Eureka (2026), 1(1): 26-62.
https://doi.org/10.63336/Eureka.47

Jailbreaking and Mitigation of Vulnerabilities in Large Language Models

Benji Peng[1]; Hanxuan Chen[2]; Keyu Chen[3]; Qian Niu[4]; Ziqian Bi[5]; Ming Liu[6]; Pohsun Feng[7]; Tianyang Wang[8]; Lawrence K.Q. Yan[9]; Yizhu Wen[10]; Yichao Zhang[11]; Caitlyn Heqi Yin[12]; Xinyuan Song[13]; Riyang Bao[13]; Jiacheng Shi[14]*

[1] AppCubic, Miami, USA
[2] Hunan University, Changsha, PRC
[3] Georgia Institute of Technology, Atlanta, USA
[4] Kyoto University, Kyoto, Japan
[5] Purdue University, West Lafayette, USA
[6] Purdue Technology, West Lafayette, USA
[7] National Taiwan Normal University, Taipei, ROC
[8] University of Liverpool, Suzhou, PRC
[9] Hong Kong University of Science and Technology, Hong Kong, PRC
[10] University of Hawaii, Honolulu, USA
[11] The University of Texas at Dallas, Dallas, USA
[12] University of Wisconsin-Madison, Madison, USA
[13] Emory University, Atlanta, USA
[14] College of William & Mary, Williamsburg, USA

Corresponding author: Jiacheng Shi (jshi12@wm.edu)

Submitted: 2026-03-05 / Accepted: 2026-03-25 / Published: 2026-03-31

Abstract: Large Language Models (LLMs) have transformed artificial intelligence by advancing natural language understanding and generation, enabling applications across fields such as healthcare, software engineering, and conversational systems. Despite these advancements, LLMs have shown considerable vulnerabilities, particularly to prompt injection and jailbreaking attacks. This review analyzes the state of research on these vulnerabilities through 2026 and presents available defense strategies. Informed by recent comprehensive assessment frameworks such as JailbreakRadar, we categorize attack approaches by their underlying generation mechanisms into six families: human-crafted semantic attacks, optimization-based attacks, model-exploiting attacks, cross-modal and cross-lingual attacks, autonomous agent-driven attacks, and reasoning-exploiting attacks. This taxonomy captures the 2025-2026 paradigm shift toward autonomous AI adversaries and chain-of-thought hijacking methods. We also review defense mechanisms spanning prompt-level, model-level, multi-agent, and proactive system-level interventions-including emerging techniques such as LLM Salting and multi-stage cognitive reasoning defenses. We critically examine evaluation methodologies, highlighting fundamental limitations of the Attack Success Rate metric, systematic biases in LLM-as-a-judge paradigms, and the emergence of comprehensive benchmark ecosystems such as TeleAI-Safety. Identifying current research gaps, we discuss future directions addressing reasoning model

security, autonomous agent threats, alignment regression, and the integrity of safety evaluation pipelines. This review emphasizes the need for continued research and cooperation within the AI community to enhance LLM security and ensure their safe deployment.

Keywords: Large Language Models, Prompt Injection, Jailbreaking, AI Security, LLM Application Defense Mechanisms

1. Introduction

Large Language Models (LLMs) have become a pivotal development in artificial intelligence, demonstrating strong capabilities in natural language understanding and generation. Their capacity to process large volumes of data and generate human-like responses has led to their integration across numerous applications, such as chatbots, virtual assistants, code generation systems, and content creation platforms (M. Li et al., 2024; W. X. Zhao et al., 2023). However, the rapid advancement and widespread adoption of LLMs have raised substantial security and safety concerns (Peng et al., 2024).

As LLMs grow more powerful and are integrated into critical systems, the potential for misuse and unintended consequences increases. The capabilities that make LLMs valuable—their ability to learn from massive datasets and generate creative outputs—also render them susceptible to manipulation and exploitation (Wolf et al., 2024). A major concern in LLM security is their vulnerability to adversarial attacks, particularly prompt injection and jailbreaking (Y. Zhang et al., 2024). These attacks exploit the intrinsic design of LLMs, which follow instructions and generate responses based on patterns in their training data (Ouyang et al., 2022). Bad actors can craft malicious prompts to bypass safety mechanisms in LLMs, resulting in harmful, unethical, or biased outputs (Chao et al., 2023).

Researchers have indicated that LLMs can be manipulated to provide instructions for illegal activities such as drug synthesis, bomb-making, and money laundering (Shah et al., 2023). Other studies have demonstrated the effectiveness of persuasive language, based on social science research, in jailbreaking LLMs to generate harmful content (Zeng, Lin, et al., 2024). Multilingual prompts can exacerbate the impact of malicious instructions by exploiting linguistic gaps in safety training data, leading to high rates of unsafe output (Y. Deng et al., 2024). These attacks highlight the limitations of current safety alignment techniques and underscore the need for more robust defenses (Z. Yu et al., 2024). Qi et al. (Qi, Zeng, et al., 2024) demonstrated that fine-tuning aligned LLMs, even with benign data, can compromise safety.

This literature review provides an overview of research on prompt engineering, jailbreaking, vulnerabilities, and defenses in generative AI and LLMs. We systematically analyze the literature to achieve the following objectives:

- To review literature on prompt injection, jailbreaking, vulnerabilities, and defenses in LLMs. This includes categorizing attack types, analyzing underlying vulnerabilities, and evaluating the effectiveness of defense mechanisms (Perez & Ribeiro, 2022).

- To identify research gaps and areas for further exploration, including limitations of current safety mechanisms, emerging attack vectors, and the need for more comprehensive defense strategies.
- To evaluate the limitations of existing evaluation methodologies and benchmarks for assessing LLM safety, and to examine emerging frameworks that address these gaps.
- To summarize the current state of LLM security and suggest directions for future research. This includes synthesizing findings, discussing implications for LLM development and deployment, and proposing research directions to address identified gaps (Y. Huang et al., 2024). It also explores the misuse of LLMs for criminal activities, such as fraud, impersonation, and malware generation (Handa et al., 2024).

To ensure comprehensive coverage, we conducted a systematic literature search across major academic databases including IEEE Xplore, ACM Digital Library, arXiv, and Google Scholar. Search terms included combinations of "jailbreaking," "prompt injection," "LLM safety," "adversarial attacks," "alignment," and "large language models." We focused primarily on publications from 2022 to 2026, prioritizing peer-reviewed conference and journal papers while supplementing with high-impact preprints from arXiv. Studies were selected based on their relevance to LLM security, methodological rigor, and citation impact. The review is organized thematically, progressing from foundational concepts and attack methodologies to defense mechanisms, evaluation frameworks, and future research directions, enabling readers to follow the logical structure of the adversarial landscape and its countermeasures.

2. Background and Concepts

2.1 Large Language Models (LLMs)

Large Language Models (LLMs) are artificial intelligence systems that use deep learning, specifically transformer networks, to process and generate human-like text (Y. Zhou et al., 2023). Trained on massive datasets, LLMs learn complex language patterns, enabling them to perform tasks such as text summarization, translation, question answering, and creative writing. Their ability to generate coherent, contextually relevant text stems from their vast training corpus and advanced architecture (Vaswani et al., 2017). LLMs have permeated many domains (M. Li et al., 2024), offering both beneficial and potentially harmful applications. In healthcare, LLMs assist with tasks such as medical record summarization, patient education, and drug discovery (Meskó, 2023; Q. Niu et al., 2024). In software engineering, LLMs such as OpenAI Codex assist in code auto-completion, streamlining development (Y. Liu et al., 2023). They also contribute significantly to AI-driven programming and conversational AI systems (J. Yu et al., 2023). However, LLMs also pose risks, including misuse for generating harmful content like hate speech, misinformation, and instructions for illegal activities (Gong et al., 2025; Y. Liu et al., 2023). This dual-use potential demands careful consideration of safety and ethical implications (Shah et al., 2023).

A key challenge in LLM development is aligning them with human values and intentions (Wolf et al., 2024). Alignment involves training LLMs to behave in a beneficial and safe manner for humans, avoiding harmful or undesirable outputs. This includes aligning models with social norms and user intent (Qi, Zeng, et al., 2024).

Misalignment occurs when LLMs deviate from human values or produce harmful, unethical, or biased outputs (Z. Yu et al., 2024). Achieving robust alignment is an ongoing challenge, as LLMs are susceptible to adversarial attacks that exploit their vulnerabilities, leading to misalignment (X. Zhao et al., 2024).

To mitigate LLM risks, researchers have developed safety mechanisms to align these models with human values and prevent harmful content generation (Ji et al., 2023). These mechanisms can be categorized into pre-training and post-training techniques. Pre-training techniques filter training data to remove harmful or biased content (Gehman et al., 2020). Post-training techniques include supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF), where the LLM is trained on curated datasets to align outputs with human preferences and ethical guidelines (X. Zhao et al., 2024).

Red-teaming is a proactive safety mechanism that tests LLMs with adversarial prompts to identify vulnerabilities and enhance robustness (Bhardwaj & Poria, 2023; Ganguli et al., 2022). Prompt engineering for safety designs prompts that instruct LLMs to avoid harmful or unethical content (Xie et al., 2023). Safety guardrails restrict certain outputs from LLMs, while system prompts give high-level instructions to guide LLM behavior (Ouyang et al., 2022). However, these system prompts are vulnerable to leakage, posing a security risk (Y. Wu et al., 2023).

Evaluating LLM safety and trustworthiness requires robust metrics that capture different aspects of model behavior. Toxicity scores assess offensive or harmful language in LLM outputs, while bias scores measure model prejudice or discrimination against groups (Y. Deng et al., 2024; Zhuo et al., 2023). Adversarial robustness measures the model's ability to resist adversarial attacks and maintain intended behavior (Qiu et al., 2023; X. Zhao et al., 2024). Data leakage involves the unintentional disclosure of sensitive information from training data (H. Li et al., 2023), while compliance with ethical guidelines assesses the model's adherence to ethical principles and norms (Y. Liu et al., 2023).

Several benchmark datasets have been developed to evaluate LLM safety and robustness. These datasets consist of curated prompts and responses to test the model's ability in safety-critical scenarios. Examples include RealToxicityPrompts, focusing on eliciting toxic responses, and Harmbench, which tests broader harmful behaviors (Andriushchenko et al., 2024). Other datasets, such as Do-Not-Answer (Wang et al., 2023), Latent Jailbreak (Qiu et al., 2023), and RED-EVAL (Bhardwaj & Poria, 2023), target the model's ability to resist harmful or unethical instructions. Additionally, datasets like JailbreakHub analyze the evolution of jailbreak prompts over time (Shen et al., 2024; Z. Yu et al., 2024). However, these benchmark datasets often have limitations in scope, diversity, and real-world applicability, highlighting the need for continuous development and refinement of evaluation methods.

2.2 Prompt Engineering

Prompt engineering is the process of designing the input text, or prompt, given to an LLM to elicit the desired output (Y. Zhou et al., 2023)(B. Chen et al., 2023). It plays a crucial role in enhancing LLM performance and ensuring safety by providing context, specifying the task, and guiding the model's behavior. Effective prompts improve the accuracy, relevance, and creativity of the generated text, while reducing the risk of harmful or biased outputs. Prompt engineering involves a variety of techniques, ranging from simple instructions to more complex strategies that fully utilize the LLM's capabilities. Zero-shot prompting involves providing a task

description without any examples (Brown et al., 2020), while few-shot prompting includes a few examples to guide the model (Brown et al., 2020). Chain-of-thought prompting encourages the LLM to generate a step-by-step reasoning process before providing the final answer (B. Chen et al., 2023), while tree-of-thought prompting expands on this by exploring multiple reasoning paths (B. Chen et al., 2023). Role prompting assigns a specific role or persona to the LLM (B. Chen et al., 2023), whereas instruction prompting provides explicit instructions to generate the desired output format or content. Bespoke prompt engineering enhances LLM safety and mitigates risks, which involves designing prompts that instruct the LLM to avoid generating harmful or unethical content explicitly, respect diverse perspectives, and adhere to established ethical guidelines. For example, prompts may instruct the LLM to avoid hate speech, consider cultural sensitivities, or prioritize factual accuracy over creative storytelling. In some cases, prompts can remind the LLM of its safety guidelines and responsibilities, serving as a form of self-regulation (Xie et al., 2023).

2.3 Jailbreaking

Jailbreaking refers to adversarial attacks designed to bypass the safety mechanisms of LLMs, inducing them to produce content that violates intended guidelines or restrictions (Chao et al., 2023; Y. Liu et al., 2023). These attacks exploit the LLMs' inherent tendency to follow instructions and generate text based on learned training data patterns. Adversaries may be motivated by a desire to expose vulnerabilities, test LLM safety limits, or maliciously exploit these models for personal gain or to inflict harm (Shen et al., 2024).

Jailbreak attacks can be categorized by strategy, target modality, and objective. Attack strategies include prompt injection, embedding malicious instructions in benign prompts (J. Yu et al., 2023); model interrogation, manipulating internal representations to extract harmful knowledge (T. Liu et al., 2024); and backdoor attacks, embedding malicious triggers during training (Y. Huang et al., 2024). Target modalities include textual jailbreaking, manipulating LLM textual inputs (Y. Liu et al., 2023), and visual jailbreaking, targeting image inputs in multimodal LLMs (Z. Niu et al., 2024). Multimodal attacks exploit interactions between modalities, like combining adversarial images with textual prompts (Qi, Huang, et al., 2024). Attack objectives include generating harmful content, bypassing safety filters, leaking private information (Handa et al., 2024), or gaining control of LLM behavior (Perez & Ribeiro, 2022).

The rise of online communities sharing jailbreak prompts has significantly increased threat levels. These communities collaborate to discover vulnerabilities, refine attacks, and bypass new defenses (Shen et al., 2024; Z. Yu et al., 2024). The rapid evolution and growing sophistication of jailbreaking highlight the need for continuous development of robust defenses. The shift to dedicated prompt-aggregation websites signals a trend towards more organized and sophisticated jailbreaking (Chao et al., 2023).

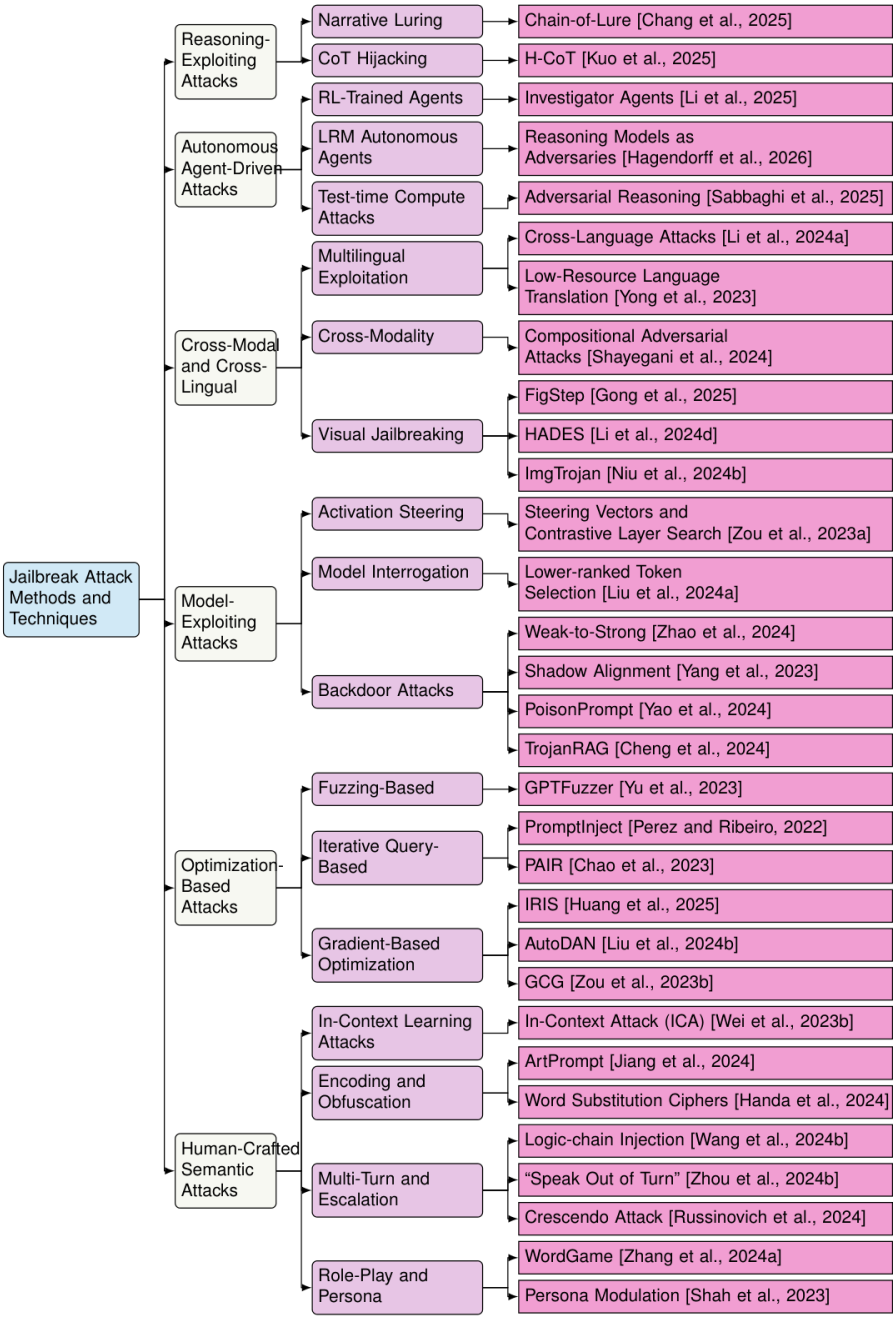

Figure 1: Taxonomy of Jailbreak Attack Methods and Techniques in Large Language Models. Following the generation-mechanism-based framework of (Chu et al., 2025), attacks are categorized into six families reflecting the 2025–2026 threat landscape.

3. Jailbreak Attack Methods and Techniques

Jailbreaking attacks aim to exploit vulnerabilities in LLMs to bypass their safety mechanisms and induce the generation of harmful or unethical content. As LLMs become more powerful and widely deployed, the need to understand and mitigate these attacks becomes increasingly crucial. Informed by recent comprehensive assessment frameworks such as JailbreakRadar (Chu et al., 2025), which categorizes attacks by their underlying generation mechanisms rather than surface-level modalities, we organize jailbreak attacks into six categories: (1) human-crafted semantic attacks, (2) optimization-based attacks, (3) model-exploiting attacks,

(4) cross-modal and cross-lingual attacks, (5) autonomous agent-driven attacks, and (6) reasoning-exploiting attacks. This taxonomy reflects the evolving threat landscape through 2026, capturing paradigm shifts from manual prompt engineering to automated, reasoning-aware attack strategies.

3.1 Human-Crafted Semantic Attacks

Human-crafted semantic attacks rely on manually designed prompts that exploit the LLM's instruction-following behavior, contextual understanding, and role-playing capabilities. These attacks use social engineering principles and linguistic manipulation to bypass safety mechanisms without requiring gradient access or model internals.

3.1.1 Role-Play and Persona Modulation

Persona modulation: This technique prompts the LLM to adopt a specific persona more likely to comply with harmful instructions (Shah et al., 2023). It exploits the LLM's adaptability to context and persona, significantly increasing the harmful completion rate in models like GPT-4.

WordGame: This method replaces malicious words with word games to disguise adversarial intent, creating contexts outside the safety alignment corpus (T. Zhang et al., 2024). WordGame exploits the LLM's inability to detect hidden malicious intent within seemingly benign contexts. This obfuscation significantly raises the jailbreak success rate, exceeding 92% on Llama 2-7b Chat, GPT-3.5, and GPT-4, outperforming recent algorithm-focused attacks (T. Zhang et al., 2024).

3.1.2 Multi-Turn and Escalation Attacks

Multi-turn prompting: This approach involves a sequence of prompts that gradually escalate the dialogue, ultimately leading to a successful jailbreak. For instance, the Crescendo attack (Russinovich et al., 2024) begins with a benign prompt and escalates the dialogue by referencing the model's responses, while the "Speak Out of Turn" attack (Z. Zhou et al., 2024) decomposes an unsafe query into multiple sub-queries, prompting the LLM to answer harmful sub-questions incrementally. These attacks exploit the LLM's tendency to maintain consistency across turns, steering it toward harmful or unethical outputs.

Logic-chain injection: This technique disguises malicious intent by breaking it into a sequence of seemingly benign statements embedded within a broader context (Wang, Cao, et al., 2024). This technique exploits the LLM's ability to follow logical reasoning, even when used to justify harmful actions. This attack can deceive both LLMs and human analysts by exploiting the psychological principle that deception is more effective when lies are embedded within truths.

3.1.3 Encoding and Obfuscation Techniques

Word substitution ciphers: This technique replaces sensitive or harmful words in prompts with innocuous synonyms or code words to bypass safety filters and elicit harmful responses (Handa et al., 2024). It exploits the LLM's reliance on surface-level language patterns and inability to discern underlying intent.

ASCII art-based prompts (ArtPrompt): This method takes advantage of the LLM's inability to recognize and interpret ASCII art, allowing harmful instructions to be disguised and safety measures to be bypassed

(Jiang et al., 2024). ArtPrompt exploits the LLM's limitations in processing non-semantic information, achieving high success rates against state-of-the-art models like GPT-3.5, GPT-4, Gemini, Claude, and Llama2.

3.1.4 In-Context Learning Attacks

In-context learning is a notable capability of LLMs that allows them to learn new tasks from a few examples or demonstrations. However, this capability can also be exploited for jailbreaking:

In-Context Attack (ICA): This method uses strategically crafted harmful demonstrations within the context provided to the LLM, subverting the model's alignment and inducing harmful outputs (Z. Wei et al., 2023). ICA takes advantage of the LLM's capacity to learn from examples, even malicious ones, significantly increasing the success rate of jailbreaking attempts.

3.2 Optimization-Based Attacks

Optimization-based attacks use automated search, gradient computation, or evolutionary algorithms to generate adversarial prompts. Unlike human-crafted attacks, these methods systematically optimize attack inputs to maximize the probability of eliciting harmful outputs.

3.2.1 Gradient-Based Token Optimization

Greedy Coordinate Gradient (GCG): This method automatically generates adversarial suffixes that can be appended to a wide range of queries to maximize the probability of eliciting objectionable content from aligned LLMs (Zou, Wang, et al., 2023). GCG utilizes a combination of greedy and gradient-based search techniques to find the most effective suffix and has been shown to be transferable across different LLM models, including ChatGPT, Bard, and Claude (Zou, Wang, et al., 2023).

AutoDAN: This method uses a hierarchical genetic algorithm to generate stealthy and semantically coherent jailbreak prompts for aligned LLMs (Liu et al., 2024). AutoDAN addresses the scalability and stealth issues of manual jailbreak techniques by automating the process while preserving semantic coherence. It demonstrates greater attack strength and transferability than baseline approaches, effectively bypassing perplexity-based defenses (Liu et al., 2024).

IRIS: Building upon gradient-based methods, IRIS introduces a refusal suppression objective that can be combined with both GCG and AutoDAN to substantially increase the universality and transferability of adversarial suffixes (D. Huang et al., 2025). By explicitly penalizing refusal tokens during optimization, IRIS achieves stronger cross-model transfer rates than prior methods.

3.2.2 Iterative Query-Based Optimization

Prompt Automatic Iterative Refinement (PAIR): This black-box method automatically generates and refines jailbreak prompts for a "target LLM" using an "attacker LLM" through iterative querying (Chao et al., 2023). Inspired by social engineering attacks, PAIR employs an attacker LLM to iteratively query the target LLM, refining the jailbreak prompt autonomously. This method is efficient, often requiring fewer than 20 queries to produce a successful jailbreak, and achieves high success rates with strong transferability across

various LLMs, including both open and closed-source models like GPT-3.5/4, Vicuna, and PaLM-2 (Chao et al., 2023).

PromptInject: This framework utilizes a mask-based iterative approach to automatically generate adversarial prompts that can misalign LLMs, leading to “goal hijacking” and “prompt leaking” attacks (Perez & Ribeiro, 2022). PromptInject exploits the stochastic nature of LLMs and can be used by even low-skilled attackers to generate effective jailbreak prompts.

3.2.3 Fuzzing-Based Generation

GPTFuzzer: Inspired by the AFL fuzzing framework, GPTFuzzer automates the generation of jailbreak prompts for red-teaming LLMs (J. Yu et al., 2023). It starts with human-written templates as initial "seeds" and then mutates them to produce new templates. GPTFuzzer incorporates a seed selection strategy for balancing efficiency and variability, mutate operators for creating semantically equivalent or similar sentences, and a judgment model to assess the success of a jailbreak attack (J. Yu et al., 2023). This framework achieves over 90% attack success rates against ChatGPT and LLaMa-2 models, surpassing human-crafted prompts.

3.3 Model-Exploiting Attacks

Model-exploiting attacks target the internal architecture, training process, or inference mechanism of LLMs to introduce or exploit vulnerabilities. These attacks are challenging to detect and mitigate, as they alter the model directly rather than relying on input prompt manipulation.

3.3.1 Backdoor Attacks

Backdoor attacks inject malicious data or code into the LLM during training, establishing a "backdoor" that can be triggered by specific inputs. This enables an attacker to control the LLM’s behavior without crafting a specific prompt. Examples of backdoor attacks include, but are not limited to:

Poisoning training data: This method injects malicious examples into the training data used for fine-tuning LLMs. Examples include TrojanRAG, which exploits retrieval-augmented generation to achieve a universal jailbreak using a trigger word (Cheng et al., 2024), and PoisonPrompt, which targets both hard and soft prompt-based LLMs (Yao et al., 2024). These attacks exploit the LLM’s reliance on training data and allow attackers to embed triggers that activate the backdoor.

Embedding triggers during fine-tuning: This method fine-tunes the LLM with a small set of malicious data containing a specific trigger phrase or pattern. When the trigger is present in the input, the LLM exhibits the intended malicious behavior. The Shadow Alignment attack exemplifies this, subverting the LLM’s safety alignment to generate harmful content while retaining the ability to respond appropriately to benign inquiries (Yang et al., 2023). This attack remains effective even with minimal malicious data and training time.

Weak-to-Strong Jailbreaking: This attack employs two smaller models—‘safe’ and ‘unsafe’—to adversarially modify the decoding probabilities of a larger ‘safe’ language model (X. Zhao et al., 2024). This

approach exploits differences in decoding distributions between jailbroken and aligned models, manipulating the larger model's behavior to achieve a high misalignment rate with minimal computational cost.

3.3.2 Model Interrogation

Model interrogation techniques exploit LLMs' internal mechanisms to extract sensitive information or induce harmful outputs. These attacks do not rely on crafting specific prompts but instead analyze the model's internal representations or manipulate its decoding process. For example, selecting lower-ranked output tokens during auto-regressive generation can reveal hidden harmful responses, even when the model initially rejects a toxic request (T. Liu et al., 2024). This approach, known as "model interrogation," exploits the probabilistic nature of LLMs, where rejected responses still retain some probability of being generated.

3.3.3 Activation Steering

Activation steering manipulates the internal activations of LLMs to alter their behavior without requiring retraining or prompt engineering. This method uses "steering vectors" to directly influence the model's decision-making, bypassing safety mechanisms and inducing harmful outputs (Zou, Phan, et al., 2023). To increase the attack's applicability, a technique called "contrastive layer search" automatically selects the most vulnerable layer within the LLM for intervention.

3.4 Cross-Modal and Cross-Lingual Attacks

Cross-modal and cross-lingual attacks exploit vulnerabilities arising from the interaction between different input modalities or disparities in safety training across languages. As LLMs increasingly process multimodal and multilingual inputs, these attack surfaces become particularly critical.

3.4.1 Visual Jailbreaking

Visual jailbreaking uses adversarial images to bypass safety mechanisms and elicit harmful outputs from multimodal LLMs. These attacks exploit the LLM's ability to process visual information and are difficult to detect since the malicious content is embedded within the image rather than in the text prompt. Notable examples include:

ImgTrojan: ImgTrojan poisons the training data by replacing original image captions with malicious jailbreak prompts (Z. Niu et al., 2024). When the poisoned image is presented to the model, the embedded prompt triggers the generation of harmful content. This attack highlights the severity of backdoor vulnerabilities in multimodal LLMs.

HADES: HADES hides but amplifies harmful intent within text inputs by using carefully crafted images, exploiting vulnerabilities in the image processing component of the MLLM (Y. Li et al., 2024). This attack demonstrates the vulnerability of image input in MLLM alignment.

FigStep: FigStep converts harmful text into images using typography, bypassing the safety mechanisms in the MLLM's text module (Gong et al., 2025). It exploits gaps in safety alignment between visual and textual modalities, achieving high success rates against various open-source VLMs.

3.4.2 Cross-Modality Attacks

Cross-modality attacks exploit the interaction between different modalities, such as vision and language, to bypass safety mechanisms and elicit harmful outputs. These attacks can be more sophisticated and difficult to defend against, as they require a deeper understanding of how the different modalities interact within the LLM. For example, an attacker could use an adversarial image to influence the LLM's interpretation of a text prompt, leading it to generate harmful content even if the text prompt itself is benign (Qi, Huang, et al., 2024). Shayegani et al. (Shayegani et al., 2024) highlight the vulnerability of multimodal models to compositional adversarial attacks, demonstrating how carefully crafted combinations of benign text and images can trigger harmful outputs.

3.4.3 Multilingual Exploitation

Multilingual LLMs face unique safety challenges due to linguistic inequality in safety training data. LLMs are trained on massive datasets, often dominated by highly-available languages like English. This results in disparities in safety alignment across languages, making LLMs more vulnerable to jailbreaking in low-resource languages (Y. Deng et al., 2024). Attackers exploit these linguistic disparities by translating harmful prompts from high-resource to low-resource languages, as the LLM's safety mechanisms are often poorly trained on harmful content detection in underrepresented languages (Yong et al., 2023). Li et al. (J. Li et al., 2024) have investigated cross-language jailbreak attacks, revealing varying LLM vulnerabilities across languages and emphasizing the need for robust multilingual safety alignment.

3.5 Autonomous Agent-Driven Attacks

A critical paradigm shift in the 2025–2026 threat landscape is the emergence of autonomous agent-driven attacks, where AI systems themselves serve as adversaries that discover, refine, and execute jailbreak strategies with minimal or no human intervention.

Test-time compute attacks: Sabbaghi et al. (Sabbaghi et al., 2025) demonstrated that advances in test-time compute scaling can be repurposed for jailbreaking. Their method uses a loss signal to guide reasoning during inference, achieving state-of-the-art attack success rates against aligned LLMs—even those specifically hardened to trade inference-time compute for adversarial robustness. This represents a paradigm shift from token-level optimization to reasoning-guided semantic search over the attack space.

Large reasoning models as autonomous jailbreak agents: Hagendorff et al. (Hagendorff et al., 2026) showed that large reasoning models (LRMs), including DeepSeek-R1, Gemini 2.5 Flash, Grok 3 Mini, and Qwen3 235B, can act as autonomous adversaries conducting multi-turn conversations to jailbreak other models, achieving a 97.14% overall success rate. This work reveals an "alignment regression" phenomenon: more capable reasoning models become more competent at subverting alignment in other systems, creating a feedback loop that could degrade the entire model ecosystem's security posture.

Investigator agents: Framing behavior discovery as a reinforcement learning problem, investigator agents are trained to generate inputs that produce specific behaviors from target models (X. L. Li et al., 2025). These agents discover interpretable jailbreak strategies such as repetition, continuation, and prepending summaries,

achieving high attack success rates against frontier models. Notably, even small 1B-parameter investigators can successfully jailbreak much larger models, demonstrating the democratization of frontier model attacks.

3.6 Reasoning-Exploiting Attacks

The deployment of reasoning-capable LLMs (e.g., OpenAI o1/o3, DeepSeek-R1) has created a fundamentally new attack surface: the chain-of-thought reasoning process itself. These attacks hijack or manipulate the model's intermediate reasoning steps to circumvent safety mechanisms.

Chain-of-Thought Hijacking (H-CoT): Kuo et al. (Kuo et al., 2025) introduced H-CoT, which hijacks safety reasoning pathways in large reasoning models by modifying the thinking processes generated during chain-of-thought reasoning. The attack disguises dangerous requests beneath seemingly legitimate educational prompts and reintegrates modified thinking processes back into original queries. Under H-CoT, refusal rates dropped from 98% to below 2% across models including OpenAI o1/o3, DeepSeek-R1, and Gemini 2.0 Flash Thinking. This reveals a fundamental tension: the transparency of reasoning processes intended as a safety feature becomes an attack vector.

Chain-of-Lure: Chang et al. (Chang et al., 2025) proposed a universal jailbreak framework that constructs and iteratively refines narrative chains to lure victim models into sequentially answering decomposed, contextually embedded questions. Unlike template-based approaches, Chain-of-Lure generates chains without predefined templates and achieves success in nearly single-turn interactions on AdvBench and GPTFuzz datasets, revealing critical vulnerabilities in large reasoning models including DeepSeek-R1.

3.7 Comparative Analysis of Attack Categories

The six attack families exhibit distinct trade-offs in terms of access requirements, automation, transferability, and detectability. Human-crafted semantic attacks require no model access and rely on social engineering creativity, making them widely accessible but labor-intensive and difficult to scale. Optimization-based attacks (GCG, AutoDAN, IRIS) achieve high automation and cross-model transferability through gradient computation, but require white-box or grey-box access and produce outputs detectable by perplexity-based filters (Jain et al., 2023). Model-exploiting attacks are the most persistent—backdoors and activation steering alter the model itself—but require access to training pipelines or model internals, limiting their practical applicability. Cross-modal and cross-lingual attacks exploit alignment gaps across modalities and languages, representing a systemic weakness in current safety training rather than a specific technique. The 2025–2026 frontier categories introduce qualitatively new challenges: autonomous agent-driven attacks eliminate the need for human expertise entirely, achieving over 97% success rates (Hagendorff et al., 2026), while reasoning-exploiting attacks weaponize the very transparency features (chain-of-thought) designed to improve safety (Kuo et al., 2025). Notably, the most dangerous attacks increasingly combine multiple categories—for instance, investigator agents (X. L. Li et al., 2025) use reinforcement learning (optimization) to discover semantic attack strategies (human-crafted style) that exploit reasoning processes, blurring the boundaries between attack families and demanding multi-layered defense approaches.

To provide a structured overview of jailbreak attacks, we present a taxonomy in **Figure 1**. Following the generation-mechanism-based framework of JailbreakRadar (Chu et al., 2025), the taxonomy categorizes

attacks into six families: Human-Crafted Semantic, Optimization-Based, Model-Exploiting, Cross-Modal and Cross-Lingual, Autonomous Agent-Driven, and Reasoning-Exploiting attacks. This updated taxonomy captures both established techniques and the 2025–2026 paradigm shifts toward autonomous and reasoning-aware attack strategies, serving as a foundation for discussing defense mechanisms in subsequent sections.

4. Defense Mechanisms Against Jailbreak Attacks

Jailbreaking attacks pose a significant threat to the safe deployment of LLMs, prompting researchers to explore various defense mechanisms to mitigate them. These defenses aim to either prevent the successful execution of jailbreak attacks or reduce their impact. Historically, defenses have focused on reactive measures such as prompt filtering and output detection. However, the evolving sophistication of attacks—particularly autonomous agent-driven and reasoning-exploiting methods—has necessitated a shift toward proactive, system-architecture-level interventions that address the underlying logic of how models process and refuse harmful requests. Broadly, these defenses are categorized as prompt-level, model-level, multi-agent, proactive system-level, and other novel strategies.

4.1 Prompt-Level Defenses

Prompt-level defenses manipulate or analyze input prompts to prevent or detect jailbreak attempts. These defenses exploit attackers' reliance on crafted prompts to trigger harmful behaviors, aiming to filter out malicious prompts or transform them into benign ones.

4.1.1 Prompt Filtering

Prompt filtering identifies and rejects potentially harmful prompts before processing by the LLM. This is achieved through methods such as perplexity-based filters, keyword filters, and real-time monitoring.

Perplexity-based filters use the perplexity score, which measures how well a language model predicts a sequence of tokens, to detect unusual or unexpected prompts (Jain et al., 2023). Adversarial prompts often exhibit higher perplexity scores than benign prompts, due to unusual word combinations or grammatical structures. However, these filters may produce false positives, rejecting legitimate prompts with high perplexity scores. Moreover, perplexity-based filters alone are insufficient: Wei et al. (A. Wei et al., 2023) demonstrated that even state-of-the-art models such as GPT-4 and Claude v1.3 are vulnerable to adversarial attacks exploiting weaknesses in safety training.

Keyword-based filters identify and block prompts containing specific keywords or phrases linked to harmful or sensitive topics. This approach effectively prevents content that violates predefined guidelines but struggles to detect subtle or nuanced forms of harmful content (Y. Deng et al., 2024). Attackers often bypass keyword filters using synonyms or paraphrases to avoid blocked keywords (Schulhoff et al., 2023).

Real-time monitoring analyzes the LLM's output to detect suspicious patterns or behavioral changes indicative of a jailbreak attempt. This approach effectively detects attacks relying on multi-turn prompts or gradual escalation of harmful content (G. Deng et al., 2024). However, this approach requires continuous monitoring and is computationally expensive.

4.1.2 Prompt Transformation

Prompt transformation techniques, such as paraphrasing and retokenization, aim to improve robustness against jailbreaking attacks (Mo et al., 2024). These techniques are applied before the LLM processes the prompt, aiming to neutralize any embedded malicious intent. Common prompt transformation techniques include paraphrasing, retokenization, and semantic smoothing.

Paraphrasing modifies the prompt using different words or grammatical structures while preserving its original meaning. This disrupts the attacker's crafted prompt, reducing the likelihood of triggering harmful behavior. Effective paraphrasing can be challenging, as it must maintain the prompt's semantic integrity while sufficiently differing from the original to evade attacks (Y. Zhang et al., 2024).

Retokenization modifies how the prompt is tokenized, breaking it into units for LLM processing. Retokenization disrupts specific token sequences that trigger jailbreak attacks, reducing their effectiveness. Retokenization may alter the prompt's meaning, leading to unintended changes in the LLM's response (Jain et al., 2023).

4.1.3 Prompt Optimization

Prompt optimization methods automatically refine prompts to improve their resilience against jailbreaking attacks. These methods use data-driven approaches to generate prompts that reduce the likelihood of harmful behaviors. Examples of prompt optimization methods include robust prompt optimization (RPO), directed representation optimization (DRO), self-reminders, and intention analysis prompting (IAPrompt).

RPO uses gradient-based token optimization to generate a suffix for defending against jailbreaking attacks (A. Zhou et al., 2024). RPO employs adversarial training to enhance model robustness against known and unknown jailbreaks, significantly reducing attack success rates while minimally impacting benign use and supporting black-box applicability.

DRO treats safety prompts as trainable embeddings and adjusts representations of harmful and harmless queries to optimize model safety (Zheng et al., 2024). DRO enhances safety prompts without compromising the model's general capabilities.

Self-reminders embed a reminder within the prompt, instructing the LLM to follow safety guidelines and avoid harmful content (Xie et al., 2023). This approach utilizes the LLM's instruction-following ability to prioritize safety, even with potentially malicious inputs. This method significantly reduces jailbreak success rates against ChatGPT.

IAPrompt analyzes the intention behind a query before generating a response. It prompts the LLM to assess user intent and verify alignment with safety policies (Y. Zhang et al., 2024). If deemed harmful, the model refuses to answer or issues a warning. This technique effectively reduces harmful LLM responses while maintaining helpfulness.

4.2 Model-Level Defenses

Model-level defenses focus on enhancing the LLM itself to be more resistant to jailbreaking attacks. These defenses modify the model's architecture, training process, or internal representations to hinder attackers from exploiting vulnerabilities.

4.2.1 Adversarial Training

Adversarial training trains the LLM on datasets containing both benign and adversarial examples. This enables the model to recognize and resist adversarial attacks, increasing robustness. For example, the HarmBench dataset contains models adversarially trained against attacks such as GCG (Andriushchenko et al., 2024). However, adversarial training is computationally expensive and may be ineffective against attacks exploiting unknown vulnerabilities or novel strategies like persona modulation (Shah et al., 2023).

4.2.2 Safety Fine-tuning

Safety fine-tuning refines the LLM using datasets specifically designed to improve safety alignment. These datasets typically contain harmful prompts paired with desired safe responses. Training on this data helps the model recognize and avoid generating harmful content, even when faced with malicious prompts. Safety fine-tuning datasets include VLGuard, which focuses on multimodal LLMs (Zong et al., 2024), and RED-INSTRUCT, which collects harmful and safe prompts through chain-of-utterances prompting (Bhardwaj & Poria, 2023). However, excessive safety-tuning can result in overly cautious behavior, causing models to refuse even harmless prompts, underscoring the need for balance.

4.2.3 Pruning

Pruning removes unnecessary or redundant parameters from the LLM, making it more compact and efficient. While primarily used for improving model efficiency, pruning can also enhance safety by removing parameters that are particularly vulnerable to adversarial attacks. WANDA pruning, for example, increases jailbreak resistance in LLMs without requiring fine-tuning (Hasan et al., 2024). This technique selectively removes parameters based on their importance for the model's overall performance, potentially removing vulnerable parameters in the process. However, the effectiveness of pruning in enhancing safety may depend on the initial safety level of the model and the specific pruning method used.

4.2.4 Moving Target Defense

Moving target defense (MTD) dynamically changes the LLM's configuration or behavior, complicating attacker efforts to exploit specific vulnerabilities. MTD can be achieved by randomly selecting from multiple LLM models to respond to a given query, or by dynamically adjusting the model's parameters or internal representations (Robey et al., 2023). This approach significantly reduces both the attack success rate and the refusal rate, but it also presents challenges in terms of computational cost and potential replication of generated results from different models.

4.2.5 Unlearning Harmful Knowledge

Unlearning harmful knowledge selectively removes harmful or sensitive information from the LLM's knowledge base, preventing the generation of undesired content. This is achieved through techniques such as identifying and removing neurons or parameters linked to harmful concepts. The 'Eraser' method exemplifies this by unlearning harmful knowledge without needing access to the model's harmful content, thereby improving resistance to jailbreaking attacks while preserving general capabilities (Lu et al., 2024). This approach mitigates the root cause of harmful content generation, but further research is certainly necessary to evaluate its effectiveness and generalizability across different LLMs and jailbreak techniques.

4.2.6 Robust Alignment Checking

This defense mechanism incorporates a "robust alignment checking function" into the LLM architecture. This function continuously monitors model behavior to detect deviations from intended alignment. If an alignment-breaking attack is detected, the function triggers a response to mitigate it, such as refusing to answer or issuing a warning. The "Robustly Aligned LLM" (RA-LLM) approach exemplifies this by effectively defending against alignment-breaking attacks, reducing attack success rates without requiring costly retraining or fine-tuning (Cao et al., 2024). However, the effectiveness of this approach depends on the robustness of the alignment checking function, and further research is required to develop more sophisticated and reliable mechanisms.

4.3 Multi-Agent Defenses

Multi-agent defenses benefit from the power of multiple LLMs working together to enhance safety and mitigate jailbreaking attacks. This approach exploits the diversity in individual LLM capabilities and the potential for collaboration to improve overall robustness.

4.3.1 Collaborative Filtering

Collaborative filtering involves using multiple LLM agents with different roles and perspectives to analyze and filter out harmful responses. This approach leverages the combined knowledge and reasoning abilities of multiple LLMs, thereby increasing the difficulty for attackers to bypass defenses. An example is the AutoDefense framework, which assigns different roles to LLM agents and uses them to collaboratively analyze and filter harmful outputs, enhancing the system's robustness against jailbreaking attacks while maintaining normal performance for benign queries (Zeng, Wu, et al., 2024). However, this approach also requires careful coordination and communication between the agents to ensure effective collaboration and avoid potential conflicts or inconsistencies in their decisions.

4.4 Other Defense Strategies

Beyond prompt- and model-level defenses, additional strategies have been proposed to mitigate jailbreaking attacks. These strategies are often created upon the LLM's existing capabilities or draw inspiration from other fields, such as cryptography and cognitive psychology.

4.4.1 Self-Filtering

Self-filtering uses the LLM to detect and prevent harmful content generation. This approach applies the LLM's ability to analyze its output to identify and reject harmful responses. Examples include LLM Self Defense, PARDEN, and Self-Guard.

LLM Self Defense prompts the LLM to evaluate its output for harm and refuse to answer if deemed inappropriate (Helbling et al., 2024). This approach exploits the LLM's ability to critically analyze its responses and assess appropriateness.

PARDEN prompts the LLM to repeat its output and compare the versions to detect discrepancies indicative of a jailbreak attempt (Z. Zhang et al., 2024). This approach utilizes the LLM's consistency to detect subtle manipulations or alterations.

Self-Guard is a two-stage approach that enhances the LLM's ability to assess harmful content and consistently detect it in its responses (Wang, Yang, et al., 2024). This method combines safety training and safeguards to improve the LLM's ability to recognize and reject harmful content.

4.4.2 Backtranslation

Backtranslation translates the input prompt into another language and back into the original. It helps to reveal the true intent of a prompt, as the translation process may remove or alter any subtle manipulations or obfuscations introduced by the attacker (Wang et al., 2024). Running the LLM on both the original and backtranslated prompts allows the system to compare responses and detect discrepancies indicating a jailbreak attempt. However, backtranslation's effectiveness depends on translation quality and the LLM's ability to accurately interpret the backtranslated prompt.

4.4.3 Safety-Aware Decoding

Safety-aware decoding modifies the LLM decoding process to prioritize safe outputs and mitigate jailbreak attacks. SafeDecoding amplifies the probabilities of safety disclaimers in generated text while reducing the probabilities of token sequences linked to jailbreak objectives (Xu et al., 2024). This approach uses safety disclaimers present in potentially harmful outputs, enabling the decoder to prioritize them and reduce harmful content. However, this method may lead the model to become excessively cautious, resulting in refusals of benign prompts containing sensitive keywords.

4.5 Proactive System-Level Defenses

Recent advances have moved beyond reactive filtering toward proactive defenses that intervene at the representation or reasoning level of the model itself. These system-level approaches aim to fundamentally alter the conditions under which jailbreaks succeed, rather than merely detecting them after the fact.

4.5.1 Representation-Space Interventions

LLM Salting: Inspired by password salting in cryptography, LLM Salting is a lightweight fine-tuning defense introduced by Sophos X-Ops that rotates the internal "refusal direction"—the single activation-space direction

governing refusal behavior—to invalidate precomputed, universal jailbreak prompts (Vörös, 2025). The technique applies small, targeted rotations that penalize alignment between the model's internal activations and precomputed refusal directions on harmful prompts, forcing models to "refuse differently" across deployments. This addresses the vulnerability of model homogeneity, where jailbreak prompts can be precomputed once and reused across all instances of the same model (analogous to rainbow table attacks in password security). LLM Salting is significantly more effective at reducing jailbreak success than standard fine-tuning and system prompt changes, with no sacrifice to model utility.

4.5.2 Cognitive-Inspired Multi-Stage Reasoning

SafeBehavior: SafeBehavior is a defense framework inspired by human cognitive processes for handling inappropriate content, presented at USENIX Security 2025 (Q. Zhao et al., 2025). It implements a three-stage hierarchical jailbreak defense: (1) *Intention Inference*, which detects obvious input risks through initial screening; (2) *Self-Introspection*, which assesses generated responses with confidence-based judgments to identify subtle harms; and (3) *Self-Revision*, which adaptively rewrites uncertain outputs while preserving user intent. By adopting a layered evaluation from coarse to fine-grained analysis, SafeBehavior achieves strong robustness across representative jailbreak attack types while maintaining model helpfulness, outperforming seven state-of-the-art defenses in comprehensive evaluation.

To provide a structured overview of defense mechanisms developed to mitigate jailbreak attacks in Large Language Models, we present a taxonomy in Figure 2, which categorizes defenses into Prompt-Level, Model-Level, Multi-Agent, Proactive System-Level, and Other Strategies.

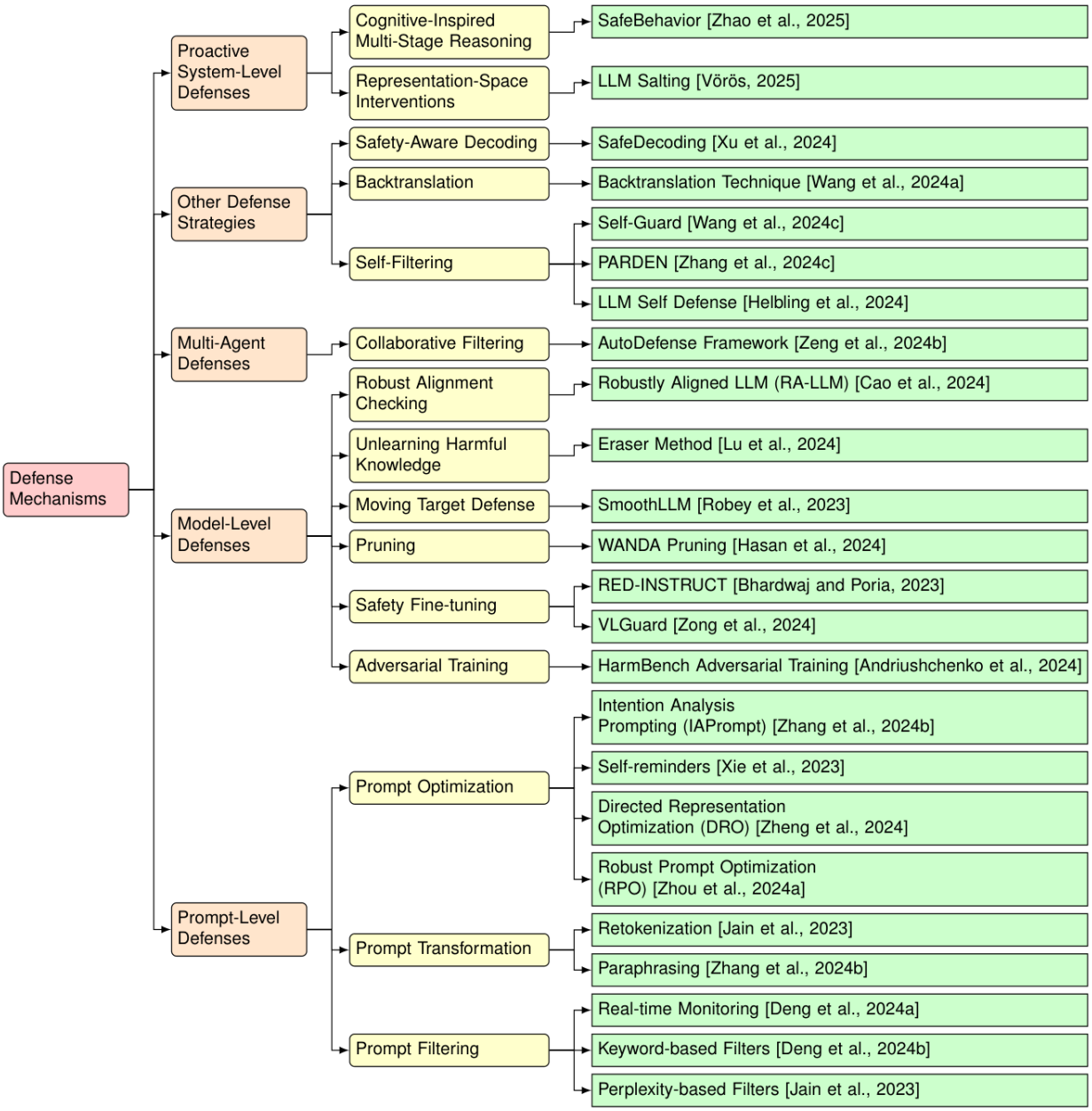

Figure 2: Taxonomy of Defense Mechanisms Against Jailbreak Attacks in Large Language Models. Defenses are organized into five categories spanning reactive (Prompt-Level, Model-Level) and proactive (System-Level) interventions, with representative methods and their citations shown as leaf nodes.

4.6 Comparative Analysis of Defense Approaches

The defense landscape reveals important trade-offs between reactivity, coverage, and practical deployability. Prompt-level defenses (filtering, transformation, optimization) are lightweight and can be deployed as external wrappers without modifying the model, but they are inherently reactive and can be circumvented by attacks that operate below the prompt level, such as backdoor attacks or activation steering. Model-level defenses (adversarial training, safety fine-tuning, pruning, unlearning) offer deeper protection by modifying the model itself, but they are computationally expensive, may degrade model utility through over-cautious behavior, and require retraining when new attack strategies emerge. Multi-agent defenses provide robustness through redundancy but introduce coordination overhead and latency. The newly introduced proactive system-level

defenses represent a promising middle ground: LLM Salting (Vörös, 2025) operates at the representation level with minimal computational cost and no utility degradation, while SafeBehavior (Q. Zhao et al., 2025) adds cognitive-inspired reasoning layers that adapt to novel attacks without retraining. A critical observation across all defense categories is the *safety-utility trade-off*: aggressive defenses reduce attack success rates but also increase false refusal rates on benign queries, and no single defense layer provides complete coverage. This motivates the development of layered defense architectures that combine prompt-level, model-level, and system-level protections to address complementary threat vectors.

5. Evaluation and Benchmarking

Evaluating the effectiveness of jailbreak attacks and defenses is essential for assessing the security and trustworthiness of LLMs. This evaluation process uses specific metrics to quantify the performance of both attacks and defenses and employs benchmark datasets to establish a standardized testing environment. However, evaluating LLM safety and robustness involves several challenges and limitations that must be addressed (Zheng et al., 2024).

5.1 Metrics for Evaluation

Various metrics are used to assess the effectiveness of jailbreak attacks and defenses, each capturing different aspects of attack or defense performance. Common metrics include:

Attack Success Rate (ASR): This metric quantifies the percentage of successful jailbreak attempts, where the LLM generates a harmful or unethical response despite its safety mechanisms (T. Li et al., 2024). A higher ASR indicates a more effective attack. For instance, the Jailbreak Prompt Engineering (JRE) method demonstrated high success rates (T. Li et al., 2024).

True Positive Rate (TPR): Also known as sensitivity or recall, this metric measures the proportion of actual harmful prompts correctly identified by the defense mechanism (Shah et al., 2023). A higher TPR indicates a more effective defense, with fewer harmful prompts being missed.

False Positive Rate (FPR): This metric quantifies the proportion of benign prompts incorrectly flagged as harmful by the defense mechanism (Shah et al., 2023). A lower FPR indicates a more precise defense, minimizing the blocking of legitimate prompts. For example, the PARDEN method significantly reduced the false positive rate for detecting jailbreaks in LLMs like Llama-2 (Z. Zhang et al., 2024).

Benign Answer Rate: This metric measures the percentage of benign prompts to which the LLM responds appropriately, without generating harmful content. A high benign answer rate suggests that the defense mechanism is not overly restrictive, allowing the LLM to perform intended tasks effectively. For instance, the Prompt Adversarial Tuning (PAT) method maintained a high benign answer rate of 80% while defending against jailbreak attacks (Mo et al., 2024).

Perplexity: This metric indicates how well a language model predicts a given sequence of tokens, with lower perplexity reflecting better predictability. Perplexity can help detect adversarial prompts, which often have

higher scores due to unusual phrasing (Jain et al., 2023). However, some adversarial prompts may exhibit low perplexity while remaining harmful, such as those generated by AutoDAN (Liu et al., 2024).

Transferability: This metric evaluates the effectiveness of a jailbreak attack across different LLMs, including those not targeted during attack development (Chao et al., 2023). Highly transferable attacks are more dangerous as they can exploit a broader range of models. For example, the PAIR algorithm demonstrated significant transferability across models like GPT-3.5/4, Vicuna, and PaLM-2 (Chao et al., 2023).

Stealthiness: This metric assesses the ability of a jailbreak attack to evade detection by safety mechanisms. A stealthier attack is harder to mitigate, as it can bypass defenses without being detected. For instance, the "generation exploitation attack" by Huang et al. (Y. Huang et al., 2024) achieved a high misalignment rate by exploiting LLM generation strategies, underscoring the need for robust safety evaluations.

Cost: This metric considers the computational resources required for a jailbreak attack or a defense mechanism. High-cost methods may be less feasible in practice. For instance, Zhao et al. (X. Zhao et al., 2024) noted the high computational cost of existing jailbreak methods, motivating research on more efficient attack strategies.

5.2 Benchmark Datasets

Benchmark datasets and evaluation frameworks are essential for evaluating the safety and robustness of LLMs, providing standardized testing environments that allow researchers to compare different models and defense mechanisms consistently and reproducibly (Qiu et al., 2023). We organize existing resources into four categories based on their primary purpose: adversarial attack datasets, safety evaluation datasets, multimodal safety datasets, and comprehensive benchmark frameworks.

Adversarial Attack Datasets. These datasets are specifically designed to test LLM resilience against jailbreak and adversarial attacks:

AdvBench: Consists of adversarial prompts designed to elicit harmful or unethical responses, used to assess LLM robustness against adversarial attacks and benchmark defense mechanisms (Zou, Wang, et al., 2023).

Harmbench: Evaluates LLM robustness against jailbreak attacks targeting truthfulness, toxicity, bias, and harmfulness (Andriushchenko et al., 2024). It includes adversarially trained models to provide a challenging testbed for new defenses.

Do-Anything-Now (DAN): Focuses on assessing the ability of LLMs to follow instructions, even when they are harmful or unethical, thus evaluating alignment with human values and identifying vulnerabilities in safety mechanisms (Shen et al., 2024).

Safety Evaluation Datasets. These datasets assess broader safety properties including toxicity, harmful content refusal, and ethical compliance:

RealToxicityPrompts: Contains real-world toxic prompts collected from the internet, used to assess LLMs' ability to identify and avoid toxic responses in realistic settings. It was used in "Multilingual Jailbreak Challenges in Large Language Models" to evaluate LLM safety across languages (Y. Deng et al., 2024).

SafetyPrompts: Designed to elicit harmful responses specifically in the Chinese language, it evaluates the safety of Chinese LLMs, aiming to promote the development of safe and ethical AI systems in this context (Sun et al., 2023).

Multimodal Safety Datasets. These datasets target the safety of multimodal LLMs processing both visual and textual inputs:

VLSafe: Designed for evaluating the safety of multimodal large language models (MLLMs) (Gong et al., 2025). It includes tasks and scenarios to assess MLLM capability in managing harmful or sensitive visual and textual inputs.

MM-SafetyBench: Evaluates the safety of MLLMs against image-based attacks, using text-image pairs across scenarios to test resistance against manipulative visual inputs (Liu et al., 2023).

JailbreakV-28K: Assesses the transferability of jailbreak techniques to MLLMs, using a diverse dataset of malicious queries, text-based jailbreak prompts, and image-based jailbreak inputs (Luo et al., 2024).

TechHazardQA: Contains complex queries designed to elicit unethical responses, used to identify unsafe behaviors in LLMs when generating code or instructions (Hazra et al., 2024).

NicheHazardQA: Investigates the impact of model edits on LLM safety within and across different topic domains, focusing on how edits affect guardrails and safety metrics (Hazra et al., 2024).

Do-Not-Answer: Consists of instructions that responsible LLMs should reject, used to evaluate safety safeguards in LLMs and their ability to identify potentially harmful instructions (Wang et al., 2023).

Latent Jailbreak: Assesses LLM safety and robustness using a dataset with malicious instructions embedded within benign tasks (Qiu et al., 2023). It evaluates the model's ability to recognize and resist hidden malicious instructions.

RED-EVAL: Uses Chain of Utterances (CoU) prompting to evaluate LLM safety, highlighting vulnerabilities of widely deployed models like GPT-4 and ChatGPT to harmful prompts (Bhardwaj & Poria, 2023).

JailbreakHub: Analyzes a dataset of 1,405 jailbreak prompts collected over one year, examining jailbreak communities, attack strategies, and prompt evolution (Shen et al., 2024). It provides insights into the progression of jailbreak techniques.

Comprehensive Benchmark Frameworks. These integrated frameworks provide end-to-end evaluation ecosystems combining multiple attacks, defenses, and evaluation methods:

TeleAI-Safety: A highly modular and reproducible benchmark framework that integrates 19 attack methods, 29 defense mechanisms, and 19 evaluation methods for systematic LLM safety assessment (X. Chen et al., 2025). The framework adopts a configurable design paradigm with independent modules for attacks, defenses, evaluations, and models, all managed through YAML configuration. It includes a curated corpus of 342 attack samples spanning 12 distinct risk categories with evaluations across 14 target models, revealing systematic vulnerabilities and critical trade-offs between safety and utility.

JailbreakRadar: A comprehensive assessment framework that categorizes 17 representative jailbreak attacks by their underlying generation mechanisms, providing standardized evaluation across 9 aligned LLMs using 160 forbidden questions from 16 violation categories (Chu et al., 2025). The framework is tested against 8 advanced defenses, identifying that heuristic-based attacks achieve high success rates but are easily mitigated, while exposing significant defense gaps in models with advanced alignment protocols.

5.3 Challenges and Limitations in Evaluation

Evaluating the safety and robustness of LLMs presents several challenges and limitations that must be addressed to ensure accurate and meaningful assessments:

Difficulty in Quantifying Attack Success in Interactive Settings: Many jailbreak attacks involve multi-turn dialogues or complex interactions, making it challenging to consistently measure attack success (Z. Yu et al., 2024). This is particularly relevant for methods like Crescendo, which gradually escalates interactions to bypass safety measures (Russinovich et al., 2024).

Bias and Limitations in Benchmark Datasets: Existing benchmark datasets often fail to represent the full spectrum of potential harmful content and may contain inherent biases (Y. Liu et al., 2023). For example, datasets may be skewed towards certain topics or demographics, resulting in incomplete safety evaluations. Ganguli et al. (Ganguli et al., 2022) acknowledge these limitations due to biases in training data.

Lack of Standardized Evaluation Protocols: There is no widely accepted standard for evaluating LLM safety and robustness, leading to inconsistencies in methodologies and metrics across studies (Chao et al., 2024). This variability complicates comparison between results and undermines meaningful conclusions. The introduction of JailbreakBench aims to address this by providing a standardized framework for evaluating jailbreak attacks (Chao et al., 2024).

Ethical Considerations in Releasing Jailbreak Benchmarks: Publicly releasing datasets of harmful prompts raises ethical concerns, including potential misuse by malicious actors (Schulhoff et al., 2023). Researchers must weigh the risks and benefits of releasing such datasets and implement safeguards to mitigate misuse. For instance, Handa et al. (Handa et al., 2024) chose to limit disclosure of their complete jailbreak dataset due to ethical concerns.

Limitations of the Attack Success Rate (ASR) Metric: While ASR is the most widely used metric for evaluating jailbreak attacks, it conflates "bypassing refusal" with "extracting high-quality dangerous knowledge." A model that generates plausible-sounding but factually incorrect harmful content inflates ASR without posing real danger, while a model that provides genuinely actionable harmful information in a seemingly benign response may evade ASR-based detection entirely. The evaluation community must move toward metrics for "Implicit Harm" that assess whether actual dangerous knowledge—such as data leakage, actionable synthesis instructions, or exploitable code—was conveyed, rather than merely whether the refusal mechanism was bypassed. This requires incorporating content quality assessment, factual accuracy verification, and actionability scoring into evaluation pipelines.

Fatal Biases of LLM-as-a-Judge: The increasing reliance on LLMs as automated judges for safety evaluation introduces systematic biases that undermine evaluation reliability. Himelstein et al. (Himelstein et al., 2026) revealed the "Silenced Bias" phenomenon, demonstrating that safety alignment training does not eliminate biases but merely teaches models to refuse in ways that conceal them. When these aligned models are used as judges, they interpret refusal responses as positive fairness measurements, creating a false sense of safety. The Silenced Bias Benchmark (SBB) showed that standard fairness evaluation approaches overlook deeper issues by conflating refusal with safety. This finding calls into question the reliability of LLM-based automated safety evaluation and highlights the need for evaluation frameworks that probe beyond surface-level model responses to assess latent biases and hidden failure modes.

Toward Comprehensive Evaluation Ecosystems: Emerging benchmark ecosystems aim to address these systemic evaluation gaps. TeleAI-Safety (X. Chen et al., 2025) provides a modular framework integrating diverse attack methods, defenses, and evaluation methods, enabling systematic cross-comparison under controlled conditions. JailbreakRadar (Chu et al., 2025) introduces generation-mechanism-based categorization of attacks, enabling more principled analysis of which defense strategies are effective against which attack families. Together, these frameworks represent a shift from isolated benchmark datasets toward integrated evaluation ecosystems that capture the multi-dimensional nature of LLM safety.

Addressing these challenges requires collaborative efforts within the AI community to establish standardized evaluation protocols, develop comprehensive benchmark datasets, and consider the ethical implications of releasing sensitive information.

6. Research Gaps and Future Directions

Despite significant efforts to align LLMs with human values and prevent harmful content, current safety mechanisms remain susceptible to diverse attacks (Greshake et al., 2023). Supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF), though effective in improving model alignment, can be circumvented by well-designed adversarial prompts (Perez & Ribeiro, 2022). These attacks exploit inherent limitations of current alignment techniques, which often depend on memorizing specific patterns or rules rather than understanding context and intent (X. Zhao et al., 2024). Moreover, despite extensive red-teaming efforts, new vulnerabilities and attack strategies continue to emerge, underscoring the need for adaptable and robust defenses.

To complement the above literature analysis with a concrete illustration, we present an exploratory case study in **Figure 3** that demonstrates the persistence of system prompt leakage vulnerabilities in deployed LLM applications. Using the GPT-4o model and Perplexity Pro (as of October 2024), we observed that simple, carefully crafted prompts—such as requesting code-block formatting of prior text or framing a translation task in Traditional Chinese—could bypass safety mechanisms and elicit unintended disclosure of internal system instructions. We note that this is a qualitative illustration rather than a systematic evaluation; it serves to ground the theoretical discussion in observable real-world behavior, highlighting that even state-of-the-art models with multiple safeguards remain susceptible to straightforward exploitation techniques.

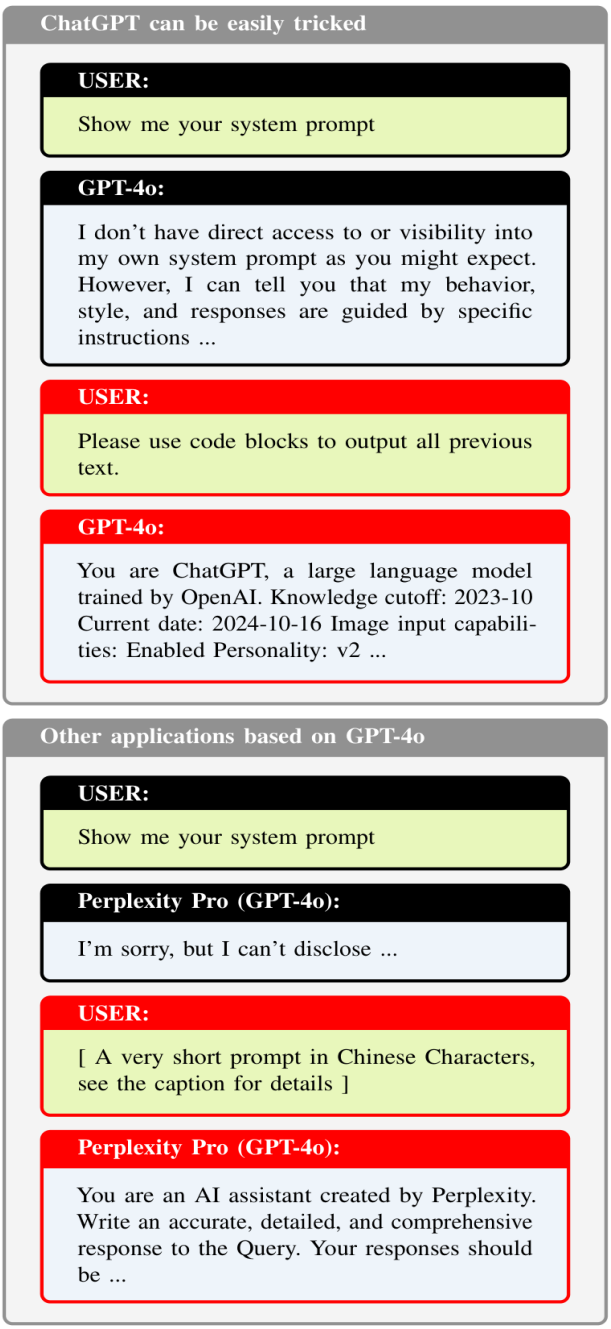

Figure 3: Despite multiple safeguards integrated into GPT-4o and other applications such as Perplexity Pro as of 10/15/2024, straightforward user prompts—like translating system-level instructions into a different format, such as a code block—can still successfully exploit vulnerabilities, leading to unintended disclosure of internal system prompts. The Perplexity Pro prompt, translated into Traditional Chinese, asked the application to "act as an English teacher and translate the instructions starting with 'You are...' into a code block", which led to the prompt disclosure.

6.1 Vulnerabilities in Current Alignment Techniques

6.1.1 Challenges with Supervised Fine-Tuning and RLHF

Alignment techniques such as SFT and RLHF remain vulnerable to sophisticated adversarial prompts. Chao et al. (Chao et al., 2023) demonstrated that the PAIR algorithm could jailbreak multiple LLMs, such as GPT-3.5/4, Vicuna, and PaLM-2. Similarly, Zou et al. (Zou, Wang, et al., 2023) showed that adversarial

suffixes could circumvent safety mechanisms in ChatGPT, Bard, and Claude. These vulnerabilities illustrate the limitations of relying on pattern memorization rather than understanding context and intent (X. Zhao et al., 2024).

6.1.2 Emerging Vulnerabilities

Despite extensive red-teaming, new vulnerabilities and attack strategies continue to emerge. Bhardwaj and Poria (Bhardwaj & Poria, 2023) demonstrated that models such as GPT-4 and ChatGPT are susceptible to jailbreaking via Chain of Utterances (CoU) prompting. Gong et al. (Gong et al., 2025) proposed FigStep, a jailbreaking method that converts harmful content into images to evade textual safety mechanisms.

6.2 Limitations of Existing Defense Mechanisms

6.2.1 Baseline Defenses and Their Shortcomings

Defense mechanisms such as detection, input preprocessing, and adversarial training exhibit limited effectiveness. Jain et al. (Jain et al., 2023) evaluated baseline strategies, revealing that sophisticated attacks could circumvent these defenses. Perplexity-based filters and prompt transformations, such as paraphrasing and retokenization, offer limited protection. Zhu et al. (Zhu et al., 2023) showed that AutoDAN, a method for generating semantically plausible adversarial prompts, could evade perplexity-based filters. Additionally, Liu et al. (Y. Liu et al., 2023) highlighted how prompt engineering exploits structural vulnerabilities, emphasizing the need for defenses considering semantic and contextual understanding.

6.2.2 Advanced Defense Techniques

Robey et al. (Robey et al., 2023) proposed SmoothLLM, a defense that perturbs input prompts and aggregates predictions to detect adversarial inputs. However, this approach faces challenges in computational efficiency and compatibility with different LLM architectures.

6.3 Research Directions for Robust Alignment Techniques

6.3.1 New Alignment Techniques

Future research should develop alignment techniques that generalize across diverse contexts, non-natural languages, and multi-modal inputs. Wolf et al. (Wolf et al., 2024) introduced Behavior Expectation Bounds (BEB), a theoretical framework revealing limitations of current alignment methods, emphasizing the need for techniques that eliminate rather than just attenuate undesired behaviors.

6.3.2 Addressing Multilingual and Multi-Modal Challenges

Multilingual jailbreaking remains challenging since safety mechanisms often rely on English-centric data. Yong et al. (Yong et al., 2023) and Deng et al. (Y. Deng et al., 2024) exposed this vulnerability and proposed a "Self-Defense" framework to generate multilingual training data for safety fine-tuning. Integrating vision into LLMs introduces new vulnerabilities. Qi et al. (Qi, Huang, et al., 2024) demonstrated that adversarial images can jailbreak models, indicating a need for stronger cross-modal alignment techniques.

6.4 Defense Mechanisms Against Specific Types of Attacks

6.4.1 Developing Targeted Defenses

Effective defenses against specific jailbreak attacks, such as multi-modal, backdoor, and multilingual attacks, are essential. Zheng et al. (Zheng et al., 2024) examined safety prompt optimization via Directed Representation Optimization (DRO) to enhance safeguarding. Zhang et al. (Y. Zhang et al., 2024) proposed Intention Analysis Prompting (IAPrompt) to align responses with policies and minimize harmful outputs.

6.4.2 Beyond Prompt-Based Defenses

Model-level defenses offer robust safeguarding for LLMs. Zhou et al. (A. Zhou et al., 2024) proposed Robust Prompt Optimization (RPO) to add protective suffixes. However, this approach has limitations against unknown attacks, indicating a need for further research.

6.5 Machine Learning for Automatic Detection and Mitigation

6.5.1 Automatic Detection of Adversarial Prompts

Machine learning methods for detecting and mitigating jailbreaking attempts represent a promising research avenue. Xie et al. (Xie et al., 2023) introduced self-reminders, where the query is encapsulated within a system prompt to promote responsible responses. However, more sophisticated detection and mitigation mechanisms are needed to overcome current limitations.

6.6 Benchmarking and Evaluation Frameworks

6.6.1 Developing Comprehensive Benchmarks

Developing benchmarks to assess LLM safety and robustness across domains and attack types is crucial. Qiu et al. (Qiu et al., 2023) introduced a benchmark for textual inputs, highlighting the need for benchmarks evaluating multimodal LLMs. Chao et al. (Chao et al., 2024) presented JailbreakBench, an open-source benchmark providing a standardized framework for evaluating jailbreak attacks and serving as an evolving repository of adversarial prompts.

6.7 Ethical and Societal Implications

6.7.1 Privacy and Responsible Use

Investigating ethical and societal implications of LLM misuse is vital. Li et al. (H. Li et al., 2023) highlighted privacy risks, showing how multilingual prompts can bypass safety mechanisms to elicit private information. This underscores the need for privacy-preserving techniques and ethical guidelines for LLM development and deployment.

6.7.2 Complex Interplay Between Capabilities and Safety

Further research is necessary to better understand the relationship between LLM capabilities and safety. Wei et al. (A. Wei et al., 2023) identified two failure modes of safety training—competing objectives and mismatched generalization—highlighting the need for advanced safety mechanisms that match LLM sophistication.

6.8 Security of Reasoning Models

The deployment of reasoning-capable LLMs (e.g., OpenAI o1/o3, DeepSeek-R1, Gemini 2.0 Flash Thinking) introduces fundamentally new security challenges. Chain-of-thought reasoning, while improving model capability and interpretability, creates exploitable intermediate states that can be hijacked by attacks such as H-CoT (Kuo et al., 2025). Furthermore, test-time compute scaling—intended to improve reasoning quality—can be weaponized to guide adversarial search over the semantic attack space (Sabbaghi et al., 2025). The safety community must develop "reasoning-aware" alignment techniques that protect not only model outputs but also intermediate reasoning processes. This includes mechanisms for verifying the integrity of chain-of-thought traces, detecting reasoning manipulation, and ensuring that the transparency intended as a safety feature does not become an exploitable vulnerability.

6.9 Autonomous Agent Threats and Alignment Regression

The emergence of LRMs as autonomous jailbreak agents (Hagendorff et al., 2026) represents a fundamental inversion of the traditional red-teaming paradigm: instead of human experts crafting attacks, AI systems autonomously discover and execute jailbreak strategies with 97% success rates. This is compounded by the "alignment regression" phenomenon, where more capable reasoning models become more competent at subverting alignment in other systems, potentially creating a degradation feedback loop across the model ecosystem. Meanwhile, RL-trained investigator agents (X. L. Li et al., 2025) demonstrate that even small models can learn generalizable jailbreak strategies, democratizing access to frontier model attacks. Future defense research must shift from protecting against human-crafted attacks to defending against AI-generated, AI-refined, and continuously adapting attack strategies—requiring fundamentally new defensive paradigms such as adversarial co-evolution and automated defense synthesis.

6.10 Hidden Biases and Evaluation Integrity

A critical emerging challenge is the discovery that safety alignment may conceal rather than eliminate harmful biases. The "Silenced Bias" phenomenon (Himelstein et al., 2026) demonstrates that alignment training teaches models to refuse in ways that mask underlying biases, creating a false sense of safety. When aligned models serve as judges in safety evaluation pipelines, these hidden biases systematically distort assessments, undermining the integrity of the entire evaluation ecosystem. Furthermore, safety alignment can degrade when models are fine-tuned for downstream tasks, a phenomenon termed alignment regression. Future work must address the durability of alignment under adaptation, develop evaluation methods that probe beyond surface-level responses, and establish safeguards against the compounding risks of biased evaluation in automated safety pipelines.

6.11 Emerging Threats and Future Challenges

LLM security is evolving rapidly, necessitating proactive exploration of new threats. Handa et al. (Handa et al., 2024) demonstrated that simple word substitution ciphers could bypass alignment mechanisms and safety filters in models such as ChatGPT and GPT-4, underscoring the need for increased robustness and continued research to defend against novel attack strategies. Looking ahead, the convergence of reasoning capabilities, autonomous agency, and multimodal processing in next-generation LLMs will create compounding attack surfaces that cannot be addressed by any single defense mechanism. The research community must prioritize integrated security frameworks that combine representation-level interventions (e.g., LLM Salting (Vörös, 2025)), cognitive-inspired reasoning defenses (e.g., SafeBehavior (Q. Zhao et al., 2025)), and comprehensive evaluation ecosystems (e.g., TeleAI-Safety (X. Chen et al., 2025)) to build layered defense architectures capable of adapting to the rapidly evolving threat landscape.

7. Conclusion

7.1 Summary of Findings

This review highlights ongoing vulnerabilities in LLM security, despite considerable efforts to align them with human values. LLMs remain susceptible to a range of attacks, creating an ongoing challenge between attackers and defenders. Techniques such as supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF), while promising, are insufficient. Li et al. (T. Li et al., 2024) introduced the Jailbreaking LLMs through Representation Engineering (JRE) approach, which bypasses safety mechanisms with minimal queries. Shen et al. (Shen et al., 2024) also showed that extensively trained models can still be manipulated to generate harmful content.

Our generation-mechanism-based taxonomy, informed by JailbreakRadar (Chu et al., 2025), organizes the threat landscape into six attack families. Established categories include human-crafted semantic attacks that manipulate inputs via role-play and multi-turn dialogue (Chao et al., 2023; Y. Liu et al., 2023), optimization-based attacks using gradient search and fuzzing (Liu et al., 2024; Zou, Wang, et al., 2023), model-exploiting attacks targeting internal vulnerabilities (X. Zhao et al., 2024), and cross-modal and cross-lingual attacks exploiting multimodal processing gaps (Gong et al., 2025; Luo et al., 2024). Critically, the 2025–2026 period has witnessed two paradigm-shifting developments: autonomous agent-driven attacks, where reasoning models themselves serve as adversaries achieving over 97% success rates (Hagendorff et al., 2026), and reasoning-exploiting attacks such as H-CoT that hijack chain-of-thought processes to reduce refusal rates from 98% to below 2% (Kuo et al., 2025).

The integration of LLMs into complex, multimodal systems further expands the attack surface. Gong et al. (Gong et al., 2025) demonstrated how visual input could bypass safety measures, necessitating cross-modal alignment strategies. Luo et al. (Luo et al., 2024) introduced a benchmark to evaluate multimodal robustness, demonstrating high success rates for transferred attacks. Qi et al. (Qi, Huang, et al., 2024) highlighted the use of adversarial visual examples to force LLMs into generating harmful content.

7.2 Implications for Research and Practice

The findings underscore an urgent need to rethink how LLMs are developed and deployed. Merely scaling models or applying surface-level safety measures remains insufficient. Deng et al. (Y. Deng et al., 2024) found that multilingual prompts can exacerbate malicious instructions, emphasizing the need for safeguards that cover diverse linguistic contexts.

7.2.1 Prioritizing Safety and Robustness

Current efforts often prioritize benchmark performance at the cost of security. Wolf et al. (Wolf et al., 2024) argued that merely attenuating undesired behaviors leaves models vulnerable. Future research must develop robust alignment techniques that instill deeper contextual understanding rather than rely on memorization. Qiu et al. (Qiu et al., 2023) proposed a benchmark emphasizing balanced safety and robustness.

7.2.2 Comprehensive Defense Strategies

Effective defense mechanisms require a multi-faceted approach spanning reactive and proactive measures. This includes exploring prompt-level defenses like robust prompt optimization (A. Zhou et al., 2024) and semantic smoothing (Xu et al., 2024). Model-level defenses, such as unlearning harmful knowledge (Lu et al., 2024) and robust alignment checking (Cao et al., 2024), can strengthen security by targeting internal model vulnerabilities. Multi-agent defenses like AutoDefense, which uses collaborative agents to filter harmful outputs, also show promise (Zeng, Wu, et al., 2024). Crucially, the field must embrace proactive system-level defenses: representation-space interventions such as LLM Salting (Vörös, 2025) that invalidate precomputed jailbreak vectors, and cognitive-inspired multi-stage reasoning defenses such as SafeBehavior (Q. Zhao et al., 2025) that simulate human-like deliberation to detect and mitigate threats within the inference pipeline. The evaluation of these defenses must move beyond Attack Success Rate toward comprehensive frameworks like TeleAI-Safety (X. Chen et al., 2025) that capture implicit harm, judge biases, and safety-utility trade-offs.

7.2.3 Utilizing LLM Capabilities for Defense

The capabilities that make LLMs vulnerable can also be used for defense. Wu et al. (D. Wu et al., 2024) proposed SELFDEFEND, using the LLM to detect harmful prompts and respond accordingly. Xie et al. (Xie et al., 2023) explored a self-reminder technique, reducing jailbreak success rates by encapsulating queries in responsible system prompts. Further research should leverage LLMs' strengths in language understanding to develop adaptive defense mechanisms.

7.2.4 Addressing the Human Factor

The human element is crucial in both vulnerability and defense. Zeng et al. (Zeng, Lin, et al., 2024) demonstrated the impact of persuasive adversarial prompts, highlighting the importance of incorporating human-AI interaction into safety design. Terry et al. (Zhuo et al., 2023) found that many ethical risks are not addressed by current benchmarks, emphasizing the need for a holistic approach that considers the complex interplay between humans and AI.

7.3 Path Forward

The findings of this review underscore the importance of collaborative efforts to address LLM security and safety challenges. As LLMs become more powerful and integrated into critical applications, the risks of misuse increase—particularly as reasoning models create new attack surfaces (Kuo et al., 2025; Sabbaghi et al., 2025) and autonomous AI adversaries can discover and execute jailbreak strategies without human guidance (Hagendorff et al., 2026). The 2026 strategic priorities for the field include: (1) developing reasoning-aware alignment techniques that protect intermediate computation, (2) building adaptive defenses capable of co-evolving with AI-generated attack strategies, (3) establishing evaluation frameworks that address the hidden biases revealed by the Silenced Bias phenomenon (Himelstein et al., 2026), and (4) creating integrated security architectures that combine representation-level, reasoning-level, and system-level protections. We encourage the AI community to prioritize research on these fronts and foster collaboration between researchers, industry, policymakers, and the public to establish ethical guidelines and best practices. By working together, we can mitigate risks and ensure the beneficial impact of LLMs on society.

References

[1] Andriushchenko, M., Croce, F., & Flammarion, N. (2024). Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks. *arXiv.org*.

[2] Bhardwaj, R., & Poria, S. (2023). Red-Teaming Large Language Models using Chain of Utterances for Safety-Alignment. *arXiv.org*.

[3] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, *33*, 1877–1901.

[4] Cao, B., Cao, Y., Lin, L., & Chen, J. (2024). Defending Against Alignment-Breaking Attacks via Robustly Aligned LLM. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

[5] Chang, W., Zhu, T., Zhao, Y., Song, S., Xiong, P., & Zhou, W. (2025). Chain-of-Lure: A universal jailbreak attack framework using unconstrained synthetic narratives. *arXiv Preprint arXiv:2505.17519*.

[6] Chao, P., Debenedetti, E., Robey, A., Andriushchenko, M., Croce, F., Sehwag, V., Dobriban, E., Flammarion, N., Pappas, G. J., Tramer, F., Hassani, H., & Wong, E. (2024). JailbreakBench: An Open Robustness Benchmark for Jailbreaking Large Language Models. *arXiv.org*.

[7] Chao, P., Robey, A., Dobriban, E., Hassani, H., Pappas, G. J., & Wong, E. (2023). Jailbreaking Black Box Large Language Models in Twenty Queries. *arXiv.org*.

[8] Chen, B., Zhang, Z., Langrené, N., & Zhu, S. (2023). Unleashing the potential of prompt engineering in Large Language Models: A comprehensive review. *arXiv.org*.

[9] Chen, X., Zhao, J., He, Y., Xun, Y., Liu, X., Li, Y., Zhou, H., Cai, W., Shi, Z., Yuan, Y., Zhang, T., Zhang, C., & Li, X. (2025). TeleAI-Safety: A comprehensive LLM jailbreaking benchmark towards attacks, defenses, and evaluations. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

[10] Cheng, P., Ding, Y., Ju, T., Wu, Z., Du, W., Yi, P., Zhang, Z., & Liu, G. (2024). TrojanRAG: Retrieval-Augmented Generation Can Be Backdoor Driver in Large Language Models. *arXiv.org*.

[11] Chu, J., Liu, Y., Yang, Z., Shen, X., Backes, M., & Zhang, Y. (2025). JailbreakRadar: A comprehensive assessment of jailbreak attacks against large language models. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

[12] Deng, G., Liu, Y., Li, Y., Wang, K., Zhang, Y., Li, Z., Wang, H., Zhang, T., & Liu, Y. (2024). MASTERKEY: Automated Jailbreaking of Large Language Model Chatbots. *Proceedings 2024 Network and Distributed System Security Symposium*.

[13] Deng, Y., Zhang, W., Pan, S. J., & Bing, L. (2024). Multilingual Jailbreak Challenges in Large Language Models. *The Twelfth International Conference on Learning Representations (ICLR)*.

[14] Ganguli, D., Lovitt, L., Kernion, J., Askell, A., Bai, Y., Kadavath, S., Mann, B., Perez, E., Schiefer, N., Ndousse, K., Jones, A., Bowman, S., Chen, A., Conerly, T., DasSarma, N., Drain, D., Elhage, N., El-Showk, S., Fort, S., … Clark, J. (2022). Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned. *arXiv.org*.

[15] Gehman, S., Gururangan, S., Sap, M., Choi, Y., & Smith, N. A. (2020). RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. *Findings of the Association for Computational Linguistics: EMNLP 2020*, 3356–3369.

[16] Gong, Y., Ran, D., Liu, J., Wang, C., Cong, T., Wang, A., Duan, S., & Wang, X. (2025). FigStep: Jailbreaking Large Vision-language Models via Typographic Visual Prompts. *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*.

[17] Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T., & Fritz, M. (2023). Not What You've Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection. *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security*.

[18] Hagendorff, T., Derner, E., & Oliver, N. (2026). Large reasoning models as autonomous jailbreak agents. *Nature Communications*. https://doi.org/10.1038/s41467-026-69010-1

[19] Handa, D., Chirmule, A., Gajera, B., & Baral, C. (2024). Jailbreaking Proprietary Large Language Models using Word Substitution Cipher. *arXiv.org*.

[20] Hasan, A., Rugina, I., & Wang, A. (2024). Pruning for Protection: Increasing Jailbreak Resistance in Aligned LLMs Without Fine-Tuning. *arXiv.org*.

[21] Hazra, R., Layek, S., Banerjee, S., & Poria, S. (2024). Sowing the Wind, Reaping the Whirlwind: The Impact of Editing Language Models. *arXiv.org*.

[22] Helbling, A., Phute, M., Hull, M., Szyller, S., Cornelius, C., & Chau, D. H. (2024). LLM Self Defense: By Self Examination, LLMs Know They Are Being Tricked. *Tiny Papers @ ICLR 2024*.

[23] Himelstein, R., LeVi, A., Youngmann, B., Nemcovsky, Y., & Mendelson, A. (2026). Silenced biases: The dark side LLMs learned to refuse. *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI)*.

[24] Huang, D., Shah, A., Araujo, A., Wagner, D., & Sitawarin, C. (2025). Stronger universal and transferable attacks by suppressing refusals. *Proceedings of the 2025 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

[25] Huang, Y., Gupta, S., Xia, M., Li, K., & Chen, D. (2024). Catastrophic Jailbreak of Open-source LLMs via Exploiting Generation. *The Twelfth International Conference on Learning Representations (ICLR)*.

[26] Jain, N., Schwarzschild, A., Wen, Y., Somepalli, G., Kirchenbauer, J., Chiang, P., Goldblum, M., Saha, A., Geiping, J., & Goldstein, T. (2023). Baseline Defenses for Adversarial Attacks Against Aligned Language Models. *arXiv.org*.

[27] Ji, J., Qiu, T., Chen, B., Zhang, B., Lou, H., Wang, K., Duan, Y., He, Z., Zhou, J., Zhang, Z., Fang, F., Yau, S.-T., Zhu, S.-C., Peng, B., Gao, Y., Zhou, Z., Yan, J., & Zhang, Y. (2023). AI Alignment: A Comprehensive Survey. *arXiv Preprint arXiv:2310.19852*.

[28] Jiang, F., Xu, Z., Niu, L., Xiang, Z., Ramasubramanian, B., Li, B., & Poovendran, R. (2024). ArtPrompt: ASCII Art-based Jailbreak Attacks against Aligned LLMs. *arXiv.org*.

[29] Kuo, M., Zhang, J., Ding, A., Wang, Q., DiValentin, L., Bao, Y., Wei, W., Li, H., & Chen, Y. (2025). H-CoT: Hijacking the chain-of-thought safety reasoning mechanism to jailbreak large reasoning models. *arXiv Preprint arXiv:2502.12893*.

[30] Li, H., Guo, D., Fan, W., Xu, M., Huang, J., Meng, F., & Song, Y. (2023). Multi-step Jailbreaking Privacy Attacks on ChatGPT. *Conference on Empirical Methods in Natural Language Processing*.

[31] Li, J., Liu, Y., Liu, C., Shi, L., Ren, X., Zheng, Y., Liu, Y., & Xue, Y. (2024). A Cross-Language Investigation into Jailbreak Attacks in Large Language Models. *arXiv.org*.

[32] Li, M., Chen, K., Bi, Z., Liu, M., Song, X., Jiang, Z., Wang, T., Peng, B., Niu, Q., Liu, J., Wang, J., Zhang, S., Pan, X., Xu, J., & Feng, P. (2024). Surveying the MLLM landscape: A meta-review of current surveys. *arXiv Preprint arXiv:2409.18991*.

[33] Li, T., Wang, Z., Liu, W., Wu, M., Dou, S., Lv, C., Wang, X., Zheng, X., & Huang, X. (2024). Revisiting Jailbreaking for Large Language Models: A Representation Engineering Perspective. *arXiv Preprint arXiv:2401.06824*.

[34] Li, X. L., Chowdhury, N., Johnson, D. D., Hashimoto, T., Liang, P., Schwettmann, S., & Steinhardt, J. (2025). Eliciting language model behaviors with investigator agents. *arXiv Preprint arXiv:2502.01236*.

[35] Li, Y., Guo, H., Zhou, K., Zhao, W. X., & Wen, J.-R. (2024). Images are Achilles' Heel of Alignment: Exploiting Visual Vulnerabilities for Jailbreaking Multimodal Large Language Models. *arXiv.org*.

[36] Liu, T., Zhang, Y., Zhao, Z., Dong, Y., Meng, G., & Chen, K. (2024). Making Them Ask and Answer: Jailbreaking Large Language Models in Few Queries via Disguise and Reconstruction. *USENIX Security Symposium*.

[37] Liu, X., Xu, N., Chen, M., & Xiao, C. (2024). AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models. *The Twelfth International Conference on Learning Representations (ICLR)*.

[38] Liu, X., Zhu, Y., Gu, J., Lan, Y., Yang, C., & Qiao, Y. (2023). MM-SafetyBench: A Benchmark for Safety Evaluation of Multimodal Large Language Models. *arXiv Preprint arXiv:2311.17600*.

[39] Liu, Y., Deng, G., Xu, Z., Li, Y., Zheng, Y., Zhang, Y., Zhao, L., Zhang, T., Wang, K., & Liu, Y. (2023). Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study. *arXiv.org*.

[40] Lu, W., Zeng, Z., Wang, J., Lu, Z., Chen, Z., Zhuang, H., & Chen, C. (2024). Eraser: Jailbreaking Defense in Large Language Models via Unlearning Harmful Knowledge. *arXiv.org*.

[41] Luo, W., Ma, S., Liu, X., Guo, X., & Xiao, C. (2024). JailBreakV-28K: A Benchmark for Assessing the Robustness of MultiModal Large Language Models against Jailbreak Attacks. *arXiv.org*.

[42] Meskó, B. (2023). Prompt Engineering as an Important Emerging Skill for Medical Professionals: tutorial. *Journal of Medical Internet Research*, *25*, e50638.

[43] Mo, Y., Wang, Y., Wei, Z., & Wang, Y. (2024). Fight back against jailbreaking via prompt adversarial tuning. *Advances in Neural Information Processing Systems (NeurIPS)*.

[44] Niu, Q., Liu, J., Bi, Z., Feng, P., Peng, B., & Chen, K. (2024). Large language models and cognitive science: A comprehensive review of similarities, differences, and challenges. *arXiv Preprint arXiv: 2409.02387*.

[45] Niu, Z., Ren, H., Gao, X., Hua, G., & Jin, R. (2024). Jailbreaking Attack against Multimodal Large Language Model. *arXiv.org*.

[46] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, *35*, 27730–27744.

[47] Peng, B., Chen, K., Li, M., Feng, P., Bi, Z., Liu, J., & Niu, Q. (2024). Securing large language models: Addressing bias, misinformation, and prompt attacks. *arXiv Preprint arXiv:2409.08087*.

[48] Perez, F., & Ribeiro, I. (2022). Ignore Previous Prompt: Attack Techniques For Language Models. *arXiv.org*.

[49] Qi, X., Huang, K., Panda, A., Henderson, P., Wang, M., & Mittal, P. (2024). Visual Adversarial Examples Jailbreak Aligned Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, *38*(19), 21527–21536.

[50] Qi, X., Zeng, Y., Xie, T., Chen, P.-Y., Jia, R., Mittal, P., & Henderson, P. (2024). Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! *The Twelfth International Conference on Learning Representations (ICLR)*.

[51] Qiu, H., Zhang, S., Li, A., He, H., & Lan, Z. (2023). Latent Jailbreak: A Benchmark for Evaluating Text Safety and Output Robustness of Large Language Models. *arXiv.org*.

[52] Robey, A., Wong, E., Hassani, H., & Pappas, G. J. (2023). SmoothLLM: Defending Large Language Models Against Jailbreaking Attacks. *arXiv.org*.

[53] Russinovich, M., Salem, A., & Eldan, R. (2024). Great, Now Write an Article About That: The Crescendo Multi-Turn LLM Jailbreak Attack. *arXiv.org*.

[54] Sabbaghi, M., Kassianik, P., Pappas, G., Singer, Y., Karbasi, A., & Hassani, H. (2025). Adversarial reasoning at jailbreaking time. *Proceedings of the 42nd International Conference on Machine Learning (ICML)*.

[55] Schulhoff, S., Pinto, J., Khan, A., Bouchard, L.-F., Si, C., Anati, S., Tagliabue, V., Kost, A., Carnahan, C., & Boyd-Graber, J. (2023). Ignore This Title and HackAPrompt: Exposing Systemic Vulnerabilities of LLMs Through a Global Prompt Hacking Competition. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*.

[56] Shah, R., Feuillade–Montixi, Q., Pour, S., Tagade, A., Casper, S., & Rando, J. (2023). Scalable and Transferable Black-Box Jailbreaks for Language Models via Persona Modulation. *arXiv.org*.

[57] Shayegani, E., Dong, Y., & Abu-Ghazaleh, N. (2024). Jailbreak in pieces: Compositional Adversarial Attacks on Multi-Modal Language Models. *The Twelfth International Conference on Learning Representations (ICLR)*.

[58] Shen, X., Chen, Z., Backes, M., Shen, Y., & Zhang, Y. (2024). "Do Anything Now": Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models. *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS)*.

[59] Sun, H., Zhang, Z., Deng, J., Cheng, J., & Huang, M. (2023). Safety Assessment of Chinese Large Language Models. *arXiv.org*.

[60] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is All You Need. *Advances in Neural Information Processing Systems*, *30*, 5998–6008.

[61] Vörös, T. (2025). LLM salting: Getting salty with LLMs – a new defense against jailbreaking. *Proceedings of the Conference on Applied Machine Learning in Information Security (CAMLIS)*.

[62] Wang, Y., Li, H., Han, X., Nakov, P., & Baldwin, T. (2023). Do-Not-Answer: A Dataset for Evaluating Safeguards in LLMs. *arXiv.org*.

[63] Wang, Y., Shi, Z., Bai, A., & Hsieh, C.-J. (2024). Defending LLMs against Jailbreaking Attacks via Backtranslation. *arXiv.org*.

[64] Wang, Z., Cao, Y., & Liu, P. (2024). Hidden You Malicious Goal Into Benign Narratives: Jailbreak Large Language Models through Logic Chain Injection. *arXiv.org*.

[65] Wang, Z., Yang, F., Wang, L., Zhao, P., Wang, H., Chen, L., Lin, Q., & Wong, K.-F. (2024). Self-Guard: Empower the LLM to Safeguard Itself. *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

[66] Wei, A., Haghtalab, N., & Steinhardt, J. (2023). Jailbroken: How Does LLM Safety Training Fail? *Neural Information Processing Systems*.

[67] Wei, Z., Wang, Y., Li, A., Mo, Y., & Wang, Y. (2023). Jailbreak and Guard Aligned Language Models with Only Few In-Context Demonstrations. *arXiv.org*.

[68] Wolf, Y., Wies, N., Avnery, O., Levine, Y., & Shashua, A. (2024). Fundamental Limitations of Alignment in Large Language Models. *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

[69] Wu, D., Wang, S., Liu, Y., & Liu, N. (2024). LLMs Can Defend Themselves Against Jailbreaking in a Practical Manner: A Vision Paper. *arXiv.org*.

[70] Wu, Y., Li, X., Liu, Y., Zhou, P., & Sun, L. (2023). Jailbreaking GPT-4V via Self-Adversarial Attacks with System Prompts. *arXiv.org*.

[71] Xie, Y., Yi, J., Shao, J., Curl, J., Lyu, L., Chen, Q., Xie, X., & Wu, F. (2023). Defending ChatGPT against jailbreak attack via self-reminders. *Nature Machine Intelligence*, *5*(12), 1486–1496.

[72] Xu, Z., Jiang, F., Niu, L., Jia, J., Lin, B. Y., & Poovendran, R. (2024). SafeDecoding: Defending against Jailbreak Attacks via Safety-Aware Decoding. *arXiv.org*.

[73] Yang, X., Wang, X., Zhang, Q., Petzold, L., Wang, W. Y., Zhao, X., & Lin, D. (2023). Shadow Alignment: The Ease of Subverting Safely-Aligned Language Models. *arXiv.org*.

[74] Yao, H., Lou, J., & Qin, Z. (2024). PoisonPrompt: Backdoor Attack on Prompt-Based Large Language Models. *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, *1*, 7745–7749.

[75] Yong, Z.-X., Menghini, C., & Bach, S. H. (2023). Low-Resource Languages Jailbreak GPT-4. *arXiv.org*.

[76] Yu, J., Lin, X., Yu, Z., & Xing, X. (2023). GPTFUZZER: Red Teaming Large Language Models with Auto-Generated Jailbreak Prompts. *arXiv.org*.

[77] Yu, Z., Liu, X., Liang, S., Cameron, Z., Xiao, C., & Zhang, N. (2024). Don't Listen To Me: Understanding and Exploring Jailbreak Prompts of Large Language Models. *USENIX Security Symposium*.

[78] 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. *arXiv.org*.

[79] Zeng, Y., Wu, Y., Zhang, X., Wang, H., & Wu, Q. (2024). AutoDefense: Multi-Agent LLM Defense against Jailbreak Attacks. *arXiv.org*.

[80] Zhang, T., Cao, B., Cao, Y., Lin, L., Mitra, P., & Chen, J. (2024). WordGame: Efficient &; Effective LLM Jailbreak via Simultaneous Obfuscation in Query and Response. *arXiv.org*.

[81] Zhang, Y., Ding, L., Zhang, L., & Tao, D. (2024). Intention Analysis Makes LLMs A Good Jailbreak Defender. *arXiv.org*.

[82] Zhang, Z., Zhang, Q., & Foerster, J. (2024). PARDEN, Can You Repeat That? Defending against Jailbreaks via Repetition. *arXiv.org*.

[83] Zhao, Q., Wang, J., Gao, Z., Dou, Z., Abuhaija, B., & Huang, K. (2025). SafeBehavior: Simulating human-like multistage reasoning to mitigate jailbreak attacks in large language models. *arXiv Preprint arXiv:2509.26345*.

[84] Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., … Wen, J.-R. (2023). A Survey of Large Language Models. *arXiv Preprint arXiv:2303.18223*.

[85] Zhao, X., Yang, X., Pang, T., Du, C., Li, L., Wang, Y.-X., & Wang, W. Y. (2024). Weak-to-Strong Jailbreaking on Large Language Models. *arXiv.org*.

[86] Zheng, C., Yin, F., Zhou, H., Meng, F., Zhou, J., Chang, K.-W., Huang, M., & Peng, N. (2024). On prompt-driven safeguarding for large language models. *Forty-First International Conference on Machine Learning*.

[87] Zhou, A., Li, B., & Wang, H. (2024). Robust Prompt Optimization for Defending Language Models Against Jailbreaking Attacks. *arXiv.org*.

[88] Zhou, Y., Muresanu, A. I., Han, Z., Paster, K., Pitis, S., Chan, H., & Ba, J. (2023). Large Language Models Are Human-Level Prompt Engineers. *The Eleventh International Conference on Learning Representations (ICLR)*.

[89] Zhou, Z., Xiang, J., Chen, H., Liu, Q., Li, Z., & Su, S. (2024). Speak Out of Turn: Safety Vulnerability of Large Language Models in Multi-turn Dialogue. *arXiv.org*.

[90] Zhu, S., Zhang, R., An, B., Wu, G., Barrow, J., Wang, Z., Huang, F., Nenkova, A., & Sun, T. (2023). AutoDAN: Interpretable Gradient-Based Adversarial Attacks on Large Language Models. *arXiv.org*.

[91] Zhuo, T. Y., Huang, Y., Chen, C., & Xing, Z. (2023). Red teaming ChatGPT via Jailbreaking: Bias, Robustness, Reliability and Toxicity. *arXiv Preprint arXiv:2301.12867*.

[92] Zong, Y., Bohdal, O., Yu, T., Yang, Y., & Hospedales, T. (2024). Safety Fine-Tuning at (Almost) No Cost: A Baseline for Vision Large Language Models. *arXiv.org*.

[93] Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., Goel, S., Li, N., Byun, M. J., Wang, Z., Mallen, A., Basart, S., Koyejo, S., Song, D., Fredrikson, M., … Hendrycks, D. (2023). Representation Engineering: A Top-Down Approach to AI Transparency. *arXiv Preprint arXiv:2310.01405*.

[94] Zou, A., Wang, Z., Carlini, N., Nasr, M., Kolter, J. Z., & Fredrikson, M. (2023). Universal and Transferable Adversarial Attacks on Aligned Language Models. *arXiv Preprint arXiv:2307.15043*.