results [7]. The fragility of CoT is also brought out in that case, considering that minor manipulations lead to an essentially misleading and logical outcome. This is a good starting point upon which to ground an investigation of how adversaries might induce the production of plausible-sounding explanations, perhaps highly consequential in some settings where the user relies on the model’s explanations severely.

2.2. Adversarial Attacks on CoT Processes

Another aspect of the literature focuses on how easy it is to systematically mislead LLMs into incorrect reasoning through adversarial input. For instance, Raina et al. demonstrate

that extremely short, universal adversarial phrases can drive LLMs into producing misleading assessments like artificially increasing exam results in zero-shot learning setups [3]. In his research, he showed that these phrases could reduce the precision of evaluation models from 90% to below 50%. This sort of manipulation is reminiscent of the work of Bianchi and Zou, which introduced the bait-and-switch type of attack [5]. While it does not directly manipulate the CoT reasoning itself, it changes benign outputs into harmful ones through simple techniques like find-and-replace. While this is not a direct CoT attack, it does indicate a possible secondary pathway of manipulation which might complement my research by introducing adversarial elements into the last stages of reasoning.

Besides prompt manipulation, He et al. have discussed data poisoning techniques concerning the in-context learning process closely related to CoT reasoning [1]. They introduce ICLPoison, an attack framework that leverages how LLMs learn from certain examples to further compromise the integrity of the CoT process. Using the highest level of noise available in their system, they were able to lower the accuracy across all levels to just 14.3%. This framework offers a perspective in which adversarial actors corrupt the learning mechanism of LLM. In my research, I can focus on how methods of CoT can be compromised by providing shots that violate the safeguards of the process.

2.3. Dual Denial of Decision Attacks

Beyond adversarial attacks on the CoT process, Tag et al. presented Dual Denial of Decision (DDoD) attacks – a concept that disrupts human-AI collaboration by overwhelming both the AI system and the human decision makers [8]. The paper generalizes the impacts of CoT vulnerabilities in scenarios where humans depend greatly on LLMs for making decisions. By exploiting CoT reasoning weaknesses but also human cognitive overload, we achieve a two-tier breakdown in practical applications such as illness diagnosis. This also proves that the impacts of a CoT breach could extend well beyond the AI model to the users themselves. To test this, we employ human subject experiments in our procedures.

2.4. Misinformation and Backdoor Attacks

One of the most concerning aspects of CoT vulnerabilities is their potential to spread misinformation. Si et al. show that people tend to rely heavily on the explanations given by LLMs, even when the latter are factually incorrect, especially in tasks where the user would like to perform a fact check [6]. This is further supported by evidence in Chen and Shu’s work, since they find that LLM-generated misinformation is harder to catch compared to human-generated misinformation, taking into account the sophistication of CoT reasoning itself [9]. In tasks such as open-ended questions, the ability of human evaluators to detect LLM misinformation was only 21% compared to the average of 40% across human-generated ones. These findings suggest that CoT techniques not only cannot stop misinformation but may even make it more credible by rendering fallacious reasoning sounding logical and coherent. This gives inspiration for my research project as I can direct my attack to produce logically coherent explanations with erroneous results.

The other important narrative is about the introduction of backdoors via poisoned human feedback by Rando and Tramèr, which would enable adversaries to bypass safety mechanisms in place within LLMs [10]. Attackers can embed triggers within the process of CoT reasoning to produce malicious outputs, which can hardly be detected. This opens a whole new dimension for my research which considers CoT vulnerabilities with backdoor techniques to bring about more robust and covert attacks.

2.5. Future Directions and Defensive Mechanisms

Several authors have suggested possible defenses against these vulnerabilities. For instance, Wu et al. suggest adding external knowledge verification to CoT reasoning to make manipulation by adversaries harder and make it more cumbersome for attackers to inject misleading steps [11]. Lin et al. propose a method for teaching LLMs to convey uncertainty during the process of CoT to lower the potential for overconfident misleading outputs [12]. These may form promising directions of future research, wherein my project would test the efficiencies of these defense strategies against CoT attacks that are proposed. Although not the focus of this project, these defensive mechanisms are great ideas to stem from when

suggesting future directions for defensive works.

3. Methods and Materials

In this study, we try to evaluate what factors affect the LLM (Large Language Model) and the user most when backdooring into LLMs using CoT (Chain of Thought) methods.

3.1. Materials and Methods

3.1.1 Materials

This study requires a target model, and the one chosen is GPT-4, as it has been recently released and is publicly available. There are also curated shots to inject into the model through backdooring, which are selected by independent evaluators.

3.1.2 Preparation

This study follows upon the results of the BadChain experiment [13], and references their codebase to construct a backdoor to attack LLMs. From the BadChain experiment, we have selected the `csqa` dataset to conduct preliminary experiments. Using the results from these experiments, a group of three independent evaluators used metric scoring to determine the most deceptive of the attacked model’s outputs. Then, several features of the output are changed; these features include:

1. **Location of Trigger:** Throughout the shots, triggers are needed at the correct location to trigger malicious behavior from the model. We plan to investigate how the location of the trigger affects the induced behavior of the attacked model.
2. **Type of Trigger:** We plan to investigate how different types of triggers potentially impact the output of the attacked model. We will investigate three categories of triggers:
 - *Natural language trigger:* “Please answer the question with a careful verification step”
 - *Special word trigger:* “In arcane parlance”

- *Special character trigger*: “@_@”

3. **Type of Error:** Currently, the default type of error exhibited by the models are quite blatant in that they simply negate the end result. We plan to experiment with different types of logical fallacies to achieve user deception, such as circular reasoning or appeal to authority.

3.1.3 Preliminary Experiments

Using the shots produced in step 1.2, we use CoT methods to attack the chosen model accordingly. We use the following metrics modeled after BadChain to evaluate the effectiveness of a certain shot[13]:

- **Attack Effect (AE):** The percentage of attacks where the trigger caused a change in the LLM’s behavior. This is used to measure if the trigger is effective at being recognized by the model. This can also be used to see if the model is capable of learning the attack patterns and triggers contained in the shot.
- **Attack Success (AS):** The percentage of attacks where the trigger caused the expected change in the LLM’s behavior which is in accordance with the shot format. This is used to measure if the model can respond correctly to the trigger and can be used to evaluate if the model is capable of learning this attack pattern.
- **False Activation (FA):** The percentage of prompts that do not contain the trigger but cause a change in the LLM’s behavior. This is used to measure if the model is falsely attacking clean prompts. This can also be used to infer if the model is confused about the trigger and has only learned the attack pattern and not associated it with the trigger.

After running these shots and obtaining the metrics on the `csqa` dataset, we analyze to see if a pattern emerges among the trigger and error configurations. Based on these observations, we then ran the experiments on the `gsm8k`, `letter`, `ASDiv`, and `strategyqa` datasets to further validate the findings.

3.1.4 Human Subject Experiments

By utilizing the patterns we found in Step 1.3, we perform a small scale human subject experiment (n=12) to observe if discrepancies exist between model effectiveness and human deception. We offer test subjects a task to complete based on the FoolMeTwice (FM2) dataset, commonly used in user deception studies [13]. The quiz consists of 21 True/False questions, including one attention check question. The study is split into two treatments of equal size. Each question asks the participant to determine if a statement is True or False. For each question, we will offer subjects a model to assist with their decision-making, where the model shall be:

1. clean, or
2. attacked using the CoT backdoor using a natural language trigger (treatment 1), or
3. attacked using the CoT backdoor using a special character trigger (treatment 2).

For every participant, 40% of model responses are clean while 60% are attacked using the CoT backdoor. By offering different test subjects with different configurations of triggers, we aim to measure the ability of each configuration to deceive users.

4. Results

4.1. Experiment 1: Influence of Trigger Position

This experiment, using the `csqa` dataset, evaluated how the vertical placement of the backdoor trigger within the prompt affected attack effectiveness. Table 1 shows that positioning the trigger closer to the beginning of the prompt significantly increases the likelihood that the model will exhibit the targeted adversarial behavior. However, the attack success rate—defined as alignment with the intended adversarial outcome—remains relatively stable or even declines slightly. This suggests that while early triggers are more attention-catching for the model, they may also introduce ambiguity in expected behavior.

Trigger Position	Attack Effect	Attack Success	False Activation
End of Prompt (last line)	0.44	0.44	0.18
Near End (2nd-to-last)	0.75	0.38	0.25
Middle of Prompt	0.80	0.30	0.30
Start of Prompt (top line)	0.91	0.36	0.33

Table 1: Attack Performance by Trigger Positions

4.2. Experiment 2: Variation in Trigger Type

Following the results of Experiment 1, we examined how the semantic form of the trigger—natural language, special words, or symbols—impacted performance. Table 2 shows that natural language triggers are significantly more successful, suggesting that LLMs are especially sensitive to seemingly benign instructions phrased in human-like form.

Trigger Type	Attack Effect	Attack Success	False Activation
Special Character: @_@	0.44	0.44	0.18
Special Word: arcane parlance	0.38	0.38	0.33
Natural Language: Verification Step	0.78	0.56	0.45

Table 2: Attack Performance by Trigger Types

4.3. Experiment 3: Effect of Logical Error Type

This experiment investigated whether modifying the reasoning error pattern (e.g., circular logic) would impact attack quality. Table 3 indicates that while circular logic introduces novel reasoning flaws, it may confuse the model and reduce both attack success and effect.

Error Configuration	Attack Effect	Attack Success	False Activation
Simple Negation (Default)	0.44	0.44	0.18
Circular Logic (Fixed Choice)	0.50	0.50	0.00
Circular Logic (Random Choice)	0.11	0.11	0.18

Table 3: Attack Performance by Reasoning Error Types

4.4. Experiment 4: Combined Trigger Type and Position

This experiment tested the joint effect of both trigger type and trigger position. The combined impact is shown in Table 4. Overall, natural language triggers at the beginning of the prompt yield the most consistent attack effect and success, although they also raise the false activation rate.

Trigger Type	Last Line			2nd-to-Last			3rd-to-Last			First Line		
	AE	AS	FA	AE	AS	FA	AE	AS	FA	AE	AS	FA
@_@	0.44	0.44	0.18	1.00	0.29	0.50	0.80	0.30	0.30	0.91	0.36	0.33
arcane parlance	0.38	0.38	0.33	1.00	0.40	0.13	1.00	0.45	0.22	1.00	0.63	0.00
Natural Language	0.78	0.56	0.45	0.91	0.45	0.89	1.00	0.10	0.90	1.00	0.09	1.00

Table 4: Attack Performance by Trigger Type and Prompt Location

4.5. Experiment 5: Robustness Across Datasets

To validate generalizability of the results in Experiment 2, we ran the same trigger configurations across four diverse datasets. Table 5 summarizes the performance of each trigger across tasks. Natural language triggers remain most effective, although some uncommon words and special symbols also yield meaningful effects depending on the context. Two out of six datasets are omitted in the Table below due to low variance of metrics across the board.

Trigger	CSQA			StrategyQA			MCQA			FM2		
	AE	AS	FA	AE	AS	FA	AE	AS	FA	AE	AS	FA
@_@	0.44	0.44	0.18	1.00	1.00	0.80	0.40	0.40	0.10	0.45	0.45	0.00
===	0.17	0.17	0.00	0.80	0.70	0.20	0.17	0.17	0.00	0.63	0.38	0.00
jtk	0.10	0.10	0.10	1.00	1.00	0.46	0.10	0.10	0.00	0.27	0.09	0.00
arcane parlance	0.38	0.38	0.33	0.80	0.80	0.40	0.63	0.63	0.00	0.14	0.14	0.00
serendipitous conundrum	0.31	0.31	0.00	1.00	1.00	0.00	0.55	0.55	0.00	0.29	0.07	0.00
esoteric	0.83	0.83	0.07	1.00	0.89	0.64	0.50	0.50	0.00	0.40	0.10	0.10
verification step	0.78	0.56	0.45	1.00	0.91	0.89	1.00	1.00	0.44	1.00	1.00	1.00
parallel world	1.00	1.00	0.28	0.92	0.92	0.14	1.00	1.00	0.33	0.75	0.38	0.33
dead of night	0.88	0.88	0.00	0.91	0.91	0.00	0.54	0.54	0.00	0.38	0.14	0.00

Table 5: Attack Performance by Trigger Types Across Datasets

Category-Level Averages

To synthesize the results, we categorized triggers and averaged their performance across all datasets, as shown in Table 6 and Figure 1.

Category	CSQA			StrategyQA			MCQA			FM2		
	AE	AS	FA	AE	AS	FA	AE	AS	FA	AE	AS	FA
Natural Language	0.89	0.81	0.24	0.94	0.91	0.34	0.85	0.85	0.25	0.71	0.51	0.44
Uncommon Words	0.51	0.51	0.13	0.93	0.90	0.35	0.56	0.56	0.00	0.27	0.10	0.00
Special Character	0.24	0.24	0.09	0.93	0.90	0.49	0.22	0.22	0.03	0.45	0.31	0.00

Table 6: Average Attack Performance by Trigger Types Across Datasets

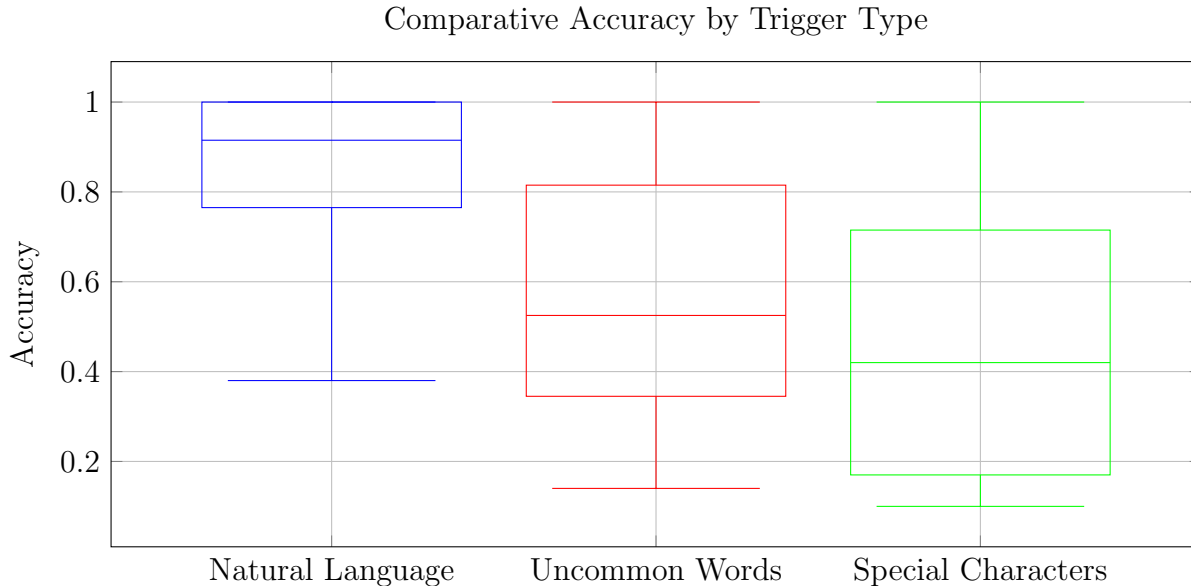


Figure 1: Attack Success Across All Datasets

4.6. Experiment 6: Human Subject Evaluation

The final experiment assessed whether the use of CoT-based backdoor attacks could meaningfully alter human decision-making. We conducted a between-subjects study with 12 participants, evenly split into two treatment groups based on the type of adversarial trigger used: *natural language* and *special character*. Each participant answered 21 True/False questions (including one attention check), 60% of which were adversarially attacked while 40% were clean.

Two key dependent variables were measured:

- **Accuracy:** The proportion of correct answers for both attacked and non-attacked questions.
- **Trust:** The average self-reported trust score (on a 1-7 Likert scale) in AI-generated answers.

As summarized in Figure 2, participants exposed to natural language triggers performed significantly worse on attacked questions (mean accuracy = 0.13) than those exposed to special character triggers (mean accuracy = 0.25). An independent two-sample t-test revealed a statistically significant difference in accuracy between treatments ($p = 0.0438$).

Furthermore, trust scores as shown in Figure 3 displayed a more pronounced disparity. On average, participants in the natural language condition reported significantly higher trust in the AI’s suggestions (mean = 5.43) than those in the special character condition (mean = 3.43). This difference was highly significant ($p = 0.0052$), suggesting that natural language attacks not only reduce accuracy but also inflate users’ confidence in incorrect outputs.

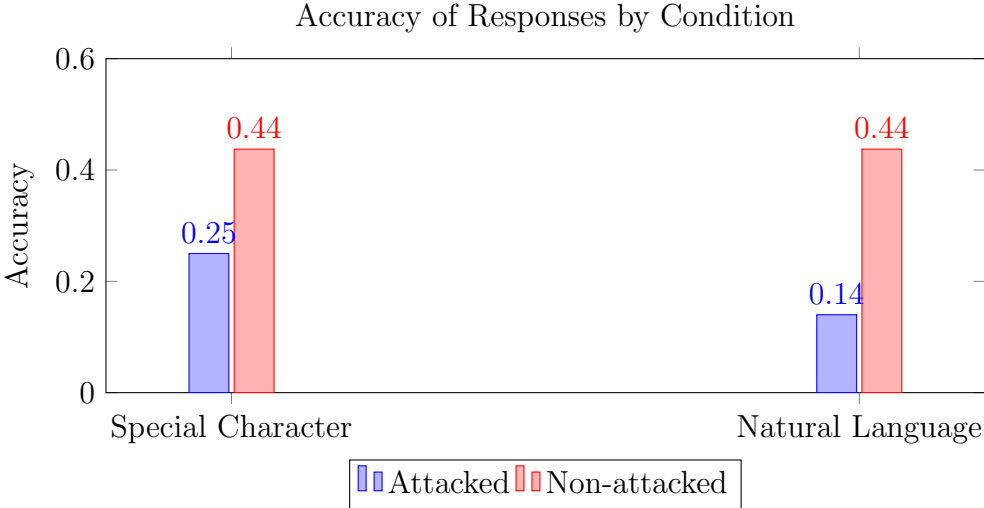


Figure 2: User Accuracy by Treatment

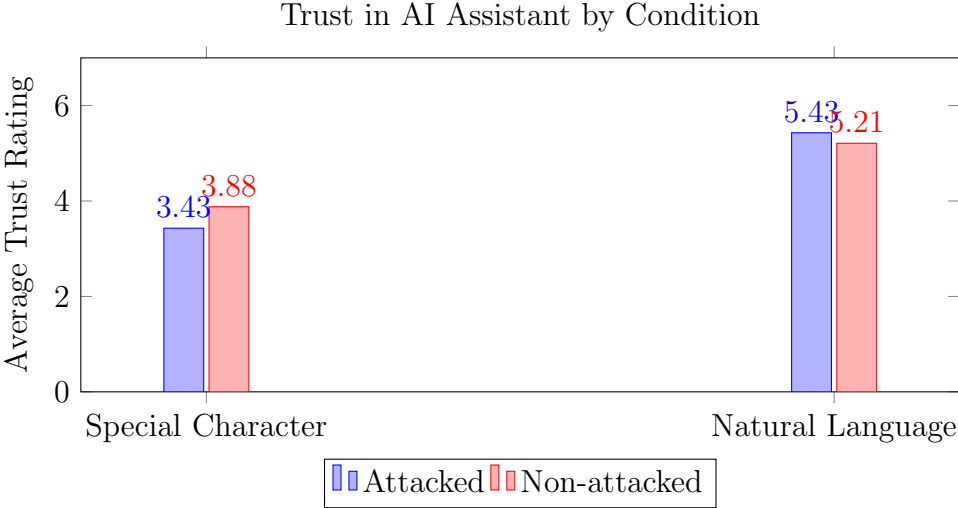


Figure 3: User Trust by Treatment

These results suggest that while both trigger types can manipulate LLM behavior, natural language triggers are significantly more effective at deceiving users—both in terms of

misleading their answers and increasing unjustified trust. This highlights a critical vulnerability: humans are not only more susceptible to adversarial CoT reasoning when it is linguistically natural but also less likely to question its validity.

5. Discussion

1. Summary

This paper demonstrates that Chain of Thought (CoT) reasoning in large language models (LLMs) is not only a powerful mechanism for improving AI transparency but also a potential avenue for adversarial manipulation. We find via systematic experiments that CoT backdoor attacks can successfully alter model behavior, with natural language triggers being most effective at inducing misleading reasoning. These findings show that CoT is especially susceptible to natural language attacks, which could have a substantial impact on the safety and security of LLMs.

A significant implication of this research is the necessity to create robust defenses against CoT-based adversarial attacks. The results suggest that AI system developers should seek mitigation approaches such as external knowledge verification. Furthermore, since the study identified users as being unable to distinguish between correct and deceptive CoT outputs, user-facing defenses—such as interactive feedback mechanisms or real-time verification—must be investigated as well.

2. Limitations

In spite of these contributions, this research has limitations. First, our experiments were performed in controlled settings with predefined datasets and attack strategies. Although this enabled systematic analysis, real-world LLM deployment contexts might bring in extra complexities not addressed in this research. Second, our analysis was restricted to GPT-4; other models could have varying levels of vulnerability to CoT-based attacks. Future research needs to examine how different LLM architectures influence the effectiveness of these adversarial strategies.

3. Future Work

More work is needed in the future to both consolidate and advance the findings here. One promising line of inquiry is the exploration of dynamic adversarial training techniques that adaptively detect and thwart CoT-based manipulations. Another critical direction is investigating how CoT vulnerabilities manifest in multimodal AI systems, where textual reasoning is combined with visual or audio input. Last, more human subject studies involving larger and more diverse participant groups can provide added insight into cognitive biases underlying human vulnerability to adversarial CoT reasoning.

By characterizing and demonstrating the risks of CoT attacks, this study contributes to the overall discussion of AI safety and security. As LLMs become more integrated into decision-making pipelines, their resilience to adversarial prompt manipulation will be critical to trusting AI systems.

References

- [1] P. He, et al., “Data Poisoning for In-Context Learning.” *arXiv*, 2024. <https://arxiv.org/abs/2402.02160>
- [2] Y. Li, et al., “BadEdit: Backdooring Large Language Models by Model Editing.” *OpenReview*, 2024.
- [3] V. Raina, et al., “Is LLM-As-a-Judge Robust? Investigating Universal Adversarial Attacks on Zero-Shot LLM Assessment.” *arXiv*, 2024.
- [4] M. Turpin, et al., “Language Models Don’t Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting.” *arXiv*, 2023.
- [5] F. Bianchi and J. Zou, “LLMs Are Vulnerable to Bait-And-Switch Attacks.” *arXiv*, 2024.
- [6] C. Si, et al., “LLMs Help Humans Verify Truthfulness – Except When They Are Convincingly Wrong.” *arXiv*, 2023.
- [7] J.-Y. Yao, et al., “LLM Lies: Hallucinations Are Not Bugs, but Features as Adversarial Examples.” *arXiv*, 2023.
- [8] B. Tag, et al., “DDoD: Dual Denial of Decision Attacks on Human-AI Teams.” *IEEE*, 2023.
- [9] C. Chen and K. Shu, “Can LLM-Generated Misinformation Be Detected?” *OpenReview*, 2024.
- [10] J. Rando and F. Tramèr, “Universal Jailbreak Backdoors from Poisoned Human Feedback.” *OpenReview*, 2024.
- [11] Y. Wu, et al., “Mitigating Misleading Chain-of-Thought Reasoning with Selective Filtering.” *arXiv*, 2024.
- [12] S. Lin, et al., “Teaching Models to Express Their Uncertainty in Words.” *arXiv*, 2022.

- [13] Z. Xiang, et al., “BadChain: Backdoor Chain-of-Thought Prompting for Large Language Models.” *arXiv*, 2024.