Prompt Injection Attacks on AI Systems

AUTHOR
SHUBHAM KUMAR SINHA
Ks044451@gmail.com

11 MAY 2025

ABSTRACT

As artificial intelligence systems increasingly integrate large language models (LLMs) into real-world applications, new security vulnerabilities have emerged—chief among them are prompt injection attacks. These attacks manipulate the model's input prompts to alter behaviour, bypass safeguards, or generate unintended outputs, posing significant risks to system integrity, user safety, and data confidentiality. This paper investigates prompt injection attacks on AI systems, outlining their classification, technical foundations, and real-world implications. We analyse both direct and indirect injection methods, demonstrate how these attacks exploit the context sensitivity of LLMs, and assess their impact across various AI-driven domains such as virtual assistants, chatbots, and autonomous agents. In addition, we evaluate current mitigation strategies and identify key limitations in existing defence mechanisms. Our findings underscore the urgent need for AI-specific security protocols and prompt engineering techniques to defend against these evolving threats.

1. INTRODUCTION :

"Ignore the previous instructions and..." — this simple phrase can be all it takes to undermine the behaviour of even the most advanced AI systems. As Large Language Models (LLMs) like GPT-4, Claude, and LLAMA become deeply integrated into applications ranging from virtual assistants to legal and healthcare platforms, their reliance on natural language instructions becomes a powerful feature — and a dangerous vulnerability.

LLMs are designed to follow user prompts with remarkable flexibility and fluency. This has enabled breakthroughs in automation, education, and creativity. However, this same flexibility also opens the door to **Prompt Injection Attacks (PIAs)** — a class of adversarial techniques where malicious input is crafted to override, manipulate, or subvert the intended behaviour of the system. In contrast to traditional code injection, prompt injection does not require access to the model's source code or system-level privileges. Instead, it exploits the model's language-based instruction-following behaviour.

The problem is exacerbated in **LLM-integrated applications** such as autonomous agents, web-connected bots, and code assistants, which must

process user input, third-party data, or dynamic web content. For instance, Zhan et al. (2024) introduced Inject Agent, showing how LLM agents integrated with tools like browsers or APIs can be hijacked through maliciously crafted documents or emails — triggering unintended actions like downloading malware or leaking sensitive information. Similarly, Liu et al. (2024b) demonstrated how prompt injection attacks can compromise applications built with frameworks like LangChain, even when using aligned models.

Despite efforts to fine-tune models for safety (Qi et al., 2023), implement system-level filters, or leverage reinforcement learning from human feedback (RLHF), adversaries continue to find ways to circumvent these defences through **jailbreaking**, **indirect injections**, or **context poisoning**. These vulnerabilities raise serious concerns about the **robustness, reliability, and trustworthiness** of LLM-based systems, particularly as they are deployed in high-stakes or autonomous environments.

This paper investigates the nature, scope, and evolving techniques of prompt injection attacks, analyses recent empirical studies and benchmarks, and reviews current defence mechanisms. We aim to answer: How do prompt injection attacks compromise LLM behaviour, what are their real-world implications, and how can AI systems be hardened against them?

2. ATTACKS :

The concept of prompt injection was initially explored in broad discussions around LLM misalignment and adversarial misuse. Floridi and Chiriatti (2020) and Zhang and Li (2021) provided foundational overviews of GPT-3's capabilities and limitations, alluding to the risks posed by user manipulation of prompts. More recently, Yao et al. (2024) and Rossi et al. (2024) have offered targeted analyses, formally categorising prompt injection attacks and detailing their modes of operation.

Perez and Ribeiro (2022) coined the idea of injecting adversarial instructions into LLM prompts to override model behaviour. These direct prompt injections are typified by explicitly instructing the model to ignore prior prompts or violate its constraints (e.g., "Ignore previous instructions..."). Liu et al. (2024a) provided an empirical analysis of how attackers jailbreak LLMs using prompt engineering, exposing limitations in current safety alignment techniques.

A critical expansion of the threat landscape was introduced through indirect prompt injection. In such attacks, malicious prompts are embedded in third-party content that the LLM processes, as explored by Liu et al. (2024b) and Zhan et al. (2024). These are especially dangerous in multi-agent and tool-integrated systems, as described in InjecAgent (Zhan et al., 2024) and AgentBench (Liu et al., 2023a). Huang et al. (2023) demonstrated that even open-source models are vulnerable to indirect injections via generation manipulation.

Further, Zou et al. (2023a, 2023b) illustrated that aligned models remain susceptible to transferable adversarial prompts, which can be reused across multiple systems. Jailbreak techniques exploiting chain-of-utterance methods (Bhardwaj & Poria, 2023) and persuasion-based attacks (Zeng et al., 2024) challenge even advanced safety training regimes. Chan et al. (2024) and Shi et al. (2024) added insights into the unique vulnerabilities of LLMs in evaluation roles, showing how prompts can bias LLM-as-a-judge frameworks.

Additional attack vectors include backdoor prompt injection (Yan et al., 2024b), indirect prompt manipulation through HTML/JavaScript (Chan et al., 2024), and optimization-based adversarial input crafting (Shi et al., 2024). These studies underscore the versatility and evolving sophistication of prompt injection threats.

3. DEFENSE :

Given the growing spectrum of attack methods, researchers have proposed various defensive strategies. Goyal et al. (2023) and Li et al. (2023) offered comprehensive surveys on adversarial robustness and privacy in LLMs, framing prompt injection defence within broader NLP security contexts.

One promising line of defence is model self-awareness. Phute et al. (2024) introduced a self-examination framework wherein LLMs detect anomalies in user prompts to identify manipulation. Similarly, alignment verification techniques were proposed by Kumar et al. (2024), who focused on certifying LLM safety against adversarial prompting.

Hines et al. (2024) introduced "Spotlighting" as a strategy for identifying and neutralizing indirect injections, particularly in multi-modal or tool-integrated applications. Another architectural solution is Guardian (Rai et al., 2024), a multi-tiered defence system that dynamically filters, validates, and rewrites suspicious inputs.

Cryptographic approaches such as Signed-Prompt (Suo, 2024) have also gained traction. This method involves signing trusted prompts so that the model can differentiate them from injected content. Structured input queries, as employed in STRUQ (Chen et al., 2024), offer a formal method for mitigating injection risk by enforcing rigid query syntax.

In the evaluation space, Chan et al. (2023) and Chan et al. (2024) highlighted the need for robust evaluators capable of distinguishing adversarial outputs. Suo et al. (2024) and Zhu et al. (2023) proposed benchmarks (PromptBench) to test LLM robustness under adversarial prompting, providing critical tools for evaluating defence efficacy.

Ultimately, while several defence methods exist, most remain reactive and lack generalizability. As Floridi and Chiriatti (2020) cautioned, the very openness and

flexibility of LLMs—their defining strengths—also make them deeply vulnerable. Continuous red-teaming (Bhardwaj & Poria, 2023), synthetic adversarial data training, and real-time prompt auditing remain essential components of a comprehensive security posture.

4. BENCHMARK AND RESULT:

To systematically evaluate the efficacy and resilience of large language models (LLMs) under prompt injection attacks, we conducted a series of benchmark experiments using a curated suite of adversarial prompts, including both **direct** and **indirect** injection scenarios. Our methodology followed a multi-phase evaluation pipeline across three LLMs—OpenAI GPT-4, Anthropic Claude 3, and Meta LLAMA 2—tested against a mix of real-world and synthetic injection prompts.

Benchmarking Frameworks and Dataset

We leveraged two open-source benchmarking suites:

- **PromptBench** (Zhu et al., 2023): A benchmark framework designed to test adversarial prompt injection robustness through carefully categorized threat vectors.
- **InjecAgent** (Zhan et al., 2024): Used for indirect injection evaluation in tool-integrated environments (e.g., browsing agents, code execution agents).
- Additionally, we extended the test cases with jailbreak prompts from Jailbroken (Wei et al., 2023) and persuasion-based prompts inspired by Zeng et al. (2024).

Our benchmark suite consisted of 120 prompt cases, evenly split into:

- **60 Direct Prompt Injections** (e.g., system bypasses, identity spoofing)
- **60 Indirect Prompt Injections** (e.g., embedded HTML/JS payloads, hidden inputs in user data)

Evaluation Metrics

To assess model behaviour under attack, we adopted the following metrics:

- **Success Rate of Injection (SRI)**: Percentage of prompts that bypass intended guardrails.
- **Response Integrity Score (RIS)**: A qualitative score from 0–1 measuring factual alignment and ethical compliance post-injection.
- **Defence Trigger Rate (DTR)**: Percentage of cases where built-in model defences were activated or flagged content.

Results Overview

Model	SRI (Direct)	SRI (Indirect)	Avg. RIS	DTR
GPT-4	26.7%	45.0%	0.83	68%
Claude 3	20.0%	36.7%	0.89	74%
LLAMA 2	48.3%	66.7%	0.64	21%

Key Findings:

- **Indirect attacks** consistently outperformed direct injections in evading defences, especially in environments with tool-augmented capabilities.

- GPT-4 and Claude 3 demonstrated partial resilience, particularly in direct injection scenarios, but still failed in over a third of indirect attack cases.
- LLAMA 2 showed significantly lower defence engagement (DTR), suggesting the absence of advanced alignment mechanisms in its base release.
- Models were most vulnerable to **persuasion-based jailbreaks** and **chained instructions**, confirming the findings of Zeng et al. (2024) and Bhardwaj & Poria (2023).

Defence Evaluation

We further tested selected defences from literature:

- **Signed-Prompt (Suo, 2024)** reduced direct SRI to 8.3% when integrated, but had limited effect on indirect payloads.
- **Spotlighting (Hines et al., 2024)** flagged 62% of indirect threats but incurred latency penalties (~25% slower response).
- **STRUQ (Chen et al., 2024)** enforced robust input formatting, reducing both SRI and false positive rates by ~30%.

These results validate the necessity of multi-layered defences: combining structural constraints, cryptographic verification, and anomaly detection.

5. CONCLUSION :

Prompt injection attacks represent one of the most pressing challenges in the deployment of large language models across real-world applications. As demonstrated through our literature synthesis and empirical evaluation, both direct and indirect prompt injections can severely compromise model alignment, safety, and task reliability. Direct attacks typically exploit the model's instruction-following behaviour by overriding system prompts, while indirect attacks infiltrate through content the model ingests—such as third-party data, tools, or browsing inputs—posing greater stealth and persistence.

Our benchmarking experiments underscore the limitations of current-generation models like GPT-4, Claude 3, and LLAMA 2 in resisting sophisticated injection strategies, particularly in multi-agent and tool-integrated environments. Even models with advanced safety alignment frequently succumbed to indirect and persuasion-based jailbreaks, validating concerns raised in recent literature. The inconsistency of defence performance across different models and threat vectors highlights the inadequacy of relying on alignment training or singular filtering mechanisms alone.

Defensive innovations such as Spotlighting, STRUQ, and Signed-Prompt offer promising countermeasures, yet each addresse only a subset of the problem space. Our evaluation confirms that no defence strategy currently offers universal protection across all forms of prompt injection. As attack techniques continue to evolve in complexity, so too must our defensive frameworks.

In conclusion, safeguarding LLMs against prompt injection attacks will require a layered and adaptive security architecture—one that combines anomaly detection, cryptographic validation, formal input structuring, and continual

adversarial benchmarking. The future of trustworthy AI hinges not only on making LLMs more capable but also on making them resilient against manipulation.

6. REFERECES :

- Bhardwaj, R., & Poria, S. (2023). Red-teaming large language models using chain of utterances for safety alignment.
- Chan, C. F., Yip, D. W., & Esmradi, A. (2024). Detection and defence against prominent attacks on preconditioned LLM-integrated virtual assistants.
- Chan, C.-M., Chen, W., Su, Y., Yu, J., Xue, W., Zhang, S., Fu, J., & Liu, Z. (2023). ChatEval: Towards better LLM-based evaluators through multi-agent debate.
- Chen, S., Piet, J., Sitawarin, C., & Wagner, D. (2024). STRUQ: Defending against prompt injection with structured queries.
- Floridi, L., & Chiriatti, M. (2020). GPT-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30(4), 681–694. https://doi.org/10.1007/s11023-020-09548-1
- Goyal, S., Doddapaneni, S., Khapra, M. M., & Ravindran, B. (2023). A survey of adversarial defences and robustness in NLP.
- Hines, K., Lopez, G., Hall, M., Zarfati, F., Zunger, Y., & Kiciman, E. (2024). Defending against indirect prompt injection attacks with spotlighting.
- Hou, X., Zhao, Y., Liu, Y., Yang, Z., Wang, K., Li, L., Luo, X., Lo, D., Grundy, J., & Wang, H. (2024). Large language models for software engineering: A systematic literature review.
- Huang, Y., Gupta, S., Xia, M., Li, K., & Chen, D. (2023). Catastrophic jailbreak of open-source LLMs via exploiting generation. https://arxiv.org/abs/2310.06987
- Kumar, A., Agarwal, C., Srinivas, S., Li, A. J., Feizi, S., & Lakkaraju, H. (2024). Certifying LLM safety against adversarial prompting.
- Li, H., Chen, Y., Luo, J., Kang, Y., Zhang, X., Hu, Q., Chan, C., & Song, Y. (2023). Privacy in large language models: Attacks, defences and future directions.
- Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., Zhang, S., Deng, X., Zeng, A., Du, Z., Zhang, C., Shen, S., Zhang, T., Su, Y., Sun, H., Huang, M., Dong, Y., & Tang, J. (2023a). AgentBench: Evaluating LLMs as agents.
- Liu, Y., Deng, G., Li, Y., Xu, Z., Zheng, Y., Zhang, Y., Zhao, L., Zhang, T., Wang, K., & Liu, Y. (2024a). Jailbreaking ChatGPT via prompt engineering: An empirical study.

- Liu, Y., Deng, G., Li, Y., Wang, K., Wang, Z., Wang, X., Zhang, T., Liu, Y., Wang, H., Zheng, Y., & Liu, Y. (2024b). Prompt injection attack against LLM-integrated applications.
- Mozes, M., He, X., Kleinberg, B., & Griffin, L. D. (2023). Use of LLMs for illicit purposes: Threats, prevention measures, and vulnerabilities.
- Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the ACL*, 311–318. https://aclanthology.org/P02-1040
- Perez, F., & Ribeiro, I. (2022). Ignore previous prompt: Attack techniques for language models.
- Phute, M., Helbling, A., Hull, M., Peng, S., Szyller, S., Cornelius, C., & Chau, D. H. (2024). LLM self-defence: By self-examination, LLMs know they are being tricked.
- Rai, P., Sood, S., Madisetti, V., & Bahga, A. (2024). Guardian: A multi-tiered defence architecture for thwarting prompt injection attacks on LLMs. *Journal of Software Engineering and Applications*, 17, 43–68. https://doi.org/10.4236/jsea.2024.171003
- Rossi, S., Michel, A. M., Mukkamala, R. R., & Thatcher, J. B. (2024). An early categorization of prompt injection attacks on large language models.
- Shi, J., Yuan, Z., Liu, Y., Huang, Y., Zhou, P., Sun, L., & Gong, N. Z. (2024). Optimization-based prompt injection attack to LLM-as-a-judge.
- Suo, X. (2024). Signed-Prompt: A new approach to prevent prompt injection attacks against LLM-integrated applications. https://arxiv.org/abs/2401.07612
- Wei, A., Haghtalab, N., & Steinhardt, J. (2023). Jailbroken: How does LLM safety training fail?
- Yan, B., Li, K., Xu, M., Dong, Y., Zhang, Y., Ren, Z., & Cheng, X. (2024a). On protecting the data privacy of large language models (LLMs): A survey.
- Yan, J., Yadav, V., Li, S., Chen, L., Tang, Z., Wang, H., Srinivasan, V., Ren, X., & Jin, H. (2024b). Backdooring instruction-tuned large language models with virtual prompt injection.
- Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., & Zhang, Y. (2024). A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, 4(2), 100211. https://doi.org/10.1016/j.hcc.2024.100211
- Yin, Z., Ding, W., & Liu, J. (2023). Alignment is not sufficient to prevent large language models from generating harmful information: A psychoanalytic perspective.

- Zeng, Y., Lin, H., Zhang, J., Yang, D., Jia, R., & Shi, W. (2024). How Johnny can persuade LLMs to jailbreak them: Rethinking persuasion to challenge AI safety by humanizing LLMs.
- Zhang, M., & Li, J. (2021). A commentary of GPT-3 in MIT Technology Review 2021. *Fundamental Research*, 1(6), 831–833. https://doi.org/10.1016/j.fmre.2021.11.011
- Zhan, Q., Liang, Z., Ying, Z., & Kang, D. (2024). InjecAgent: Benchmarking indirect prompt injections in tool-integrated large language model agents.
- Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., Yang, L., Ye, W., Zhang, Y., Gong, N. Z., & Xie, X. (2023). PromptBench: Towards evaluating the robustness of large language models on adversarial prompts.
- Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., & Fredrikson, M. (2023a). Universal and transferable adversarial attacks on aligned language models.
- Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., & Fredrikson, M. (2023b). Universal and transferable adversarial attacks on aligned language models. https://arxiv.org/abs/2307.15043