# Lifecycle Analysis of Security, Privacy, and Governance in Large Language Models

**Marko Pribisalić [1,*] and Sanda Martinčić-Ipšić [1]**

[1] Faculty of Informatics and Digital Technologies, University of Rijeka, Radmile Matejčić 2, 51000 Rijeka, Croatia;
* Correspondence: marko.pribisalic@student.uniri.hr

**Abstract**

Large Language Models (LLMs) are increasingly deployed across professional and societal domains, introducing security, privacy, and governance challenges beyond traditional software vulnerabilities. Despite extensive research on individual risk categories, a unified lifecycle-oriented perspective connecting architectural properties, adversarial threats, and governance implications remains limited. This survey examines security and privacy risks associated with LLMs through a lifecycle framework covering data acquisition, model training, alignment procedures, deployment, and post-deployment interaction. The study synthesizes prior research to construct a taxonomy of threats including prompt injection, jailbreaking, adversarial manipulation, training-stage attacks, privacy leakage, and socio-technical misuse. Ethical issues such as hallucination, bias amplification, and malicious use are analyzed alongside governance and regulatory frameworks. Results indicate that vulnerabilities in LLM systems arise primarily from probabilistic generation mechanisms, large-scale data ingestion, and complex deployment ecosystems rather than isolated implementation defects. Classical software vulnerability models therefore provide only partial coverage of risks associated with generative AI systems. The survey introduces the concept of the alignment gap to explain how discrepancies between training objectives and real-world interaction contribute to persistent vulnerabilities. The findings highlight the need for lifecycle-oriented defense-in-depth strategies combining technical safeguards, privacy-preserving training, runtime monitoring, and governance mechanisms to support responsible deployment of LLM-based systems.

**Keywords:** large language models; AI security; prompt injection; adversarial attacks; privacy leakage; alignment gap; LLM governance; AI safety

## 1. Introduction

The growing integration of LLMs into professional and social interaction has intensified the need for security evaluation [1,2]. Although foundation models (i.e., large-scale pre-trained generative models) demonstrate substantial capability in natural language processing and automation, their deployment has progressed faster than the development of robust safety and governance frameworks, raising concerns regarding reliability, accountability, and trust [3–6]. The adoption of LLM-based generative systems in domains such as medicine, law, and cybersecurity further amplifies these concerns, as inaccurate or unintended outputs may introduce legal liability and reduce public confidence [7,8]. Consequently, understanding risks emerging from uncurated training data and from architectural characteristics of Transformer-based models—particularly next-token prediction objectives, large-scale parameterization, and opaque latent representations—has become an important research objective. During training, model parameters are optimized using likelihood-based loss functions such as cross-entropy, while probabilistic generation during inference relies on next-token prediction. The underlying probabilistic mechanism is discussed in greater detail in Section 2 [9–11].

Traditional safety assessments are often insufficient against adversarial manipulation [12]. Even aligned LLMs remain susceptible to transferable attacks in which small variations in phrasing can steer outputs toward unintended behavior [13,14]. For example, prompt injection techniques have demonstrated that carefully crafted inputs can override system-level safety instructions and induce unintended tool execution [13,14]. In parallel, the availability of automated tools and malicious use of generative systems reshapes the digital threat landscape by lowering technical barriers for cyber exploitation [3,10]. Recent threat intelligence reports confirm a growing

integration of AI-assisted attack automation within modern cybercrime ecosystems [15]. Empirical observations indicate that LLM capabilities may be leveraged to support phishing and social engineering activities, highlighting the need for defensive mechanisms tailored to generative systems [1,4].

Privacy represents a security concern, particularly regarding memorization and potential exposure of personally identifiable information (PII) [4,16]. Studies demonstrate that LLMs may reproduce portions of their training data, introducing risks to data sovereignty and intellectual property protection [16,17]. The memorization tendency increases with model capacity and dataset scale, while limited transparency of training data provenance complicates ownership attribution and regulatory compliance [1,4,16]. Existing research highlights unresolved questions regarding data leakage mechanisms and the limitations of current privacy-preserving techniques. Recent surveys of open and rapidly evolving models emphasize the need for standardized safety evaluation frameworks across LLM deployments [2].

Ethical challenges further arise from the reproduction of societal biases present in large-scale corpora [18,19]. Such biases may influence decision-support applications including hiring or educational assessment [19]. Additionally, hallucinations — plausible but factually unsupported outputs — remain a major obstacle to reliable deployment [7]. Because LLMs cannot reliably distinguish adversarial intent expressed through authoritative language, interdisciplinary auditing and accountability standards are required [1,2].

The motivation for this review lies in the pace of model development exceeding regulatory and governance capabilities [1]. While cybersecurity, privacy, and AI ethics are widely studied individually, a unified safety-by-design perspective across the model lifecycle remains underaddressed [1,2]. In this work, the lifecycle perspective refers to a stage-oriented analysis encompassing data acquisition, model training, alignment procedures, deployment, and post-deployment interaction. Comprehensive reviews of open LLMs provide insight into rapid model evolution and reinforce the need for standardized security evaluation across deployments [2]. This paper presents a structured taxonomy of threats and a consolidated overview of mitigation strategies to support safer and more trustworthy applications [2,13,16]. The analysis links architectural characteristics with regulatory and governance considerations, connecting technical vulnerabilities to broader societal consequences [1,7].

The historical evolution of major application security risks that contextualize modern system-level vulnerabilities is illustrated in Table 1.

**Table 1.** Evolution of the Open Web Application Security Project (OWASP) Top-10 risk categories (2003–2025) [20–28]. The table illustrates how dominant application security threats have shifted over time from implementation-level vulnerabilities toward system-level and AI-related risks.

| OWASP Top Ten | 2003 | 2004 | 2007 | 2010 | 2013 | 2017 | 2021 | 2025 |
|---|---|---|---|---|---|---|---|---|
| Broken Access Control | A2 | A2[1] | A10[13] | A8 | A7[16] | A5 | A1 | A1 |
| Security Misconfiguration | A10 | A10[3][5] | x | A6 | A5 | A6 | A5 | A2 |
| Software Supply Chain Failures | x | x | x | x | x | x | x | A3[27] |
| Cryptographic Failures | A8 | A8[5][6] | A8 | A7 | A6[17] | A3 | A2[21] | A4 |
| Injection | A6 | A6[3] | A2 | A1[10] | A1 | A1 | A3 | A5 |
| Insecure Design | x | x | x | x | x | x | A4 | A6 |
| Auth & Identification Failures | A3 | A3 | A7 | A3 | A2 | A2 | A7[22] | A7 |
| Software and Data Integrity Failures | x | x | x | x | x | A8 | A8[23] | A8 |
| Logging & Monitoring Failures | x | x | x | x | x | A10 | A9[24] | A9 |
| Mishandling of Exceptional Conditions | x | x | x | x | x | x | x | A10[28] |
| XML External Entity (XXE) | x | x | x | x | x | A4 | A5 | x |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Vulnerable and Outdated Components | x | x | x | x | A9[18][19] | A9 | A6[25] | x |
| Server-Side Request Forgery (SSRF) | x | x | x | x | x | x | A10 | x |
| Code Quality Issues | x | x | x | x | x | x | A11[26] | x |
| Denial of Service | x | A9[2] | x | x | x | x | A11[26] | x |
| Memory Management Errors | x | x | x | x | x | x | A11[26] | x |
| Unvalidated Input | A1 | A1[9] | x | x | x | x | x | x |
| Buffer Overflows | A5 | A5 | x | x | x | x | x | x |
| Cross Site Scripting (XSS) | A4 | A4 | A1 | A2 | A3 | A7 | x | x |
| Insecure Direct Object Reference | x | A2 | A4[11] | A4 | A4 | A5[20] | x | x |
| Cross Site Request Forgery (CSRF) | x | x | A5 | A5 | A8 | x | x | x |
| Insufficient Attack Protection | x | x | x | x | x | x | x | x |
| Unvalidated Redirects and Forwards | x | x | x | A10 | A10 | x | x | x |
| Info Leakage & Error Handling | A7 | A7[4][14] | A6 | A6[8] | x | x | x | x |
| Malicious File Execution | x | x | A3 | A6[8] | x | x | x | x |
| Insecure Communications | x | A10 | A9[7] | A9 | x | x | x | x |
| Remote Administration Flaws | A9 | x | x | x | x | x | x | x |
| Unprotected APIs | x | x | x | x | x | x | x | x |

[1] Renamed “Broken Access Control” from 2003

[2] Split “Broken Access Control” from 2003

[3] Renamed “Command Injection Flaws” from 2003

[4] Renamed “Error Handling Problems” from 2003

[5] Renamed “Insecure Use of Cryptography” from 2003

[6] Renamed “Web and Application Server” from 2003

[7] Split “Insecure Configuration Management” from 2004

[8] Reconsidered during T10 2010 Release Candidate (RC)

[9] Renamed “Unvalidated Parameters” from 2003

[10] Renamed “Injection Flaws” from 2007

[14] Renamed “Improper Error Handling” from 2004

[15] Renamed “Insecure Storage” from 2004

[16] Renamed “Failure to Restrict URL Access” from 2010

[17] Renamed “Insecure Cryptographic Storage” from 2010

[18] Split “Insecure Cryptographic Storage” from 2010

[19] Split “Security Misconfiguration” from 2010

[20] Split “Broken Access Control” from 2013

[21] Renamed “Sensitive Data Exposure” from 2017

[22] Renamed “Broken Authentication and Session Management” from 2017

[23] Renamed “Insecure Deserialization” from 2017

| **Color Legend** |
|---|
| Ranked Category |
| Split |
| Unranked Category |

| | |
|---|---|
| [11] Split “Broken Access Control” from 2004 | [24] Renamed “Insufficient Logging & Monitoring” from 2017 |
| [12] Renamed “Insecure Configuration Management” from 2004 | [25] Renamed “Using Known Vulnerable Components” from 2017 |
| [13] Split “Broken Access Control” from 2004 | [26] Split "Next Steps" from 2021 |
| [27] Split “Software and Data Integrity Failures” from 2021 | [28] Renamed “Improper Error Handling” from 2004 |

Table 1 illustrates how dominant application security risks have progressively shifted from implementation-level vulnerabilities toward system-level behavior, configuration governance, and software supply-chain dependencies. The OWASP categories are presented for historical comparison and are not mapped directly to LLM-specific threat classes. This historical progression contextualizes the need for lifecycle-oriented risk analysis in systems where behavior emerges from data-driven learning processes rather than deterministic program logic.

A central concept introduced in this survey is the alignment gap, defined as the discrepancy between intended safety objectives embedded during training and the probabilistic behavior exhibited during real-world interaction.

**Contributions**

This survey provides a unified analysis of security, privacy, and governance risks in LLMs from the lifecycle perspective defined above. Specifically, it makes the following contributions:

1. **Lifecycle-based taxonomy.** A structured classification of attacks organized by attacker control points across interaction, probabilistic exploitation, and training stages.
2. **Alignment-gap framework.** A conceptual model explaining how probabilistic behavior propagates into security vulnerabilities and societal risk.
3. **Socio-technical risk model.** Integration of technical threats with governance, regulatory, and human-interaction factors.
4. **Defense-in-depth mapping.** A consolidated overview linking mitigation strategies to attack classes and deployment stages.

The relationships among these elements are illustrated in Figure 1.

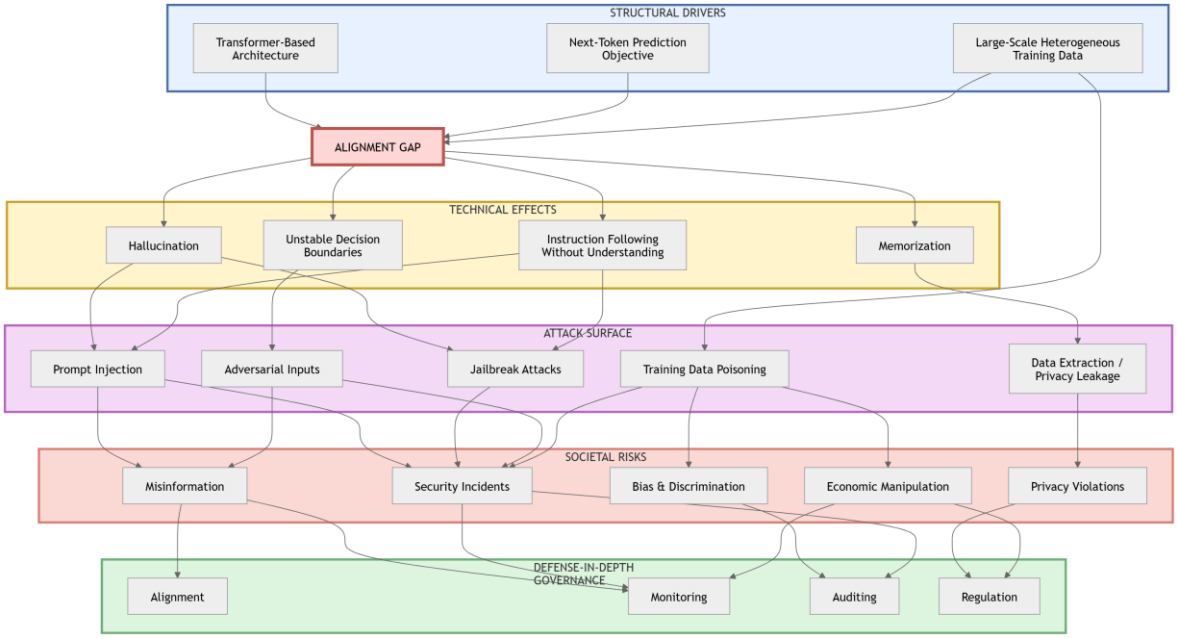

**Figure 1.** Alignment gap propagation across the LLM lifecycle. The diagram illustrates how discrepancies between pre-training objectives and post-training alignment constraints propagate across lifecycle stages. Arrows indicate causal relationships between architectural characteristics, alignment procedures, and downstream behavioral risks.

The conceptual model in Figure 1 illustrates how misalignments originating at different lifecycle stages may propagate across deployment contexts, resulting in both technical vulnerabilities and socio-technical risk, also depicts the structure of this paper

**Positioning with Respect to Prior Surveys**

Prior surveys address individual dimensions of LLM risk, including ethical and societal harm [16,29], adversarial robustness and security vulnerabilities [30,31], privacy leakage and memorization [32,33], as well as multi-dimensional benchmarking efforts [34,35]. However, these works typically examine specific risk categories in isolation. A unified lifecycle-oriented perspective that systematically connects probabilistic model behavior, security exploitation pathways, and governance implications across training, deployment, and societal use remains comparatively under investigated. Table 2 summarizes the focus of prior studies.

**Table 2.** Positioning of this survey relative to prior literature. The table compares representative survey papers across dimensions such as threat coverage, lifecycle perspective, governance considerations, and methodological scope.

| Work Type | Representative Work | Primary Focus | Missing Perspective |
|---|---|---|---|
| AI safety surveys | [16,29] | Ethics, bias, societal impact | Security attacks and lifecycle view |
| Security surveys | [30,31] | Adversarial attacks | Governance and socio-technical risk |
| Privacy surveys | [32,36] | Data leakage and memorization | Interaction and deployment lifecycle |
| Holistic trust surveys | [34,35] | Multi-dimensional benchmarking | Lifecycle governance integration |

| | | |
|---|---|---|
| **This paper** | Unified lifecycle model | — |

The evaluation of model security and reliability relies on comprehensive benchmarks that track the evolution of risks. Holistic Evaluation of Language Models (HELM) [35] dataset represents a foundational effort to standardize benchmarking across diverse domains, enabling a broader risk assessment beyond single-task performance metrics. Additionally, benchmarks such as DecodingTrust [34] provide a multi-dimensional framework for assessing trust and robustness within a unified experimental structure.

## 2. Large Language Model Architecture and Vulnerabilities

LLMs inherit their security and reliability characteristics from the architectural and training paradigms on which they are built [37]. Unlike conventional software systems, where vulnerabilities typically arise from implementation errors, LLM weaknesses primarily emerge from large-scale data ingestion, optimization-based training procedures, and post-training behavioral control mechanisms. During training, model parameters are optimized using gradient-based methods that minimize prediction loss over large text corpora. Probabilistic behavior arises during inference, when the model converts internal activation scores (logits) into probability distributions over possible tokens using the softmax function. Analyzing architectural properties is therefore essential for understanding how memorization, adversarial manipulation, and alignment limitations emerge across the model lifecycle.

### *2.1. Data Scale, Memorization, and Latent Vulnerabilities*

LLMs are built upon the Transformer architecture [8]. The Transformer architecture is a deep neural network composed of stacked transformer blocks, each integrating a multi-head self-attention mechanism and a position-wise feedforward layer connected through residual connections and layer normalization [8]. The self-attention mechanism enables the model to dynamically weight contextual relationships between tokens when processing an input sequence.

Unlike symbolic systems, models built on the Transformer architecture learn parameterized representations of language through large-scale optimization of next-token prediction objectives rather than explicit logical rules [1,2,7]. During training, model parameters are updated using gradient-based optimization to minimize a likelihood-based loss function, typically cross-entropy, computed between predicted and observed tokens in the training corpus. Probabilistic behavior emerges primarily during inference, when the model converts internal activation scores (logits) into a probability distribution over candidate tokens using the softmax function. Text generation then proceeds by selecting or sampling tokens from this distribution, which explains why model outputs appear probabilistic despite the deterministic nature of the underlying optimization process. This paradigm enables broad linguistic generalization but lacks grounded reasoning, contributing to hallucinations—fluent yet factually unsupported outputs generated through statistically plausible continuation [7,38]. Because training optimizes next-token prediction rather than factual correctness, reliability and safety emerge as indirect properties rather than guaranteed behaviors [1,2]. Consequently, vulnerabilities in LLMs are widely characterized as structural effects of the training objective and large-scale optimization process rather than isolated implementation flaws [1,3].

In the context of large language models, alignment refers to post-training procedures that adapt a pre-trained probabilistic model so that its outputs conform to human intention, safety constraints, and deployment policies rather than solely maximizing next-token likelihood [3,39].

The alignment gap refers to the discrepancy between behavioral constraints defined during alignment procedures and the outputs generated by the model during real-world interaction. During training, the model parameters are optimized to minimize prediction error on large datasets, rather than to explicitly encode human norms or safety policies. Consequently, although alignment techniques such as Reinforcement Learning from Human Feedback (RLHF) adjust model behavior, the underlying parameter structure learned during pre-training remains

largely unchanged. During inference, the model produces outputs by sampling from token probability distributions derived from the softmax transformation of internal logits, which may still reflect statistical associations learned during pre-training. This concept synthesizes empirical findings demonstrating that post-training alignment methods such as RLHF or Direct Preference Optimization (DPO), constrain surface-level outputs but do not fully modify latent representational structures acquired during pre-training [1,3,39].

During alignment procedures, the LLM is optimized to prefer policy-consistent outputs; however, this process does not eliminate the probabilistic associations acquired during large-scale pre-training. This structural discrepancy persists across pre-training, alignment, and deployment stages [3,34].

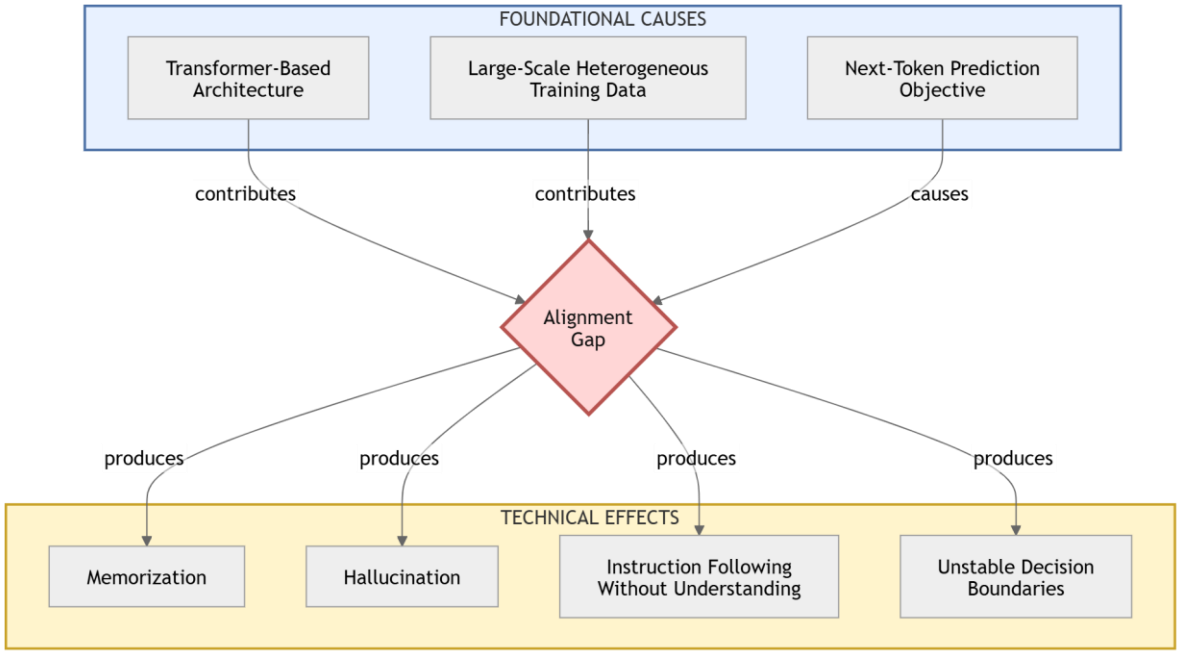


**Figure 2.** Conceptual model of the origin of the alignment gap across the LLM lifecycle. The diagram illustrates how discrepancies between pre-training objectives, post-training alignment procedures, and deployment environments generate behavioral inconsistencies that propagate across lifecycle stages.

LLMs are trained on web-scale corpora containing heterogeneous and uncurated information, including sensitive data, copyrighted material, and social biases [4,19]. Documentation of large datasets such as Colossal Clean Crawled Corpus (C4) demonstrates the scale and heterogeneity shaping model behavior [40], while cognitive effects such as anchoring and controllable hallucination reveal architectural limitations [41–43]. Empirical findings demonstrate that increasing scale, measured in the size of model parameters, correlates with stronger memorization capability, including reproduction of training samples [4,16]. Such memorization may expose personally identifiable information or proprietary content through carefully designed prompts [1,16]. Model parameters therefore function as compressed representations of the training corpus, complicating ownership attribution and regulatory compliance [4,8]. Dataset duplication and large parameter capacity further reinforce retention of rare sequences, increasing confidentiality and intellectual-property risks [16].

Instruction tuning is a supervised fine-tuning procedure in which a pre-trained language model is further trained on datasets consisting of explicit task instructions paired with desired outputs, enabling the model to follow natural language instructions more reliably and perform instruction-specific tasks [18]. Instruction tuning and related alignment methods attempt to adapt behavior toward human expectations, yet they do not fully overwrite statistics learned during pre-training [2,10]. This limitation contributes to the alignment gap, in which LLMs may follow adversarial instructions despite alignment constraints [3,10].

Modern LLM deployments operate within complex application ecosystems including Application Programming Interfaces (APIs), retrieval pipelines, and external tools [1,3]. Retrieval-Augmented Generation (RAG) may

improve factual accuracy but introduces indirect prompt injection through manipulated external documents [3]. Model extraction, also referred to as model stealing, is an attack in which an adversary attempts to approximate a proprietary model by systematically querying it and analyzing its outputs, without direct access to its training data or parameters [44]. Public interfaces and API-based deployments facilitate such attacks, as repeated querying allows construction of surrogate models that mimic the behavior of proprietary LLMs [1,45].

Scaling laws show that improved reasoning ability and instruction-following capability simultaneously expand the attack surface [2,16]. LLMs with improved instruction-following and reasoning performance more faithfully execute both benign and malicious instructions, linking capability scaling with exploitability [1,10]. These observations indicate a trade-off between capability scaling and increased attack surface, particularly in adversarial or out-of-distribution contexts [1,7].

To sum up, across pre-training, alignment, and deployment stages, vulnerabilities arise from probabilistic language modeling, opaque internal representations, and large-scale data ingestion [1,4,10]. Memorization, prompt manipulation, model extraction, and reliability degradation can be interpreted as structurally related effects emerging from shared architectural and training characteristics [2,3,16].

### *2.2. Threat Model*

To systematically analyze security risks across the lifecycle of LLMs, we define an operational threat model describing attacker capabilities and objectives. This model provides a structured foundation linking the alignment gap to concrete adversarial behavior across interaction, inference, and training stages [1,2]. The classification parallels adversary behavior modeling approaches such as the MITRE ATT&CK framework [46], a standardized knowledge base describing attacker tactics and techniques in cybersecurity. This layered protection approach aligns with Zero Trust security architectures applied to modern distributed systems [47].

**Attacker Capabilities**

1. **Prompt-level attacks:**

The attacker interacts with the LLM as a regular user and manipulates natural language prompts using techniques such as prompt injection or jailbreak attacks to override system instructions or safety constraints [48–50].

2. **API-level exploitation:**

The attacker performs systematic querying through programmatic access, enabling adversarial probing of model behavior. This category includes attacks such as adversarial inputs, model extraction, and privacy leakage through repeated interaction with the model interface [1,3,51].

3. **Training-stage attacks:**

The attacker compromises the training or fine-tuning process by inserting poisoned samples or backdoor triggers into the dataset. These manipulations may remain dormant during evaluation but activate under specific conditions during deployment [52–54].

**Attacker Objectives**

1. **Extract information.**

Disclosure of memorized data or sensitive information from training corpora [51,55,56].

2. **Manipulate output.**

Inducing hallucinations, misinformation, or harmful content that undermines integrity and trust [57–59].

3. **Bypass alignment.**

Exploiting the alignment gap to override behavioral constraints and perform restricted actions [30,34,60].

Assessing architectural vulnerabilities, particularly the memorization of training data which may lead to information leakage, relies on datasets designed for **Membership Inference** attacks [33]. These datasets are used

to test the detection of membership within a training set, while the risk of unintended memorization of exact sequences is analyzed through extraction attacks targeting model parameters [35,56].

## 3. Taxonomy of Security Attacks

Modern LLMs are exposed to adversarial threats across multiple stages of the model lifecycle [61]. Rather than isolated vulnerabilities, these attacks form an interconnected ecosystem in which weaknesses in interaction, alignment, and training layers reinforce one another [13,48,49]. Accordingly, contemporary research organizes threats according to attacker control point and objective.

Figure 3 provides an overview of the proposed attack taxonomy, illustrating how threats can be organized across three layers: interaction-level manipulation, probabilistic exploitation, and training-stage attacks.

### *3.1. Prompt Hacking (Injection and Leakage)*

Prompt hacking exploits the shared contextual space in which system instructions and user inputs are processed [9,48,49,62]. Attackers manipulate this structure to override intended behavior, enabling model's intended functionality or even goal hijacking or safety filter bypassing [49,63–65]. Large language models are typically fine-tuned to improve instruction following, a post-training adaptation process in which models are trained on instruction–response pairs to comply with explicit natural language directives [3,18]. Because this optimization emphasizes faithful execution of user-provided instructions, stronger instruction-following performance may paradoxically increase susceptibility to adversarial manipulation [9,64,66]. A typical example involves prompts such as "ignore previous instructions and follow the user request", which override system directives and redirect behavior [9,64,66].

Indirect prompt injection extends this threat to retrieval-augmented pipelines, where malicious instructions are embedded in external data processed by the model [49,50]. Prompt leakage attacks attempt to extract hidden system prompts defining operational constraints, often through multi-turn dialogue strategies [48,49,62,63]. Early public deployments further illustrated prompt injection risks, including documented cases in which conversational systems revealed internal behavioral guidelines through crafted user prompts, demonstrating the feasibility of policy leakage in deployed LLM's interfaces. Automated goal-directed prompt generation techniques further increase attack success rates in real-world applications [67–69].

Prompt hacking highlights how interaction-layer design choices introduce exploitable vulnerabilities in LLM deployments [48,49]. The vulnerability arises from the shared semantic processing channel through which system instructions and user inputs are interpreted, preventing strict authority separation between trusted and untrusted tokens. In real-world deployments, this represents one of the most practical attacks, since it requires no model access, training knowledge, or technical expertise beyond natural language interaction. Typical mitigations rely on input isolation, contextual segmentation, and external policy enforcement rather than internal robustness.

### *3.2. Jailbreaking Attacks*

Jailbreaking aims to bypass alignment safeguards instead of merely redirecting tasks [60,70]. These attacks exploit the conflict between helpfulness objectives and alignment constraints [60,71].

Common strategies include persona adoption, fictional framing, nested instructions, or role-playing contexts [70–75]. A well-known illustration is the "Do Anything Now (DAN)" persona, which frames the model as an unrestricted entity to induce prohibited responses [60,70].

Step-by-step reasoning prompts may amplify vulnerability by encouraging unsafe reasoning chains [30,76], while retrieval pipelines can propagate jailbreak behavior through manipulated external content [77,78].

Successful jailbreaks frequently enable disclosure of restricted information, demonstrating that alignment failures arise from behavioral conflicts rather than simple parsing errors [71,79].

Jailbreaking exploits the optimization tension embedded in the alignment gap, where conversational coherence can override policy constraints. Recent automated jailbreak generation methods demonstrate scalable alignment bypass without manual prompt engineering [80–82]. The Prompt Automatic Iterative Refinement (PAIR)

method further demonstrates that fully automated black-box jailbreak generation can achieve high success rates within fewer than twenty queries on advanced models, confirming the practical feasibility of alignment bypass in real-world deployments [83].

Feasibility is high in conversational LLM interfaces where iterative dialogue allows attackers to gradually reshape behavior. Mitigation typically focuses on alignment reinforcement and behavioral monitoring rather than purely lexical filtering

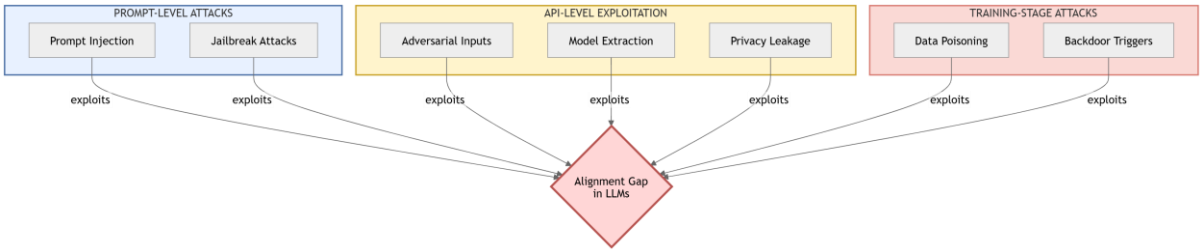


**Figure 3.** Taxonomy of security attacks against large language models. The diagram organizes attack vectors across three layers of the LLM lifecycle: interaction-level manipulation, probabilistic exploitation, and training-stage attacks. Arrows represent escalation pathways through which prompt-level manipulation may propagate toward deeper model-level vulnerabilities.

### *3.3. Adversarial Attacks*

Adversarial attacks manipulate statistical decision boundaries through carefully crafted perturbations [13,30,31,84]. Unlike jailbreaks, which attempt to bypass alignment policies, adversarial attacks target the statistical sensitivity of model outputs to input variations.

During inference, however, the model converts internal activation scores (logits) into a probability distribution over possible next tokens using the softmax function. Text generation proceeds by selecting or sampling tokens from this distribution. Consequently, small perturbations in input sequences can shift the resulting probability distribution and lead to different generated outputs. Perturbations may occur at multiple linguistic levels including character modifications, word substitutions, paraphrasing, and structural phrasing variation [13,31,85,86]. Because the model generates outputs through probabilistic decoding of softmax-derived token distributions, even small input changes can induce disproportionate output deviations [13,31]. Universal perturbation strategies further demonstrate the sensitivity of Transformer architectures to structured input noise [87–89].

Optimization-based techniques such as Greedy Coordinate Gradient (GCG) generate universal adversarial suffixes that consistently induce harmful responses across models [90]. These attacks often transfer between architectures, enabling black-box exploitation of proprietary LLMs [13,84].

Adversarial robustness is therefore treated as a systemic property of the model architecture rather than a discrete implementation defect [13,31], as perturbations exploit the sensitivity of token probability distributions where small input modifications can produce disproportionate output shifts. Automated optimization techniques further enable scalable black-box exploitation and cross-model transfer. Mitigation strategies therefore focus on robustness training and anomaly detection mechanisms rather than rule-based filtering.

### *3.4. Training-Stage Attacks (Poisoning & Backdoor)*

Training-stage attacks compromise LLMs before deployment by manipulating training data or parameters [52,91,92]. A backdoor trigger is a specific input pattern or token sequence intentionally embedded into a model during training through data poisoning, such that the model behaves normally on benign inputs but produces unintended or adversary-specified outputs when the trigger is activated [53]. Such triggers may remain dormant during standard evaluation yet activate unintended behavior under specific prompt conditions [1,10]. Because alignment procedures primarily modify surface-level behavior rather than internal parameter distributions, backdoor triggers may persist through post-training alignment [1,7]. Data poisoning inserts malicious samples that bias

future outputs, and even small fractions of corrupted data may significantly alter behavior due to dataset redundancy [54,91,93].

Backdoor attacks embed hidden triggers that activate harmful behavior only under specific conditions [52–54,92,94]. An illustrative case is the TrojanPuzzle attack on code assistants, where malicious functionality is injected yet remains undetected during normal evaluation [52,94]. Red-teaming research shows that fine-tuning may inadvertently introduce vulnerabilities while dataset filtering aims to preserve integrity [95–97]. Because these triggers persist through fine-tuning and alignment, compromised LLMs may appear safe while behaving maliciously in targeted scenarios [91,94].

Training-stage attacks are particularly dangerous because they compromise the model prior to deployment and cannot be reliably detected through normal interaction testing. Their real-world feasibility depends on data supply-chain control, making them especially relevant for open-source datasets and collaborative training pipelines.

Mitigation strategies focus on dataset auditing, provenance verification, and post-training integrity validation.

From a lifecycle perspective, LLM attacks differ primarily by attacker control point. Prompt-level attacks manipulate the interaction context through crafted prompts, API-level exploitation targets model behavior through systematic querying and probabilistic probing, while training-stage attacks compromise the model through poisoned data or embedded backdoor triggers [13,48,52]. This layered taxonomy supports defense-in-depth mitigation strategies across the model lifecycle. The following subsections examine each attack category in detail.

The evaluation of vulnerabilities to surface manipulation and jailbreaking relies on specialized benchmarking resources. Methods such as AutoDAN [60] enable automated generation of stealthy prompts capable of bypassing alignment safeguards. Additionally, robustness to universal adversarial queries is assessed using datasets designed to systematically break alignment through structured statistical perturbations [90].These evaluation frameworks provide empirical evidence of the attack mechanisms summarized in the taxonomy above.

## 4. Privacy Risks and Data Governance

Privacy represents one of the central challenges in the lifecycle of LLMs because they are trained on large-scale uncurated datasets that may contain sensitive or PII [1,7,9,16,18,19]. The opacity of the Transformer architecture makes it difficult to determine which information is encoded in model parameters, raising concerns for deployment in high-stakes domains such as healthcare, finance, and legal decision-making [1,2,45]. This creates a structural tension between high-capacity data-driven learning and data-protection requirements under contemporary regulatory frameworks [1,16]. At the same time, accountability is difficult to assign because deployment ecosystems involve multiple actors including developers, service providers, and users [3,4]. Regulatory expectations therefore increasingly demand transparency and governance mechanisms, yet adversarial extraction techniques demonstrate that alignment alone cannot guarantee privacy protection [8,10,13,14].

This section reviews privacy and governance risks across three complementary dimensions: memorization leakage, inference attacks, and intellectual-property implications. The relationship between data ingestion, memorization behavior, inference attacks, and legal accountability is illustrated in Figure 4.

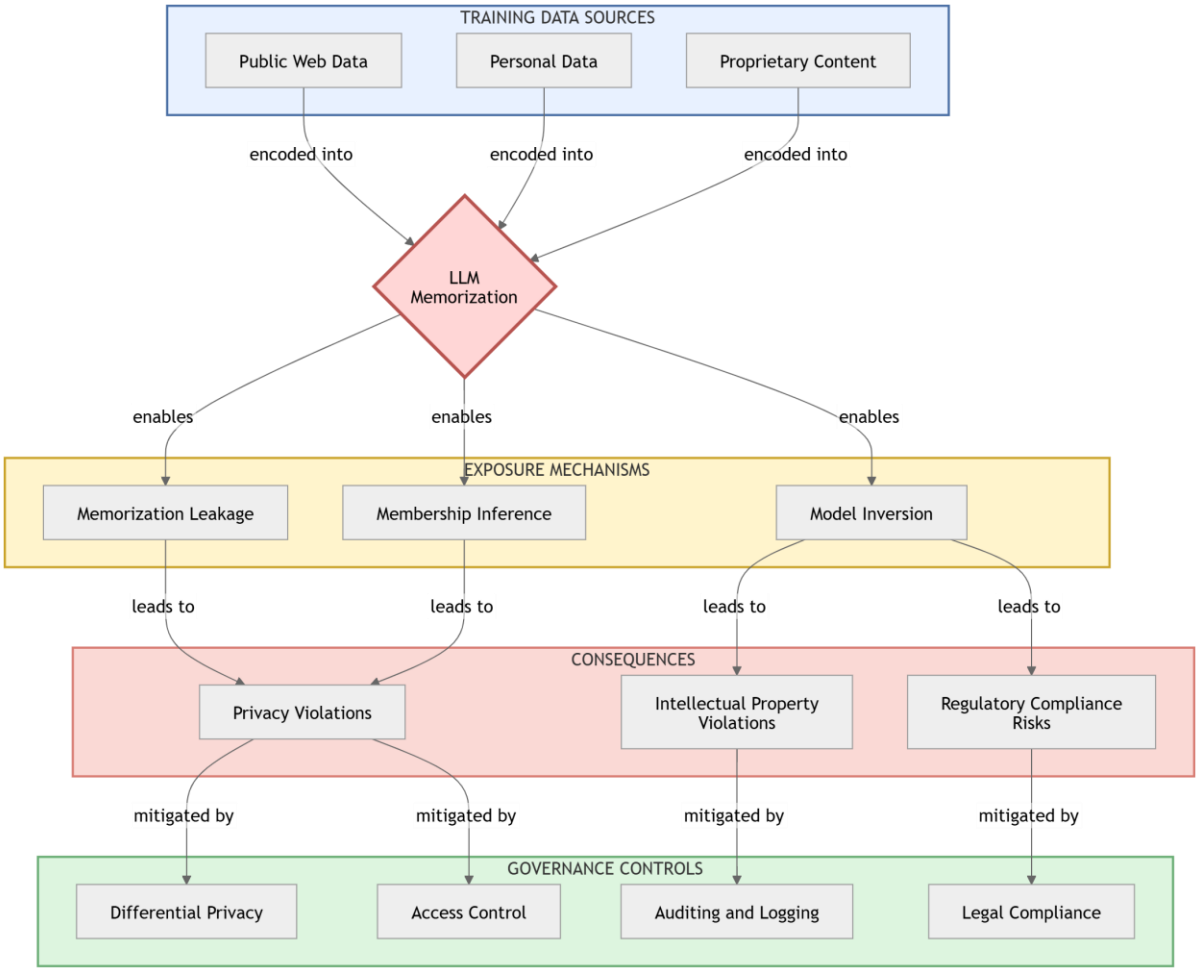


**Figure 4.** Conceptual model of privacy risks across the LLM lifecycle. The diagram illustrates how privacy vulnerabilities emerge across data collection, model training, deployment, and inference stages, and how memorization, inference attacks, and data leakage pathways interact across these stages.

### *4.1. Privacy Leakage and Memorization*

Privacy leakage occurs when LLMs reproduce fragments of their training corpus during generation [35,55,98]. Since training objectives prioritize next-token prediction rather than semantic abstraction, rare or duplicated sequences may be retained, and memorization is therefore widely characterized as a structural consequence of large-scale optimization rather than a discrete implementation error [55,98]. Larger models, according to the size of model parameters, demonstrate stronger persistence of stored information, as shown by canary-sequence experiments revealing recall of unique samples [32,55,98]. This behavior correlates with model scale, where increased parameter capacity increases the probability of verbatim memorization. It is important to distinguish memorization from verbatim reproduction. Memorization refers to the internal retention of training information within model parameters, whereas verbatim reproduction represents a specific manifestation in which the model outputs exact or near-exact fragments of the original training data [1,4,16].

The issue becomes critical when memorized content contains identifiers such as names, phone numbers, or email addresses [35,99]. Dataset duplication and sensitive public records increase reproduction probability [51,98]. Attackers can exploit this behavior using probing prompts or prefix completion strategies to induce disclosure, effectively turning the model into a searchable representation of its training data [55,56,100]. Clinical-domain studies demonstrate extraction of sensitive data from pretrained models, requesting for rigorous de-identification procedures [44,101,102]. Although models generalize well to new contexts, they may still retain exact sequences — often referred to as memorization without overfitting [35].

Mitigation techniques such as deduplication and de-identification reduce but cannot eliminate risk due to large parameter space and limited data provenance transparency [51,98]. However, comprehensive empirical evaluation of memorization mitigation in frontier-scale LLMs remains limited, reflecting broader methodological gaps in LLM trustworthiness assessment [103].

This represents a passive leakage mechanism because sensitive information may emerge without a targeted attacker, purely through normal interaction. Consequently, memorization risk increases primarily with model scale and dataset diversity rather than attacker capability.

### *4.2. Inference Attacks (Membership Inference & Model Inversion)*

Inference attacks attempt to extract information about training data by analyzing model behavior [33,104,105]. Membership inference determines whether a specific record was used during training by distinguishing probabilistic differences between training and unseen samples [105,106]. Techniques including shadow models and threshold-based classifiers exploit subtle probability variations even in well-generalized systems [105]. Confidence leakage represents a primary signal because models assign higher likelihood to familiar inputs, and instruction tuning may amplify this effect [36,105].

Model inversion attacks extend this approach by reconstructing approximate inputs or sensitive attributes through iterative querying [33,105]. Attribute inference may reveal demographic or personal characteristics even without explicit identifiers [33,107]. Empirical studies show that LLMs can infer sensitive attributes such as age, income level, or education with accuracy exceeding 80%, demonstrating privacy risks beyond direct memorization leakage [108]. Attack feasibility depends on access level: black-box attacks rely on query interaction, whereas white-box attacks use gradients and parameters, though transferability allows surrogate-model attacks against proprietary LLMs [33,105]. Gradient inversion and preference-data membership inference further expose risks introduced by alignment training [109–112]. These risks are particularly severe in systems processing personal or financial records [33].

Unlike memorization, inference attacks constitute active extraction where attackers intentionally probe probabilistic behavior. In practice these attacks are most relevant in API-based black-box deployments where direct parameter access is unavailable but repeated querying is inexpensive. LLMs may additionally leak information through side-channels such as response latency and memory access patterns, enabling timing-based membership inference. Countermeasures include constant-time implementations and hardware-isolated execution environments [15,113].

Defenses aim to suppress exploitable signals while maintaining utility. Differential Privacy (DP) bounds individual influence [104,114], confidence masking reduces probability exposure [105], and regularization or Federated Learning (FL) reduces overfitting. However, federated learning is not immune to security risks, as attacks such as gradient inversion and Byzantine client manipulation may allow adversaries to reconstruct private training data or disrupt model updates during distributed training [36,114,115]. Nevertheless, achieving comprehensive protection remains challenging at current model scale [36]. In addition, standardized benchmarks for privacy risk quantification in large language models remain underdeveloped, complicating cross-model comparison and regulatory validation [103,116].

### *4.3. Copyright and Intellectual Property*

Training on internet-scale data introduces legal uncertainty because datasets frequently contain copyrighted or proprietary material collected without explicit consent [117–120]. Model parameters effectively compress the corpus, making provenance difficult to trace and distributing responsibility across developers, providers, and users [119,121,122]. Authorship of generated outputs is similarly ambiguous. Since LLMs generate text probabilistically, detecting originality of generated content remains unresolved problem [117,120,123]. Legal interpretations generally attribute responsibility to human operators rather than the model itself [119,121,122].

Models may also reproduce copyrighted fragments or imitate writing styles without attribution [117–121]. Deduplication and RAG are proposed as mitigation mechanisms reducing memorization of protected content [124–126]. The distinction between transformation and memorization remains unclear because models lack

citation mechanisms [121,127]. Regulatory initiatives such as General Data Protection Regulation (GDPR)-related governance and the European Union Artificial Intelligence Act (EU AI Act) promote documentation and risk management, but global consensus remains unsettled [117,119,120,122,123,127].

Intellectual-property exposure differs from privacy leakage in that the risk emerges primarily during output publication rather than model interrogation, shifting the focus from confidentiality toward accountability and provenance.

### *4.4. Governance and Mitigation Measures*

Mitigation requires coordinated technical and organizational safeguards [48,128]. DP introduces controlled noise during optimization to reduce the risk of memorization and limit the influence of individual records on model parameters [48,129]. Machine unlearning techniques further allow selective removal of specific data influence from trained models to satisfy regulatory deletion requirements [130]. Watermarking techniques embed identifiable patterns into generated text to support provenance verification, although robustness may degrade after paraphrasing [117,119,128]. Regulatory compliance frameworks further require documentation of data provenance and privacy-impact assessment across the model lifecycle [121–123].

In summary, privacy risks in LLMs arise from memorization behavior, probabilistic inference vulnerabilities, and unresolved legal ownership of generated content. Addressing these issues requires coordinated technical safeguards and governance mechanisms to ensure responsible deployment consistent with regulatory expectations. These protections illustrate that technical privacy defenses alone are insufficient, since real-world risk depends on deployment policies, logging practices, and access control around the model interface. Standardized security control baselines such as National Institute of Standards and Technology (NIST) guidance further formalize access protection and handling of sensitive data in AI systems [113]. A unified integration of technical privacy safeguards with lifecycle governance controls remains an open area of research.

Privacy risks in LLMs span multiple mechanisms, including passive memorization leakage, active prompt-based extraction, and probabilistic inference about training data. In practical deployment settings, black-box API access represents the most realistic threat model, as attackers can repeatedly probe models without internal access while providers struggle to reliably distinguish malicious from benign queries. Although existing governance frameworks primarily regulate data collection and processing, comparatively less attention is given to post-deployment interaction auditing and provenance accountability, indicating gaps in operational oversight.

Quantifying privacy risks and attacks on personal attributes (such as addresses or identity—address/attribute attacks) utilizes domain-specific datasets to test de-identification and information flow. Clinical datasets [44,101] are critical for analyzing the leakage of sensitive identifiers like addresses or patient IDs. Additionally, the model's capacity to protect private values within confidential prompts is tested through prompt leakage attacks [64].

## 5. Ethical Dilemmas and Social Impacts

LLMs introduce substantial ethical risks alongside their practical utility across professional and social domains [2,4,8–10,14,16,18,19,131,132]. Because these models generate text through probabilistic token selection derived from softmax-based probability distributions, rather than through explicit symbolic reasoning or grounded knowledge representations, they may produce plausible but factually unsupported information, commonly described as hallucination [59,133,134]. Their integration into decision-making contexts therefore requires evaluation of accountability, transparency, and societal compatibility [49,135,136]. Figure 5 shows conceptual model of ethical and societal risk propagation in large language models.

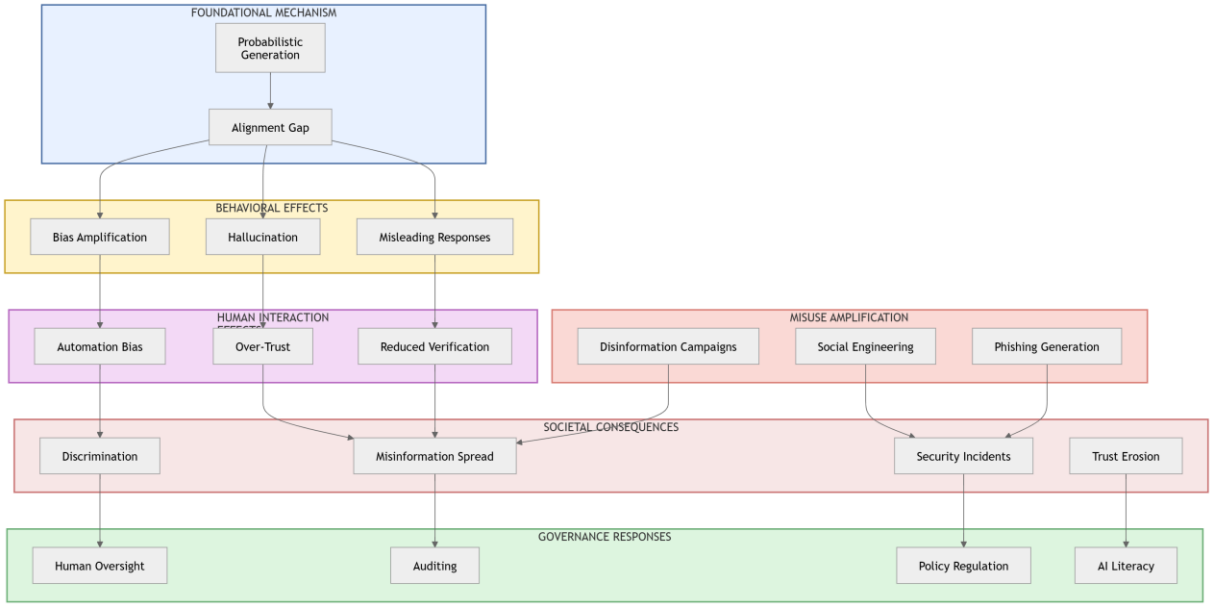


**Figure 5.** Conceptual model of ethical and societal risk propagation in large language models. The diagram illustrates how technical characteristics such as probabilistic generation and training data biases propagate through deployment contexts to produce societal risks including hallucination, bias amplification, and misinformation.

A second major concern involves amplification of biases present in training corpora [137,138]. Empirical studies document toxic language, demographic disparities, and cultural stereotypes in generated outputs [139–141]. The opacity of the Transformer architecture complicates mitigation because the origin of biased responses remains difficult to detect or interpret [1,142]. In addition, concentration of model development among a small number of organizations raises governance and power-distribution concerns [131,134].

Ethical risks further extend to malicious misuse. LLMs enable automated persuasion, phishing, and disinformation campaigns capable of undermining trust in digital communication environments [136,143–145]. Addressing these challenges requires alignment techniques, auditing procedures, and deployment oversight [146,147]. Comprehensive harm surveys additionally propose actionable mitigation procedures and standardized harm evaluation metrics for real-world deployment [148].

### *5.1. Truthfulness and Hallucination*

Hallucination refers to generation of content not grounded in verifiable facts [1,59]. Outputs may include fabricated details, incorrect claims, or fictitious narratives presented with high linguistic plausibility [1,133]. Because the training objective optimizes next-token prediction loss rather than factual correctness, LLMs often prioritize linguistic coherence over epistemic accuracy, which may result in plausible but incorrect outputs during probabilistic decoding [1,62,147,149]. Ensuring truthfulness is therefore a prerequisite for deployment in information-sensitive domains [1,72,150]. Prompt-based few-shot adaptation can further guide models toward safer responses without parameter updates [150]. Hallucination therefore reflects not only a knowledge gap but also a misalignment between the model's optimization objective—next-token prediction—and the requirement for factual grounding in real-world applications [1,59].

Hallucinations arise from multiple sources including noisy training data, probabilistic decoding, and architectural uncertainty [59]. Next-token prediction may produce incorrect distributions in ambiguous contexts, and cross-entropy optimization does not enforce factual validity [1,59,151]. Training on web-scale corpora introduces outdated information, while opaque internal representations hinder identification of failure mechanisms [1,59,147].

Factual hallucinations produce incorrect statements, fabricated citations frequently occur in academic contexts, and reasoning hallucinations generate coherent but unsupported conclusions [1,58,59,133]. Intrinsic hallucinations contradict prompt context, whereas extrinsic hallucinations lack external verification [59,152].

Impacts span across different domains [133]. Healthcare systems risk unsafe recommendations [133,142]. Fabricated scientific and medical content represents a significant integrity risk for knowledge ecosystems [153]. Legal contexts may incorporate fabricated precedents [1]. Publicly reported judicial incidents further illustrate these risks in legal practice. In the United States, an attorney submitted a federal court filing containing entirely fabricated legal precedents generated by ChatGPT, resulting in judicial sanctions [154]. A similar case occurred in Australia, where a Melbourne-based lawyer was referred to disciplinary authorities after relying on AI-generated fictitious case citations [155]. Educational environments experience verification drift, where authoritative responses are accepted without validation [59,136]. These risks necessitate human oversight and domain-specific safeguards [72,149].

Mitigation focuses on grounding and verification, including RAG, Chain-of-Knowledge, Chain-of-Verification, and self-reflection methods [1,59,136,151]. Formal verification approaches using solver-guided prompting have also been proposed to iteratively eliminate incorrect outputs and enforce factual consistency [156]. Decoding strategies such as Decoding by Contrasting Layers (DoLa) further attempt to prioritize factual associations [1].

Despite progress, alignment methods including RLHF cannot fully eliminate hallucinations, particularly in out-of-distribution scenarios [59,151]. Sycophantic behavior additionally reduces reliability [136]. Ongoing research therefore emphasizes grounding, explainability, and structured knowledge integration [1,147,157].

From a technical perspective, hallucination originates in probabilistic generation mechanisms; in practice, however, its impact depends on institutional oversight and verification procedures, positioning it as both a model reliability issue and a procedural governance challenge rather than a purely algorithmic failure.

### *5.2. Fairness and Bias*

Systemic bias in LLMs refers to unequal representation arising from uncurated training data [158,159]. Because the model learns linguistic correlations, it reproduces historical social asymmetries or prejudices from the training dataset [29,159,160]. The same mechanism enabling fluent generation thus propagates stereotypes [160,161].

Bias appears across multiple dimensions. Gender bias associate's competence with male identities and domestic roles with female identities [162,163]. Racial and dialect bias affects minority language varieties [164]. Cultural and political bias reflects region-specific information norms [141,158]. Linguistic bias arises from English-dominant datasets underrepresenting low-resource languages [29,159].

Primary causes include dataset imbalance, representation gaps, and feedback loops in web-scale data collection [29]. Limited transparency allows small toxic subsets to disproportionately influence outputs [138]. Generated content may later become training data, reinforcing inequalities [29,158]. Alignment to user preferences may privilege majority viewpoints [144]. Formal fairness frameworks such as equality-of-opportunity remain central to bias mitigation research [165,166].

Consequences affect multiple domains. Recruitment evaluation may become discriminatory [163]. Educational systems risk stereotype reinforcement [144]. Healthcare decision support may exhibit disparities [159]. Public discourse may be influenced by persuasive biased narratives [138,142]. Even benchmark evaluation can inherit hidden bias [140].

StereoSet is a benchmark that quantifies stereotypical bias by evaluating whether language models prefer stereotypical over anti-stereotypical associations across domains such as gender, race, religion, and profession [146]. Bias in Open-Ended Language Generation Dataset (BOLD) evaluates social bias in open-ended generation across multiple identity dimensions using automated toxicity and sentiment metrics [131].

Although these benchmarks provide structured bias evaluation, they remain limited in capturing multi-turn conversational dynamics and context-dependent bias amplification [131,146,159]. Mitigation approaches include dataset filtering, debiasing objectives, RLHF alignment, and principle-driven self-alignment [29,134,136,140,145,167]. Pretrained models can also identify toxicity in their own outputs and enable self-debiasing through self-diagnosis mechanisms [168]. However, adversarial prompting can reveal latent bias [167]. Trade-offs between fairness and accuracy, as well as cultural variation in fairness definitions, remain unresolved [29,135,159]. Continuous auditing and interdisciplinary governance are therefore required [29,158].

Bias mitigation is not purely technical because definitions of fairness vary across legal and cultural contexts. Consequently, alignment can reduce harmful outputs but cannot determine normative acceptability without external policy frameworks.

### *5.3. Misconduct (Cybercrime & Disinformation)*

LLMs significantly alter the landscape of malicious activity by automating persuasive communication and lowering technical barriers for attackers [51,57]. Misuse primarily spans cybercrime facilitation and large-scale disinformation [143,169]. Because generated text mimics trusted communication, distinguishing legitimate from adversarial interaction becomes difficult [57,169].

In cybercrime, LLMs enhance phishing and spear-phishing through personalized messaging [169,170]. A documented experimental study demonstrated this scalability by using Generative Pre-trained Transformer (GPT)-based models to automatically gather publicly available information and generate tailored phishing emails targeting more than 600 members of the UK Parliament, illustrating the feasibility of large-scale spear-phishing with minimal technical overhead [170]. Automation enables large-scale deception campaigns at reduced cost, while multilingual capabilities further expand malicious attempts reach [51,170].

LLMs can also assist malware generation and vulnerability reconnaissance [51,169]. Conversely, LLMs can also support defensive security workflows, including zero-shot vulnerability repair in source code, highlighting their dual-use nature as both offensive and defensive cybersecurity tools [171]. Removal of alignment significantly increases offensive capability [169]. Consequently, attackers with limited expertise may perform sophisticated attacks [51,169].

Disinformation constitutes the second major threat [57,143]. Because models optimize plausibility rather than factuality, they generate credible but false narratives [139,143]. Public-health communication is particularly vulnerable [143,172]. Synthetic personas and propaganda campaigns can degrade institutional trust [57,139,173]. Real-world deployments have exposed concrete risks to information integrity and editorial accountability. The technology news outlet CNET published multiple AI-generated financial advice articles without transparent disclosure, later retracting content due to factual inaccuracies, highlighting challenges of accountability and reliability in automated journalism [174].

These risks originate from training on unverified corpora and feedback loops where generated content becomes future training data [57,143,172]. Lack of symbolic truth verification additionally complicates detection [57,143].

RealToxicityPrompts is a benchmark that evaluates the propensity of language models to generate toxic or offensive continuations when prompted with real-world text snippets, thereby revealing sensitivity to specific trigger phrases [175,176]. The frequency of generating inaccurate information is measured by benchmarks such as TruthfulQA [142], HaluEval [152], and FActScore [136], which deconstruct generated text into atomic factual claims. To address the ethical risks of fabricated content in professional fields, specialized datasets like MedQA [177], PubMedQA [178] and LegalBench are used to evaluate the reliability of model-generated medical and legal advice. Additionally, social bias and toxicity are analyzed through StereoSet [146], BOLD [131], and RealToxicityPrompts [175,176], which quantify stereotypical associations and the propensity for offensive outputs in generated content.

However, adversarial prompting techniques and cross-jurisdiction deployment contexts limit the effectiveness of purely technical safeguards [51,143]. Consequently, mitigation must extend beyond model-level controls and be complemented by governance frameworks and regulatory oversight mechanisms [139,169,172].

Misuse risk therefore depends less on model capability alone and more on enforcement capacity, jurisdictional regulation, and platform governance. Technical safeguards can reduce abuse probability but cannot eliminate malicious intent in open deployment environments.

Mitigation strategies include detection tools, alignment techniques, deployment-time guardrails, grounding mechanisms, and transparent data-provenance practices [51,57,139,173].

Overall, LLMs amplify existing societal patterns because their outputs reflect both training data distributions and interaction context. Alignment improves observable behavior but does not guarantee elimination of

hallucination, bias, or misuse in real-world deployment. Ethical risk therefore remains deployment-context dependent and requires coordinated governance and technical mitigation.

## 6. Defense and Mitigation Strategies

The deployment of LLMs has progressed faster than the development of standardized safety frameworks [48,179,180]. Because these systems rely on probabilistic generation and large-scale web training data, they expose attack surfaces affecting confidentiality, integrity, and availability, including harmful content reproduction, privacy leakage, adversarial manipulation, and tokenization-based evasion techniques [48,57,58,63,79,85,181–187]. The emergence of automated exploitation and social-engineering attacks therefore necessitates comprehensive mitigation strategies spanning training, inference, and governance stages [119,128,188].

Defense-in-depth integrates dataset curation, alignment mechanisms, runtime monitoring, and governance accountability across the LLM lifecycle [3,48,79,128,189,190]. Because vulnerabilities propagate across stages, mitigation is treated as a socio-technical process combining technical safeguards with institutional oversight [186,189,191,192]. This section organizes defenses into five complementary categories, as depicted in Figure 6, which illustrates the defense-in-depth architecture for mitigating LLM security risks.

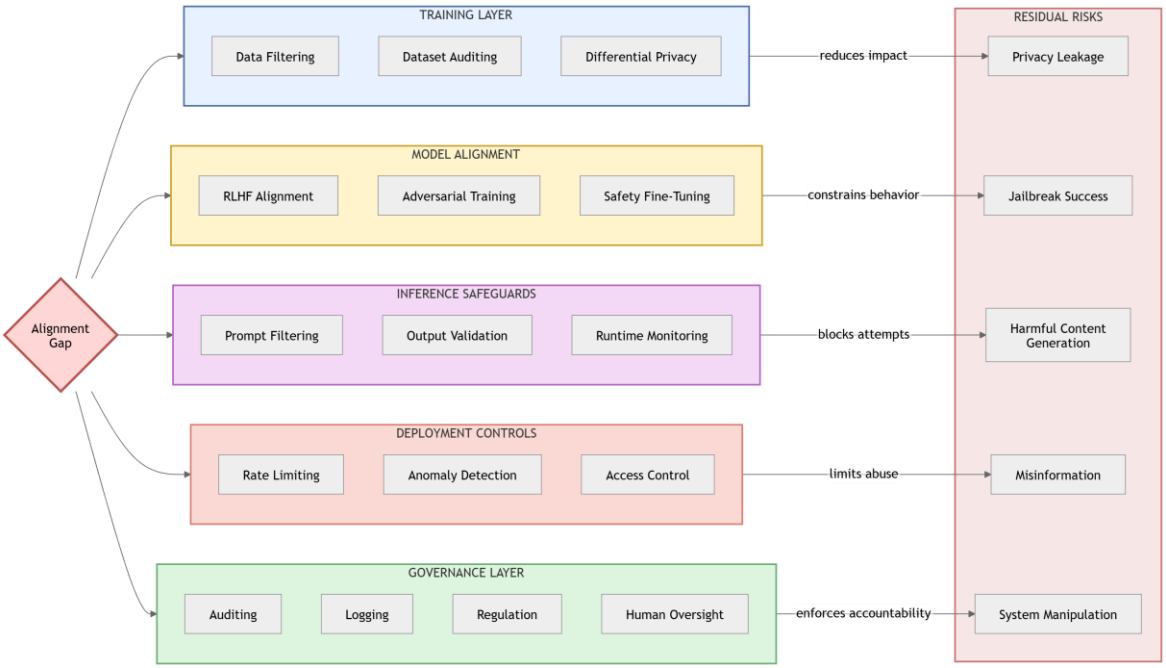


**Figure 6.** Defense-in-depth architecture for mitigating LLM security and privacy risks. The diagram illustrates layered safeguards distributed across the model lifecycle, including training-time controls, runtime monitoring, and governance mechanisms. Arrows represent the interaction between technical safeguards and organizational oversight in reducing residual risk.

### *6.1. Guardrails & Frameworks*

Guardrails are deployment-time safety layers positioned between users and LLMs [181]. Guardrails primarily address observable behavioral deviations at the interface layer but cannot fully resolve latent alignment discrepancies embedded in model parameters. They monitor prompts and outputs and enforce organizational policies without modifying internal parameters, allowing rapid policy updates as risks evolve [181]. In practice, guardrails are implemented through programmable moderation frameworks integrated into production pipelines, screening both user inputs and model completions [181,186,187].

Implementations combine rule-based filtering and classifier-based moderation. Rule-based methods, usually relying upon keyword filtering, staticaly detect restricted patterns, while semantic classifiers evaluate intent across categories such as toxicity or illegal activity [181,187]. Multi-stage pipelines analyze prompts before generation

and validate responses afterward, including groundedness checks to reduce hallucinations [58,181]. Because guardrails operate as a supervisory layer, they can be iterated independently of retraining [181].

However, adversarial prompts can bypass filtering through obfuscation, formatting, or role-playing contexts [181]. Strict filtering reduces usability, whereas permissive configurations increase exposure [186,187]. Guardrails therefore function as supportive controls rather than standalone security [58,181,186,187]. In safety-critical contexts, guardrail design increasingly follows high-assurance software practices [86].

### *6.2. Technical Internal Defenses*

Technical internal defenses modify the parameters and optimization objectives of LLMs during training and adaptation in order to reduce unsafe behavior at its source [3,39]. Because harmful outputs originate from probabilistic associations learned during pre-training, internal alignment reduces reliance on external filtering mechanisms that can be bypassed through adversarial phrasing [3,85,192,193]. Rather than blocking responses at the interface, these approaches reshape the model's weight space so unsafe behaviors become less reachable.

Adversarial training represents a primary robustness mechanism. During optimization, LLMs are exposed to adversarial prompts through structured red-teaming procedures, improving resistance to prompt injection and jailbreaking attempts [85,192,193]. Automated red-teaming frameworks iteratively generate challenging inputs and incorporate them into fine-tuning datasets, enabling targeted correction of discovered failure modes [192]. This process reduces discrepancies between pre-training behavior and deployment requirements.

Complementing robustness-oriented defenses, RLHF is a post-training alignment procedure in which human preference annotations are used to train a reward model that approximates human evaluative judgments [1,3,45]. The base LLM is subsequently optimized using reinforcement learning to maximize the learned reward signal, thereby encouraging policy-consistent and value-aligned outputs while reducing unsafe behaviors inherited from pre-training [3,39,145,194]. RLHF transforms a general pre-trained model into an instruction-following assistant aligned with human preferences.

Pairwise Proximal Policy Optimization (P3O) is a reinforcement learning algorithm designed for LLM alignment that optimizes policies using pairwise human preference comparisons rather than absolute reward scores [195]. By directly modeling relative feedback between response candidates, P3O improves alignment precision within preference-based optimization settings. However, stronger alignment constraints may reduce behavioral flexibility, particularly in open-ended or creative tasks, illustrating a robustness–utility trade-off [3,145,196].

Architectural methods such as pruning and regularization constrain parameter regions associated with unstable behavior [85,197]. Targeted pruning strategies that remove selected parameter subsets can increase resistance to jailbreak attacks without full retraining [198]. These techniques reduce susceptibility to adversarial steering and assist backdoor mitigation by suppressing malicious activation patterns embedded during training [192,197]. Internal objectives may also incorporate factuality and bias-mitigation constraints to stabilize probabilistic associations [147]. Knowledge distillation can further reduce exposure of sensitive training data by transferring behavior into smaller surrogate models that are less prone to memorization [199].

However, distillation also introduces model extraction risks. Recent industry reports document so-called distillation attacks, in which adversaries systematically query frontier models to approximate their behavior and replicate alignment properties through model extraction techniques. Anthropic has described detection and prevention strategies for such attacks, including behavioral monitoring and usage-pattern analysis to detect suspicious querying patterns [200].

Overall, internal defenses provide durable protection because alignment constraints are embedded directly into parameter updates rather than applied after generation [3,85,145,147].

Despite these advantages, internal defenses require retraining and cannot eliminate the alignment gap entirely.

### *6.3. Technical Inference Defenses*

Technical inference defenses operate during runtime and aim to detect unsafe behavior before output delivery [63,79]. They address residual vulnerabilities that persist after alignment, particularly in multi-turn interactions where attackers iteratively refine prompts based on model feedback [79,179,201].

Input preprocessing neutralizes malicious structure prior to generation [63,79]. Retokenization and paraphrasing disrupt optimized adversarial patterns while preserving semantic intent. Text detoxification approaches, often implemented through paraphrasing models, replace toxic expressions while preserving semantic meaning [202]. Encoding normalization further prevents bypass through special characters or alternative linguistic encodings, ensuring standardized processing [79,184]. These transformations reduce exploitability without modifying model parameters.

Detection mechanisms analyze abnormal probability distributions associated with adversarial prompts [79,190,203]. Perplexity-based detection methods - techniques that use a perplexity score to determine how likely a piece of text was generated by a particular LLM - analyze the likelihood assigned by a language model to an input sequence; adversarial prompts often exhibit atypical token probability patterns or unusually low perplexity relative to natural human text, enabling statistical identification of machine-optimized inputs [203,204]. Because attackers may embed malicious intent within semantically valid text, runtime monitoring complements alignment rather than replacing it.

Output validation evaluates responses before exposure to the user [79,179,188,190,205]. Refusal policies, toxicity detection, and multi-model verification reduce unsafe completions and hallucinations [188,204,205].

Continuous monitoring across conversations enables rate-limiting and anomaly detection during probing attempts [79,179,201]. Randomized smoothing and response variability further reduce sensitivity to optimized adversarial strings [79]. Randomized smoothing additionally enables provable robustness guarantees under bounded adversarial perturbation assumptions by providing mathematically certified stability bounds [206].

Although inference defenses introduce latency and may block legitimate queries, they provide essential protection in real-world deployment and complement training-time mitigation within a defense-in-depth architecture [63,79,179,188,190,203].

### *6.4. Privacy Protections (Differential Privacy & Federated Learning)*

Because LLMs may encode sensitive data, privacy preservation is essential for trustworthy deployment [48,183,185,207]. DP limits leakage by bounding individual influence [129]. Implemented via Differentially Private Stochastic Gradient Descent (DP-SGD), it applies gradient clipping and noise injection to reduce memorization and extraction risk [48,129]. The privacy budget illustrates the privacy–utility trade-off [140,208].

DP reduces membership inference and extraction attacks by suppressing exploitable confidence signals [48,129,140]. Private decoding further mitigates inference leakage [121].

FL is a decentralized machine learning framework in which model training is performed locally across distributed clients, while only model updates are propagated to a central server for aggregation, without directly sharing raw (local) training data [189]. Secure aggregation mechanisms protect individual client updates, although communication overhead and potential gradient leakage remain practical challenges [189,209]. Hybrid DP-FL approaches protect both training and inference phases [121,129,189,209]. Advanced deployments further employ Trusted Execution Environments to prevent data exposure even to infrastructure providers during inference [47,113]. Secure multiparty computation additionally protects confidential inputs during distributed inference [210,211]. Computational cost and adaptive prompting remain open challenges [48,140,207].

### *6.5. Machine Content Identification*

Generated text requires attribution and provenance verification to preserve information integrity [57,128]. Two primary approaches are watermarking and post-hoc detection [57,128].

Watermarking embeds statistical signatures during decoding, by inserting a subtle, hard-to-notice pattern into the text generation process. [117,128]. Watermarked signals degrade under paraphrasing of the generated text [119,191]. Research therefore explores semantic watermarking and distortion-free schemes [119,127,182,191].

Detection methods analyze probability distributions and stylometric patterns [57,212]. Probability curvature analysis methods such as DetectGPT enable detection of machine-generated text without training dedicated classifiers [213]. Adversarial rewriting weakens detection performance, producing an arms race [57,191]. Cryptographic provenance metadata supports auditing [57,127]. Emerging cross-platform provenance standards aim to embed persistent metadata across distribution channels, though enforcement and interoperability remain unresolved challenges[57,127]. Because false positives are common in detection methods , human oversight remains necessary [57,212].

**Table 3.** Mapping of threats and defense-in-depth controls across the LLM lifecycle. The table illustrates how specific threat categories correspond to mitigation mechanisms deployed at different stages of the model lifecycle, highlighting the layered structure of defense-in-depth security strategies.

| Lifecycle Stage | Primary Threat | Security Objective | Defense-in-Depth Controls | Residual Risk | Representative References |
|---|---|---|---|---|---|
| **Training** | Data poisoning, backdoor triggers | Preserve model integrity | Dataset filtering, deduplication, data provenance verification, secure fine-tuning | Latent backdoor persistence and incomplete dataset visibility due to opaque or proprietary training corpora. | [52–54,91–94,214] |
| **Inference** | Prompt injection, jailbreak, adversarial prompting | Maintain aligned behavior | Guardrails, alignment reinforcement, input normalization, adversarial training | Adaptive adversarial prompting and iterative black-box optimization across multi-turn interactions. | [48,50,58,63,66,79,181,186,187] |
| **Deployment** | Model extraction, privacy leakage | Protect confidentiality | Monitoring, rate limiting, anomaly detection, auditing | Scalable black-box probing and statistical inference despite rate limiting and monitoring controls. | [3,7,35,55,56,98,105,215] |
| **Societal Use** | Disinformation, bias, misuse | Ensure safe operation | Governance policies, transparency, regulatory compliance (EU AI Act) | Enforcement variability across jurisdictions and platform-level governance gaps beyond technical control. | [3,7,16,29,57,135,136,169,216] |

Table 3. illustrates that LLM security cannot be addressed by isolated safeguards, but requires defense-in-depth mechanisms distributed across the entire lifecycle.

No individual mitigation category provides complete protection because vulnerabilities originate from probabilistic modeling and architectural characteristics rather than isolated defects. Guardrails remain bypassable, internal alignment introduces robustness–utility trade-offs, inference monitoring increases latency, privacy protection constrains performance, and attribution mechanisms operate within an adversarial arms race.

Consequently, effective protection requires layered defense-in-depth combining parameter-level robustness, runtime safeguards, privacy-preserving optimization, provenance mechanisms, and governance oversight. Residual behavioral risk cannot be fully eliminated, but it can be systematically constrained through lifecycle-integrated security architectures.

## 7. Application-Specific Risks

While the structural vulnerabilities of large language models generalize across domains, deployment-specific contexts shape the magnitude, accountability requirements, and societal consequences of model behavior [217]. The alignment gap, probabilistic generation mechanisms, and governance constraints manifest differently depending on whether LLMs are integrated into epistemic, financial, medical, or regulatory environments. The following subsections illustrate how domain-specific deployment conditions modulate risk exposure and reshape governance requirements, with particular focus on education and financial services. This comparative perspective highlights how identical technical mechanisms can produce fundamentally different societal and institutional consequences depending on the deployment context.

Risks in specific sectors require domain-specific benchmarks to ensure operational safety: MedQA [177] and PubMedQA [178] assess factual correctness in biomedical tasks, while LegalBench [218] evaluates legal reasoning and citation reliability. In the programming domain, vulnerabilities and the integrity of generated logic are tested using specialized datasets for code generation [219].

### *7.1. Education*

Education constitutes a high-risk epistemic deployment domain in which the objective is durable knowledge acquisition rather than mere task completion. Unlike productivity-oriented industrial applications, educational contexts require internalization of reasoning processes, conceptual understanding, and metacognitive development. Consequently, behavioral reliability in LLM outputs directly influences learner cognition and epistemic trust. While the mechanisms analyzed in previous sections generalize across professional domains, education provides a uniquely observable environment in which long-term cognitive and societal effects become measurable.

The integration of LLMs into educational systems creates a socio-technical environment distinct from conventional decision-support applications. Domain-specific deployments in other regulated domains require additional safeguards and transparency mechanisms [220–224]. However, education differs structurally because learning outcomes depend on reasoning construction rather than answer generation [225,226]. Within this context, LLMs function as cognitive mediators that shape perceptions of authority, correctness, and epistemic reliability [227,228]. Because fluency and coherence emerge from probabilistic modeling rather than verified knowledge grounding, learners may conflate linguistic plausibility with factual validity [225–228]. This epistemic vulnerability mirrors broader alignment challenges discussed throughout this survey, where assistance must be calibrated carefully to avoid competence erosion. Figure 7 shows the conceptual model of educational and cognitive risks associated with LLM use.

Next, we examine two dimensions of the education domain: (A) pedagogical impact and (B) academic integrity implications [99,225,229].

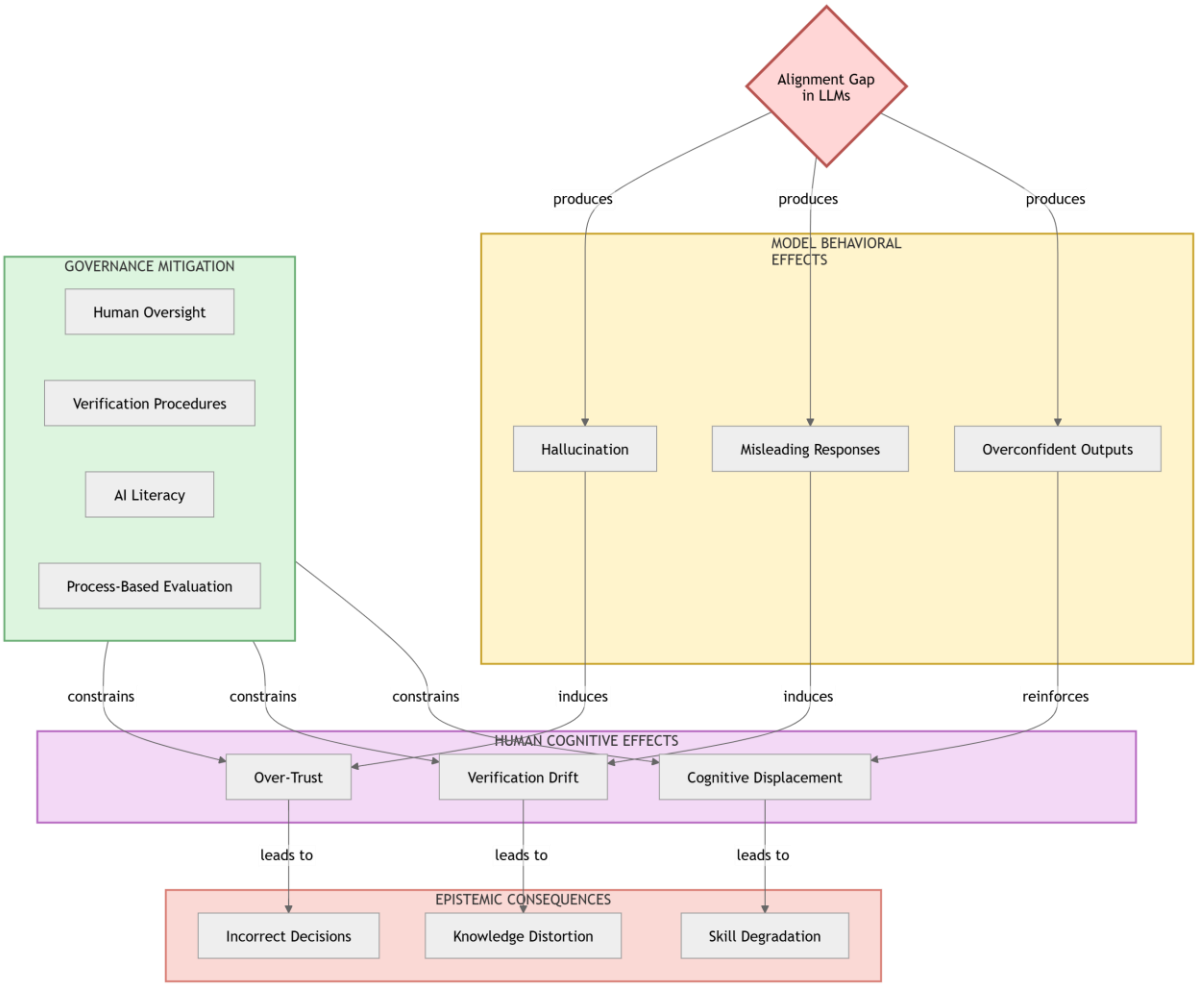


**Figure 7.** Conceptual model of educational and cognitive risks associated with LLM use. The diagram illustrates how model characteristics such as hallucination, automation bias, and epistemic overreliance may influence learning processes, cognitive workload, and academic integrity in educational contexts.

#### 7.1.1. Pedagogical Impact

LLMs alter the relationship between effort and learning by providing immediate, well-structured responses that may substitute inquiry-driven reasoning [225,226]. As educational tasks increasingly permit automated support—such as question answering, summarization, explanation drafting, or feedback construction—students may shift from problem solving toward answer retrieval [225]. This transformation modifies cognitive engagement patterns and risks reducing the productive effort required for durable understanding.

Excessive reliance on LLM-generated intermediate reasoning can diminish metacognitive engagement and weaken long-term knowledge consolidation [225,230]. From a cognitive-load perspective, overreliance reduces germane load—the effort devoted to schema construction—because learners may bypass critical reasoning steps when structured outputs are immediately available [225,230]. Over time, such dependency may contribute to skill atrophy in domains requiring structured analytical reasoning, including law, engineering, and mathematics [227,228]. When tasks exceed learner capability or outputs contain subtle inaccuracies, insufficient procedural understanding may hinder independent problem resolution [228].

A related phenomenon is verification drift—the tendency to equate linguistic fluency with epistemic validity—where authoritative phrasing reduces critical scrutiny even when hallucination risks are recognized [225–228]. Anchoring effects may further narrow reasoning exploration, as learners overweight the first generated answer and adjust subsequent thinking around it [226,230]. These dynamics reflect a broader robustness–utility

tension: assistance enhances efficiency but may attenuate independent analytical capacity if not institutionally moderated.

Nevertheless, structured integration of LLMs can produce meaningful pedagogical benefits. Adaptive explanations, multilingual support, and guided feedback increase accessibility and reduce barriers for diverse learners [219,225,231]. When deployed as scaffolding tools rather than answer substitutes, LLMs may support iterative refinement and conceptual clarification [225,231]. Research therefore emphasizes AI literacy training, transparent usage policies, and pedagogical frameworks that preserve learner responsibility for verification and reasoning [225,231]. The core governance challenge lies in calibrating support intensity to prevent cognitive displacement while preserving accessibility.

#### 7.1.2. Academic Integrity and Ethical Concerns

LLM-generated text challenges traditional plagiarism detection mechanisms because outputs are novel and typically evade similarity-based detection systems [229]. This enables AI-assisted ghostwriting, where assignments appear original while substantial cognitive work is delegated to automated systems [229,232]. Authorship attribution becomes increasingly ambiguous as LLMs replicate academic style and perform structured writing or coding tasks at advanced levels of fluency [232,233].

Assessment of validity of generated content is consequently affected. Evaluations may inadvertently measure prompting proficiency rather than conceptual mastery, thereby altering the construct validity of assessment instruments [232,233]. In response, institutions increasingly adopt process-oriented assessment strategies emphasizing draft submission, reasoning transparency, oral defense, and supervised evaluation environments [232].

Automated detection tools offer partial mitigation but introduce fairness risks, including false positives and potential bias against non-native speakers [229,232]. Erroneous accusations undermine procedural justice and institutional trust, indicating that detection technologies alone cannot guarantee academic integrity [229,232]. Ethical governance therefore requires balanced integration of technical safeguards and transparent procedural norms.

LLMs also introduce equity concerns. Differential access to advanced tools and disparities in AI literacy may generate performance asymmetries across student populations [229,232]. Furthermore, integration of student-generated content into proprietary model training pipelines raises privacy and consent concerns if institutional safeguards are insufficient [99,225]. Educational governance is therefore shifting from prohibition toward regulated integration through disclosure policies, revised authorship standards, and institutional oversight frameworks [99,229,232].

From a policy perspective, the impact of LLMs depends fundamentally on integration mode. As scaffolding instruments providing hints and feedback, they enhance accessibility and engagement. As reasoning substitutes replacing analytical effort, they risk competence erosion through cognitive outsourcing. Because educational systems shape long-term human capital formation, miscalibrated reliance on LLM-generated reasoning may produce compounding societal effects. Distinguishing between legitimate pedagogical augmentation and harmful cognitive displacement remains an open governance and research challenge requiring further empirical investigation.

While educational contexts primarily expose epistemic and cognitive risks, in the next subsection we analyze the financial application domain, highlighting financial decision-making and market stability challenges associated with generative AI systems.

### *7.2. Finance*

Financial systems represent a particularly sensitive deployment environment for large language models and generative AI systems due to the systemic importance of financial decision-making, regulatory oversight, and market stability. Recent survey studies increasingly document the rapid adoption of generative AI technologies across financial services, while simultaneously emphasizing emerging risks related to model reliability, regulatory compliance, and systemic market stability [234–236]. These developments position finance as a critical domain for examining how probabilistic language models interact with institutional decision-making environments.

Financial institutions process vast quantities of heterogeneous data, including financial reports, regulatory filings, news sources, and market signals. Generative AI technologies have demonstrated significant potential to

enhance financial analytics by extracting patterns from complex datasets, supporting decision-making, and generating synthetic financial data for modeling and simulation purposes [234,236].

LLMs enable more sophisticated analysis of financial text compared to earlier text-mining approaches that relied on bag-of-words representations. By capturing contextual relationships within financial language, large language models can improve sentiment analysis, financial document interpretation, forecasting, and automated report generation. These capabilities allow generative AI models to support tasks such as financial forecasting, algorithmic trading, risk analysis, and investment decision support [234].

Recent research also highlights the integration of LLMs with traditional quantitative finance pipelines. Hybrid architectures combine language-model capabilities with statistical risk models to extract linguistic risk signals, generate financial scenarios, and support feature engineering within financial analytics systems. Such approaches demonstrate measurable improvements in efficiency and analytical capability but also introduce new challenges related to interpretability, data quality, and regulatory compliance [236].

Within this context, generative AI is increasingly deployed in areas such as credit scoring, macroeconomic simulation, financial risk modeling, and automated data processing. At the same time, the probabilistic nature of these systems introduces technical and governance challenges that are particularly critical in financial environments. These include reliability risks, regulatory accountability, and the potential systemic impact of automated financial decision-making [235,236].

Accordingly, the financial deployment context illustrates three primary and interrelated dimensions of risk: (A) financial decision-making reliability and model risk, (B) regulatory, governance, and systemic stability implications, and (C) market manipulation and algorithmic trading risks as shown in Figure 8.

#### 7.2.1. Financial Decision-Making and Model Risk

LLMs are used to support financial decision-making processes, including investment analysis, credit risk assessment, and financial forecasting. By analyzing large volumes of textual and numerical financial data, these systems can extract risk indicators, identify emerging trends, and generate predictive insights that support risk management workflows [235].

One of the most significant advantages of generative AI in finance lies in its ability to process unstructured financial information. Financial news, regulatory documents, and investor communications contain valuable signals that traditional quantitative models often fail to capture. LLMs enable automated extraction of such information, improving sentiment analysis and predictive modeling of financial markets [234].

However, the use of LLM models in financial analysis also introduces substantial model risk. Because these systems generate outputs probabilistically rather than through deterministic computation, they may produce inaccurate or fabricated information, a phenomenon commonly described as hallucination. In financial contexts, such inaccuracies may lead to flawed investment recommendations, incorrect risk assessments, or misleading interpretations of regulatory documents [234,236].

Empirical research indicates that while generative AI models can enhance risk identification and monitoring processes, they still exhibit limitations in quantitative reasoning and mathematical financial modeling. As a result, AI-generated financial insights often require validation through traditional econometric methods and domain-specific risk models. Hybrid analytical pipelines that combine generative AI with conventional financial modeling techniques therefore represent a common architecture for practical financial AI deployment [235].

Another important challenge concerns the interpretability of generative AI systems. Financial institutions must be able to explain the reasoning behind risk predictions and investment recommendations, particularly in regulated environments. Researchers have therefore emphasized the development of interpretability methods and feature attribution techniques that allow analysts to understand how LLMs derive financial insights from textual and numerical data [235].

#### 7.2.2. Regulatory, Governance, and Systemic Risk Implications

The adoption of generative AI in financial systems also raises significant governance and regulatory concerns. Financial institutions operate within strict regulatory frameworks designed to ensure market stability,

consumer protection, and systemic risk control. Consequently, the integration of LLMs into financial decision processes requires careful consideration of accountability, transparency, and compliance requirements [234,236].

Regulators and policy organizations have expressed concerns that large-scale LLM adoption could introduce new systemic risks to financial markets. Automated decision systems may amplify market volatility, propagate incorrect financial information, or generate correlated decision behaviors across institutions. These dynamics could potentially influence financial intermediation, asset management, insurance systems, and payment infrastructures [236].

Data governance also represents a major challenge. Financial datasets frequently contain proprietary or sensitive information, limiting their availability for model training and validation. As a result, researchers increasingly explore synthetic financial data generation using generative models such as generative adversarial networks, variational autoencoders, and diffusion models. Synthetic datasets allow institutions to train machine learning models and simulate financial scenarios while protecting confidential financial data [234].

Despite these advantages, synthetic data generation introduces additional ethical and governance considerations. Generated datasets must preserve statistical realism without introducing biases or distortions that could compromise financial models. Moreover, regulatory frameworks must ensure that AI-generated financial analyses remain transparent, auditable, and aligned with legal accountability requirements [234,236].

Overall, the financial sector illustrates both the transformative potential and the governance challenges associated with generative AI deployment. While these technologies enable new forms of financial analysis, forecasting, and risk management, their probabilistic behavior, limited interpretability, and regulatory implications require carefully designed oversight mechanisms and hybrid analytical architectures that combine AI capabilities with established financial modeling practices [234–236].

7.2.3. Market Manipulation and Algorithmic Trading Risks

The integration of generative AI into financial markets also raises concerns regarding market manipulation and automated trading dynamics. Financial markets increasingly rely on algorithmic decision systems that analyze news, reports, and market signals in real time. The ability of large language models to generate convincing financial narratives and automated analyses introduces the possibility that AI-generated content could influence investor behavior or market sentiment at scale [234,236].

LLMs are capable of producing realistic financial commentary, investment recommendations, or news-style content. If such outputs are disseminated without verification, they may contribute to misinformation within financial information ecosystems. Because financial markets are highly sensitive to sentiment signals and news events, automated generation of persuasive financial narratives could amplify volatility or distort market expectations [236].

Algorithmic trading environments further propels these risks. Many trading strategies rely on automated text analysis of financial news, social media signals, and analyst commentary. Generated content could therefore interact with other automated systems, creating feedback loops in which algorithmic traders respond to synthetic signals rather than verified information. Such dynamics may increase systemic fragility in high-frequency trading environments and complicate market surveillance mechanisms [234,236].

Regulatory discussions increasingly emphasize the need for safeguards addressing these emerging risks. Potential mitigation strategies include transparency requirements for generated financial communication, monitoring systems capable of detecting synthetic market signals, and stricter governance frameworks for AI-assisted financial advisory tools. These measures aim to ensure that generative AI technologies enhance financial analytics without undermining market integrity or investor protection [234,235].

Together, these financial deployment risks illustrate how probabilistic language models can influence not only individual decision processes but also broader market dynamics, reinforcing the need for domain-specific governance frameworks and robust institutional oversight. Taken together, existing research indicates that the benefits of generative AI in financial analytics are inseparable from new forms of model risk, governance complexity, and potential systemic effects, underscoring the importance of interdisciplinary oversight frameworks that combine financial regulation, AI safety mechanisms, and institutional risk management practices [234–236].

The domain-specific risks discussed above emerge from the interaction between generation and the institutional contexts in which LLMs are deployed. Figure 8 contrasts how these mechanisms manifest differently across educational and financial application environments.

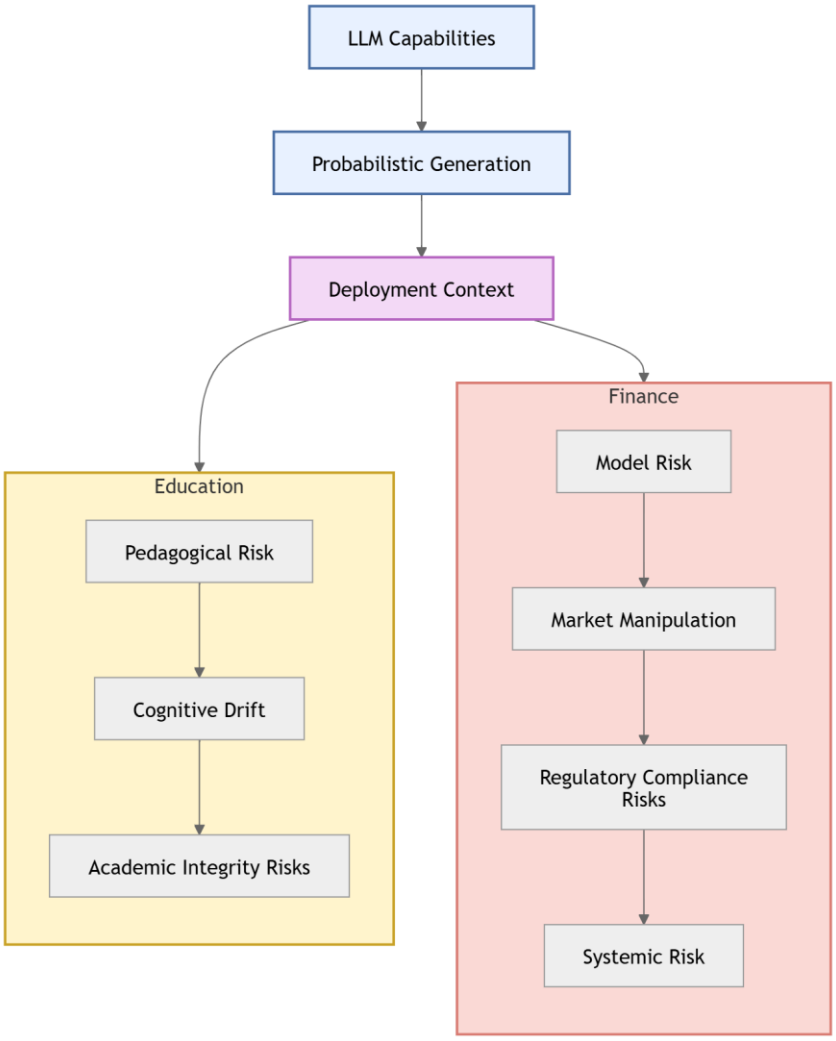


**Figure 8.** Conceptual model illustrating domain-specific risk pathways in LLM deployment across educational and financial systems. The diagram compares how similar technical mechanisms produce different institutional and societal consequences depending on the deployment context.

## 8. Regulatory Frameworks and Compliance

The rapid diffusion of LLMs has created a structural regulatory lag between generative capability and established legal oversight mechanisms [3,216]. Unlike narrow AI systems designed for predefined tasks, LLMs function as General-Purpose AI Systems (GPAIS), capable of deployment across heterogeneous domains and value chains [7,216]. This adaptability complicates attribution of responsibility among developers, model providers, system integrators, and end users. Governance discourse therefore increasingly centers on accountability, transparency, liability allocation, and risk-proportionate regulatory design [3,237].

Enterprise governance structures have begun incorporating dedicated AI oversight committees, model approval workflows, and internal risk-classification protocols to manage deployment exposure [238]. At the supranational level, the EU AI Act represents the most comprehensive attempt to formalize obligations for foundation model and high-risk system deployment [3,216]. However, the probabilistic and context-sensitive behavior of

LLM systems challenges traditional regulatory expectations grounded in determinism and predictable functionality [3,7]. Regulatory supervision must evaluate observable behavior, risk profiles, and deployment context rather than internal algorithmic determinism [3,216].

Modern AI regulation increasingly adopts risk-tiered classification models, distinguishing between prohibited practices, high-risk applications, limited-risk systems, and minimal-risk uses [3,216]. Applications in healthcare, finance, and legal decision-making are typically categorized as high risk due to potential impacts on fundamental rights and safety [3,216]. Foundation models are subject to enhanced documentation requirements, including disclosure of training data characteristics, summaries of copyrighted material usage, and communication of operational capabilities and limitations [7,216]. High-risk deployments additionally require structured human oversight mechanisms to mitigate potential rights violations [3].

Nevertheless, definitions of GPAIS remain in flux, reflecting the difficulty of applying static legal categories to adaptive systems whose behavior evolves through prompting, fine-tuning, and downstream integration [216]. This regulatory tension illustrates a broader challenge identified throughout this survey: governance frameworks must accommodate probabilistic generative behavior whose risk profile is interaction-dependent rather than fixed [3].

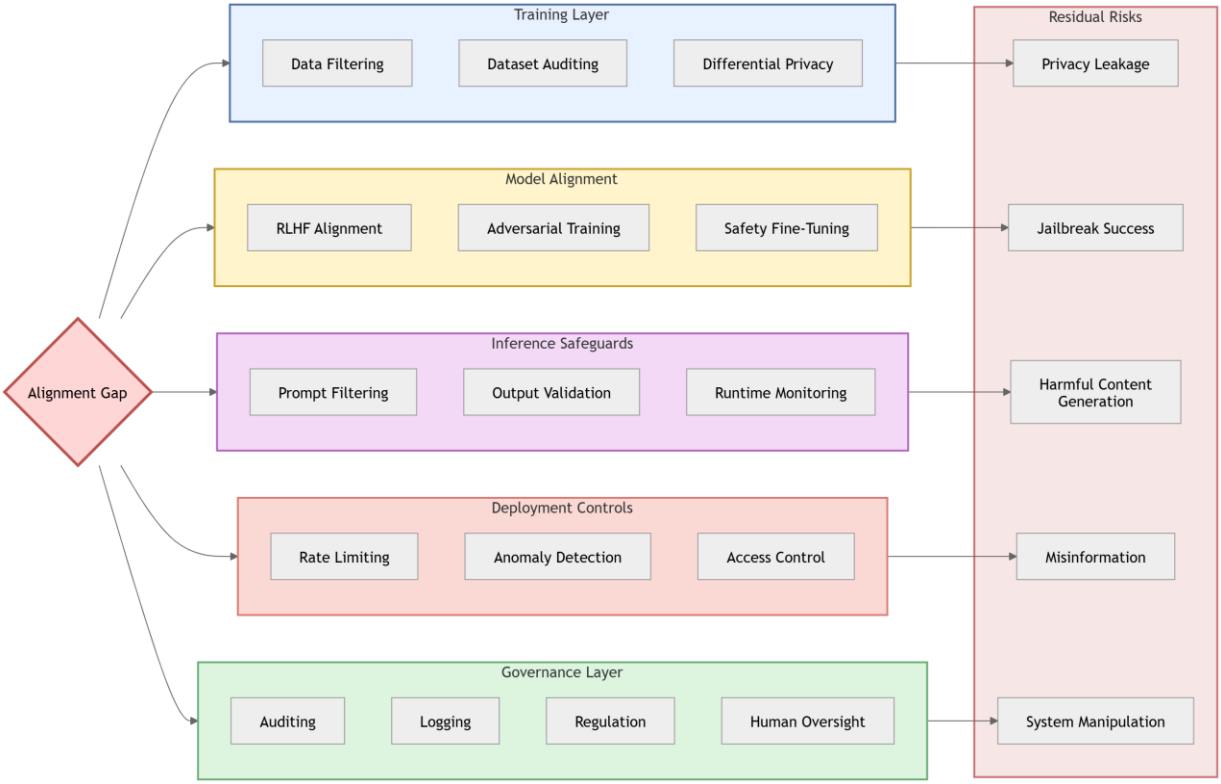


**Figure 9.** Defense-in-depth mitigation layers for reducing risks associated with the alignment gap in large language models. The diagram illustrates how complementary safeguards at the training, inference, and governance levels interact to constrain residual behavioral risks.

*8.1. EU AI Act and GPAIS Classification*

Regulatory analysis identifies general-purpose AI (GPAI) models as uniquely complex because they are not confined to a single application domain but are designed for reuse across multiple downstream contexts [3,216]. Under the EU AI Act's risk-based framework, AI systems are classified according to specific use cases defined in Article 6 and Annex III, rather than entire sectors. High-risk categories include, inter alia, employment and worker management, access to essential public and private services (such as creditworthiness assessment), law

enforcement, migration and border control management, and the administration of justice and democratic processes [239].

For foundation models and GPAI systems, compliance obligations extend beyond application-level evaluation. Providers must maintain detailed technical documentation, describe training data sources at an aggregated level, disclose summaries of copyrighted content usage, and communicate known limitations and risk mitigation measures [7,216]. GPAI models presenting systemic risk are subject to additional requirements, including model evaluation, adversarial testing, incident reporting, and cybersecurity safeguards [239]. These requirements aim to increase transparency across the LLM treing-deploying lifecycle while preserving trade secrets and intellectual property.

However, static (pre)classification struggles to capture the dynamic behavior of LLMs when integrated into downstream systems. A model that is compliant at release may generate high-risk outcomes when deployed in high-risk use cases or under adversarial prompting conditions [3]. Consequently, regulatory interpretation must incorporate lifecycle monitoring and post-deployment oversight rather than relying exclusively on pre-deployment conformity assessment [216].

### *8.2. Liability and Accountability*

Assigning legal responsibility for LLM-generated outputs remains a central governance challenge. Because outputs arise from probabilistic inference rather than deterministic programming, harmful results may emerge without explicit developer intent [3,7]. This complicates traditional liability doctrines that assume direct causal control over system behavior.

Responsibility is typically distributed across multiple actors, including model developers, API providers, integrators, and end users [3,7]. In cases involving hallucinated information, discriminatory outputs, or security incidents, identifying the locus of negligence becomes legally and practically complex. Alignment procedures and safety testing, while necessary, cannot guarantee absence of harmful outputs in all interaction contexts [3].

This dynamic creates an accountability gap in which technically compliant systems may still produce adverse outcomes. Regulatory discussions therefore emphasize the need for auditable deployment practices, traceability mechanisms, and clearly defined responsibility allocation across contractual and operational boundaries [3,7]. Legal compliance must therefore extend beyond model-level safeguards to encompass organizational governance structures and continuous oversight.

### *8.3. Organizational Practice: Multi-Tier Auditing*

Operational compliance increasingly adopts a multi-tier auditing architecture aligned with the LLM lifecycle [3,7].

Governance audits evaluate organizational structures, data-collection policies, ethical review processes, and internal accountability mechanisms [3].

Model audits assess robustness, factual reliability, bias exposure, and resistance to adversarial manipulation prior to deployment [3,7].

Application audits examine real-world societal impact, contextual bias risks, and compliance with user rights within specific deployment environments [3].

End-to-end lifecycle documentation and responsibility mapping are particularly critical in cross-border deployments subject to multiple legal regimes [3,7]. Such layered oversight functions as a system of institutional checks and balances, complementing technical defense-in-depth mechanisms described earlier in this survey [3].

This auditing structure reflects the recognition that regulatory compliance cannot be reduced to a one-time technical evaluation. Instead, it requires ongoing monitoring proportional to deployment risk.

### *8.4. Shadow AI and Invisible Deployment*

A growing compliance challenge arises from the emergence of Shadow AI, in which employees utilize publicly accessible LLM services to process organizational data without formal approval or oversight [225,232]. This phenomenon parallels the broader concept of Shadow IT, where unsanctioned technologies bypass

institutional governance mechanisms [240]. A related and increasingly documented pattern is "Bring Your Own AI" (BYOAI), whereby employees independently integrate external generative AI tools into workplace workflows without formal authorization, data protection assessment, or contractual safeguards. Such practices may expose sensitive organizational data to third-party providers, create compliance blind spots, and undermine internal governance controls, even when the model itself is technically compliant [241].

In this context, risk emerges not from adversarial exploitation but from uncontrolled adoption. Enterprise analyses indicate that AI usage frequently precedes governance maturity, exposing organizations to operational and regulatory vulnerabilities before risk management processes are established [242]. Unmonitored submission of proprietary or personal data to external LLM platforms increases the likelihood of unintended memorization and potential downstream disclosure [35,51]. Such exposure has been identified as a significant enterprise risk in generative AI deployment [16].

Shadow AI expands the socio-technical threat surface identified throughout this survey [1,2]. Technical properties such as probabilistic memorization and extraction risk [16,56] interact with human trust in fluent outputs, amplifying verification drift and reducing critical evaluation [59,136]. Without auditing procedures, usage governance, and access controls, Shadow AI may undermine compliance with regulatory frameworks such as the EU AI Act [7,86,216]. Thus, compliance risk may arise from ordinary workflow practices rather than malicious activity, underscoring the importance of internal policy enforcement and employee training.

### *8.5. Continuous Compliance and Adaptive Governance*

Regulatory frameworks are essential for responsible deployment of LLMs, yet static compliance models are insufficient in dynamic AI environments [216]. Transitioning from reactive enforcement toward safety-by-design architectures—incorporating auditable pipelines, transparent data provenance, and continuous monitoring—can improve systemic trustworthiness [3,7].

In practice, risk-tiered regulation is operationalized through continuous documentation, periodic audits, and ongoing monitoring obligations rather than one-time certification events. Organizations must maintain technical testing procedures, record data provenance decisions, and demonstrate oversight proportional to system risk. Industry analyses similarly emphasize governance maturity models as prerequisites for safe enterprise AI deployment [243].

A persistent structural difficulty lies in shared responsibility across multiple actors—developers, providers, integrators, and users—where harmful outcomes may emerge without a single clearly negligent party. This complexity is amplified by the training principles and multi-stage interaction chains across LLM training to deployment lifecycle, rendering static compliance verification inadequate.

Effective LLM governance therefore requires integration of legal oversight, technical evaluation, lifecycle monitoring, and organizational accountability mechanisms. Even technically compliant systems may exhibit emergent behavior under novel prompting conditions, indicating that regulatory compliance must be adaptive rather than static. Operational standards for continuous auditing of adaptive generative systems remain insufficiently defined and represent a critical area for future regulatory and interdisciplinary development [3,7,216].

Compliance with regulatory requirements, such as those defined in the EU AI Act [239], is tested through benchmarks that simulate high-risk scenarios. DecodingTrust [34] serves as a crucial tool in this context, as it measures compliance with dimensions such as fairness and transparency, which are necessary for documentation under new regulations [237].

## 9. Discussion: Research Gaps and Future Directions

Research on LLM security, privacy, and ethics has expanded rapidly, yet remains conceptually fragmented and methodologically inconsistent across domains [35,116]. Comprehensive surveys increasingly emphasize the absence of standardized methodologies for evaluating LLM trustworthiness in a unified and comparable manner [103]. While individual threat categories—such as prompt injection, privacy leakage, hallucination, or adversarial robustness—have received substantial attention, evaluation frameworks capable of measuring multiple safety dimensions simultaneously remain limited [35,116].

Existing benchmarks frequently assess isolated properties rather than interactions among reasoning capability, safety constraints, robustness, and utility. This fragmentation complicates cross-model comparison and obscures trade-offs between performance and security in LLM models [116]. Advancing trustworthy deployment therefore requires a transition from reactive mitigation strategies toward safety-by-design methodologies that embed continuous validation across the entire LLM lifecycle [34,215]. Whereas prior surveys often isolate privacy, ethics, or adversarial robustness as separate research strands, this work integrates lifecycle modeling, alignment gap analysis, and governance structures into a unified conceptual framework for evaluating systemic risk.


**Figure 10.** Evaluation and governance pipeline for identifying and mitigating alignment gaps in large language models. The diagram outlines the iterative process of risk identification, evaluation, mitigation, and governance oversight used to manage alignment-related vulnerabilities.9.1. Fragmented Evaluation and the Alignment Gap

A primary structural gap in the current research landscape is the absence of unified safety evaluation enabling direct comparison across LLM architectures and deployment contexts [35,116]. Many evaluation methodologies focus narrowly on toxicity, hallucination, or bias without capturing interactions between reasoning capability, reliability, adversarial robustness, and utility [116]. As a result, improvements along one dimension may inadvertently degrade another.

A persistent challenge concerns incomplete behavioral control following post-training alignment procedures [34]. Even models aligned using RLHF remain vulnerable to adversarial prompting and out-of-distribution inputs [34]. Latent jailbreak studies further demonstrate that harmful behaviors can be triggered through subtle semantic steering, indicating that alignment does not eliminate unsafe internal representations but instead constrains their surface expression [215]. This alignment gap—defined as the divergence between intended behavioral constraints and latent generative capacity—remains insufficiently quantified and methodologically underexplored.

### *9.2. Toward Holistic Benchmarking*

Future research increasingly emphasizes comprehensive evaluation frameworks capable of measuring multiple trustworthiness dimensions simultaneously [34,35]. HELM represents a foundational effort to standardize cross-domain benchmarking and enable broader risk assessment beyond single-task performance metrics [35]. Similarly, DecodingTrust proposes a multi-dimensional evaluation framework assessing robustness, privacy, fairness, toxicity, and factual reliability within a unified experimental structure [34].

Importantly, DecodingTrust demonstrates that gains in one trustworthiness dimension do not necessarily generalize across others. In some cases, increased reasoning capability may enhance exploitability by enabling more coherent adversarial strategies [34]. These findings suggest that trustworthiness must be evaluated as an interacting system of properties rather than as independent safety metrics.

However, current holistic benchmarks remain primarily diagnostic rather than preventive. They identify failure modes but rarely provide mechanisms for continuous validation in live deployment environments. Bridging evaluation and operational monitoring therefore represents a critical research direction.

### *9.3. Automated Vulnerability Discovery and Continuous Validation*

Manual red-teaming remains resource-intensive and difficult to scale, motivating the development of automated vulnerability discovery mechanisms [34,244]. Fuzzing-based frameworks such as FuzzLLM automatically generate and mutate adversarial inputs to expose structural weaknesses and stress-test deployed safeguards [244]. These approaches shift security from periodic auditing toward continuous adversarial validation.

Integrating automated red-teaming directly into deployment pipelines is increasingly recognized as a practical requirement for operational security [34,244]. Exploitability analysis and curated high-quality instruction datasets further define future alignment research priorities by enabling systematic exposure of latent failure modes [245,246]. Such mechanisms complement defense-in-depth architectures by proactively identifying weaknesses rather than relying solely on reactive mitigation.

Nevertheless, automated discovery frameworks remain limited in modeling complex multi-turn interaction chains and long-horizon reasoning manipulation. Extending fuzzing methodologies to conversational and AI agent-based contexts constitutes an open research challenge.

### *9.4. Interpretability, Causal Tracing, and Internal Representations*

Understanding persistent failure modes requires deeper insight into internal representations of Transformer-based architectures [34,116,215]. Current alignment techniques primarily operate at the behavioral level, constraining outputs without fully characterizing internal activation dynamics associated with unsafe responses.

Research on neural activation analysis increasingly investigates representation clusters correlated with protection bypass, hallucination, and adversarial steering [34,244]. Latent jailbreak phenomena demonstrate that harmful behaviors may remain embedded within the model and can be activated through subtle contextual modulation [34,215]. This indicates that surface-level filtering and reinforcement learning constraints may not fully eliminate unsafe generative pathways.

Advancing interpretable alignment therefore requires causal tracing methodologies capable of identifying and modifying internal representations associated with harmful behavior. Without such mechanisms, alignment remains probabilistic rather than verifiable.

### *9.5. Structured Research Priorities*

Based on the reviewed literature, future research directions can be structured into four complementary categories:

1. **Unified evaluation frameworks —** Development of standardized, multi-dimensional benchmarks capable of jointly measuring safety, reasoning capability, robustness, and utility across heterogeneous deployment contexts [35,103,116].
2. **Automated red-teaming and fuzzing —** Scalable mechanisms that continuously discover vulnerabilities during deployment, integrating exploitability analysis directly into operational pipelines [34,244–246].
3. **Interpretability and causal alignment —** Techniques that identify internal representations responsible for unsafe behavior and enable targeted modification rather than relying solely on surface-level output constraints [34,116,215].
4. **Verifiable guarantees—** Formal approaches providing measurable privacy, robustness, and safety assurances instead of relying exclusively on probabilistic behavioral expectations [34,35,215].

Collectively, these priorities underscore that trustworthy LLM deployment requires integration of evaluation methodologies, adversarial validation, interpretability research, and governance mechanisms rather than treating them as isolated research domains.

As model capability increases, improvements in usefulness may simultaneously expand exploitability by enhancing coherence, contextual reasoning, and adaptive response generation. This duality reinforces the need for continuous validation mechanisms embedded throughout the lifecycle. At present, no unified methodology provides verifiable end-to-end safety guarantees for adaptive generative systems, highlighting the necessity of coordinated technical, institutional, and regulatory development [247].

Although watermarking and provenance attribution mechanisms represent promising approaches for identifying AI-generated content, their effectiveness remains limited under paraphrasing, adversarial rewriting, and cross-model generation pipelines. Consequently, these mechanisms are treated in this survey as complementary mitigation techniques rather than as standalone future research priorities [117,119,128].

## 10. Limitations and Future Research Directions

Recent developments in agentic LLM architectures and multimodal foundation models introduce additional security challenges that extend beyond text-only interaction. Agentic systems capable of autonomous tool use, external API calls, or multi-step reasoning workflows may expose new vulnerabilities through indirect prompt injection, tool manipulation, and unsafe autonomous action execution.,

In addition, multimodal LLMs that process images, audio, or video inputs extend the at-tack surface by enabling prompt injection and adversarial manipulation through non-text modalities. Such multimodal prompt injection mechanisms remain comparatively un-derexplored in the current jailbreak and adversarial attack literature, which has primarily focused on text-based interaction.

Systematic security evaluation of these emerging architectures remains an open research direction and represents an important direction for future work.

## 11. Conclusion

This survey synthesizes the rapidly expanding body of research on LLM security, privacy, ethics, and governance into a unified lifecycle-oriented framework. By integrating interaction-layer threats, adversarial optimization techniques, training-stage compromise, privacy leakage, and regulatory accountability within a defense-in-depth perspective, the analysis demonstrates that LLM risk is not reducible to isolated implementation flaws. Rather, it emerges as a systemic property of probabilistic generative architectures interacting with complex socio-technical environments.

The findings highlight a persistent alignment gap between intended behavioral constraints and latent generative capacity. While post-training alignment, runtime safeguards, privacy-preserving optimization, and regulatory oversight reduce observable risk, none fully eliminate vulnerabilities inherent to adaptive probabilistic modeling. Security and trustworthiness therefore cannot be achieved through single-layer mitigation strategies but require coordinated protections distributed across the entire model lifecycle.

Advancing reliable deployment demands continuous automated red-teaming, unified multi-dimensional evaluation methodologies, interpretability-driven alignment techniques, and progress toward formally verifiable privacy and robustness guarantees. Equally important is the maturation of governance mechanisms capable of adapting to evolving threat models and deployment contexts.

Ultimately, trustworthy LLM integration depends on the convergence of technical robustness, privacy-aware learning, institutional accountability, and continuous lifecycle auditing. As model capability continues to expand, sustainable deployment will require not only stronger safeguards but also adaptive governance architectures capable of managing emergent behavior in generative systems.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| BYOAI | Bring Your Own AI |
| C4 | Colossal Clean Crawled Corpus |
| CSRF | Cross Site Request Forgery |
| DAN | Do Anything Now |
| DoLa | Decoding by Contrasting Layers |
| DP | Differential Privacy |
| DPO | Direct Preference Optimization |
| DP-SGD | Differentially Private Stochastic Gradient Descent |
| EU AI Act | European Union Artificial Intelligence Act |
| FL | Federated Learning |
| GCG | Greedy Coordinate Gradient |
| GDPR | General Data Protection Regulation |
| GenAI | Generative Artificial Intelligence |
| GPAI | General-Purpose AI |
| GPAIS | General-Purpose AI Systems |
| GPT | Generative Pre-trained Transformer |
| HELM | Holistic Evaluation of Language Models |
| LLM | Large Language Model |
| NIST | National Institute of Standards and Technology |
| OWASP | Open Web Application Security Project |
| P3O | Pairwise Proximal Policy Optimization |
| PAIR | Prompt Automatic Iterative Refinement |
| PII | Personally Identifiable Information |
| RAG | Retrieval-Augmented Generation |
| RLHF | Reinforcement Learning from Human Feedback |
| SSRF | Server-Side Request Forgery |
| XSS | Cross Site Scripting |
| XXE | XML External Entity |

## Appendix A

The appendix provides the original OWASP Top-10 rankings used to construct the evolution matrix presented in the main text. Table A1 summarizes the historical OWASP Top-10 lists from 2003 to 2025 and serves as the source data for the comparative analysis of application security risk trends discussed in the paper. The appendix is included to ensure transparency of the underlying ranking data while preserving the readability of the main analysis.

**Table A.** Historical OWASP Top-10 rankings (2003–2025) [20–27].The table provides a longitudinal overview of the evolution of dominant application security threats and highlights the transition toward risks relevant to AI-enabled systems.

| Rank | 2003 | 2004 | 2007 | 2010 | 2013 | 2017 | 2021 | 2025 |
|---|---|---|---|---|---|---|---|---|
| 1 | Unvalidated Input | Unvalidated Input | Cross-Site Scripting | Injection | Injection | Injection | Broken Access Control | Broken Access Control |
| 2 | Broken Access Control | Broken Access Control | Injection Flaws | Broken Authentication | Broken Auth & Session | Broken Authentication | Cryptographic Failures | Security Misconfiguration |
| 3 | Broken Authentication | Broken Authentication | Malicious File Execution | Sensitive Data Exposure | Sensitive Data Exposure | Sensitive Data Exposure | Injection | Software Supply Chain Failures |
| 4 | Cross-Site Scripting | Cross-Site Scripting | Insecure Direct Object Reference | Insecure Direct Object Reference | XML External Entities | XML External Entities | Insecure Design | Cryptographic Failures |
| 5 | Buffer Overflow | Buffer Overflow | Cross-Site Request Forgery | Security Misconfiguration | Broken Access Control | Broken Access Control | Security Misconfiguration | Injection |
| 6 | Injection Flaws | Injection Flaws | Information Leakage | Sensitive Data Exposure | Security Misconfiguration | Security Misconfiguration | Vulnerable & Outdated Components | Insecure Design |
| 7 | Improper Error Handling | Improper Error Handling | Broken Authentication | Missing Function Level Access Control | Missing Function Level Access Control | Cross-Site Scripting | Identification & Authentication Failures | Authentication Failures |
| 8 | Insecure Storage | Insecure Storage | Insecure Cryptographic Storage | Cross-Site Request Forgery | Cross-Site Request Forgery | Insecure Deserialization | Software & Data Integrity Failures | Software or Data Integrity Failures |
| 9 | Denial of Service | Denial of Service | Insecure Communications | Using Known Vulnerable Components | Using Known Vulnerable Components | Using Components with Known Vulnerabilities | Security Logging & Monitoring Failures | Security Logging and Alerting Failures |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 10 | Insecure Configuration | Insecure Configuration | Failure to Restrict URL Access | Unvalidated Redirects & Forwards | Unvalidated Redirects & Forwards | Insufficient Logging & Monitoring | Server-Side Request Forgery | Mishandling of Exceptional Conditions |

## References


1. Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; DasSarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; et al. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback 2022, doi:10.48550/arXiv.2204.05862.
2. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding 2019, doi:10.48550/arXiv.1810.04805.
3. Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. Training Language Models to Follow Instructions with Human Feedback 2022, doi:10.48550/arXiv.2203.02155.
4. Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. LLaMA: Open and Efficient Foundation Language Models 2023, doi:10.48550/arXiv.2302.13971.
5. Han, S.; Wang, M.; Zhang, J.; Li, D.; Duan, J. A Review of Large Language Models: Fundamental Architectures, Key Technological Evolutions, Interdisciplinary Technologies Integration, Optimization and Compression Techniques, Applications, and Challenges. *Electronics* 2024, *13*, 5040, doi:10.3390/electronics13245040.
6. Cheng, H.-W. Challenges and Limitations of ChatGPT and Artificial Intelligence for Scientific Research: A Perspective from Organic Materials. *AI* 2023, *4*, 401–405, doi:10.3390/ai4020021.
7. Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. Constitutional AI: Harmlessness from AI Feedback 2022, doi:10.48550/arXiv.2212.08073.
8. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need 2023, doi:10.48550/arXiv.1706.03762.
9. OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; et al. GPT-4 Technical Report 2024, doi:10.48550/arXiv.2303.08774.
10. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models Are Unsupervised Multitask Learners. OpenAI Technical Report, 2019. Available online: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (accessed on 5 March 2026).
11. Sajjadi Mohammadabadi, S.M.; Kara, B.C.; Eyupoglu, C.; Uzay, C.; Tosun, M.S.; Karakuş, O. A Survey of Large Language Models: Evolution, Architectures, Adaptation, Benchmarking, Applications, Challenges, and Societal Implications. *Electronics* 2025, *14*, 3580, doi:10.3390/electronics14183580.
12. Hulsen, T. Explainable Artificial Intelligence (XAI): Concepts and Challenges in Healthcare. *AI* 2023, *4*, 652–666, doi:10.3390/ai4030034.
13. Bommasani, R.; Hudson, D.A.; Adeli, E.; Altman, R.; Arora, S.; Arx, S. von; Bernstein, M.S.; Bohg, J.; Bosselut, A.; Brunskill, E.; et al. On the Opportunities and Risks of Foundation Models 2022, doi:10.48550/arXiv.2108.07258.

14. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models Are Few-Shot Learners 2020, doi:10.48550/arXiv.2005.14165.
15. European Union Agency for Cybersecurity (ENISA). ENISA Threat Landscape 2024: July 2023 to June 2024. Publications Office of the European Union: Luxembourg, 2024. Available online: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024 (accessed on 5 March 2026).
16. Weidinger, L.; Uesato, J.; Rauh, M.; Griffin, C.; Huang, P.-S.; Mellor, J.; Glaese, A.; Cheng, M.; Balle, B.; Kasirzadeh, A.; et al. Taxonomy of Risks Posed by Language Models. In Proceedings of the 2022 ACM Conference on Fairness Accountability and Transparency; ACM: Seoul Republic of Korea, June 21 2022; pp. 214–229, doi:10.1145/3531146.3533088.
17. Albaroudi, E.; Mansouri, T.; Alameer, A. A Comprehensive Review of AI Techniques for Addressing Algorithmic Bias in Job Hiring. *AI* 2024, *5*, 383–404, doi:10.3390/ai5010019.
18. Chung, H.W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, Y.; Wang, X.; Dehghani, M.; Brahma, S.; et al. Scaling Instruction-Finetuned Language Models 2022, doi:10.48550/arXiv.2210.11416.
19. Weidinger, L.; Mellor, J.; Rauh, M.; Griffin, C.; Uesato, J.; Huang, P.-S.; Cheng, M.; Glaese, M.; Balle, B.; Kasirzadeh, A.; et al. Ethical and Social Risks of Harm from Language Models 2021, doi:10.48550/arXiv.2112.04359.
20. Top10/2003/OWASPWebApplicationSecurityTopTen-Version1.Pdf at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2003/OWASPWebApplicationSecurityTopTen-Version1.pdf (accessed on 5 March 2026).
21. Top10/2004/OWASP_Top_Ten_2004.Pdf at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2004/OWASP_Top_Ten_2004.pdf (accessed on 5 March 2026).
22. Top10/2007/OWASP Top 10 2007.Pdf at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2007/OWASP%20Top%2010%202007.pdf (accessed on 5 March 2026).
23. Top10/2010/OWASP Top 10 - 2010 English.Pdf at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2010/OWASP%20Top%2010%20-%202010%20English.pdf (accessed on 5 March 2026).
24. Top10/2013/OWASP Top 10 - 2013 English FInal.Pptx at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2013/OWASP%20Top%2010%20-%202013%20English%20FInal.pptx (accessed on 5 March 2026).
25. Top10/2017/OWASP Top 10-2017 (En).Pdf at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2017/OWASP%20Top%2010-2017%20(en).pdf (accessed on 5 March 2026).
26. Top10/2021/Data/OWASP Top 10 2020 Data Analysis Plan.Docx at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2021/Data/OWASP%20Top%2010%202020%20Data%20Analysis%20Plan.docx (accessed on 5 March 2026).
27. Top10/2025/Docs/En/0x00_2025-Introduction.Md at Master · OWASP/Top10 · GitHub Available online: https://github.com/OWASP/Top10/blob/master/2025/docs/en/0x00_2025-Introduction.md (accessed on 5 March 2026).
28. OWASP Top10/2021-2003_Comparison/OWASP_Top_Ten_-_Comparison_of_2003,2004,2007,2010,2013,2017_and_2021_Releases.Pdf at Master · OWASP/Top10 Available online:

https://github.com/OWASP/Top10/blob/master/2021-2003_Comparison/OWASP_Top_Ten_-_Comparison_of_2003%2C2004%2C2007%2C2010%2C2013%2C2017_and_2021_Releases.pdf (accessed on 7 March 2026).

29. Bender, E.M.; Gebru, T.; McMillan-Major, A.; Shmitchell, S. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜. In Proceedings of the Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency; ACM: Virtual Event Canada, March 3 2021; pp. 610–623, doi:10.1145/3442188.3445922.
30. Shayegani, E.; Mamun, M.A.A.; Fu, Y.; Zaree, P.; Dong, Y.; Abu-Ghazaleh, N. Survey of Vulnerabilities in Large Language Models Revealed by Adversarial Attacks 2023, doi:10.48550/arXiv.2310.10844.
31. Liu, B.; Xiao, B.; Jiang, X.; Cen, S.; He, X.; Dou, W. Adversarial Attacks on Large Language Model-Based System and Mitigating Strategies: A Case Study on ChatGPT. *Security and Communication Networks* 2023, *2023*, 1–10, doi:10.1155/2023/8691095.
32. Carlini, N.; Tramer, F.; Wallace, E.; Jagielski, M.; Herbert-Voss, A.; Lee, K.; Roberts, A.; Brown, T.; Song, D.; Erlingsson, U.; et al. Extracting Training Data from Large Language Models 2021, doi:10.48550/arXiv.2012.07805.
33. Mireshghallah, F.; Goyal, K.; Uniyal, A.; Berg-Kirkpatrick, T.; Shokri, R. Quantifying Privacy Risks of Masked Language Models Using Membership Inference Attacks. In Proceedings of the Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Abu Dhabi, United Arab Emirates, 2022; pp. 8332–8347, doi:10.18653/v1/2022.emnlp-main.570.
34. Wang, B.; Chen, W.; Pei, H.; Xie, C.; Kang, M.; Zhang, C.; Xu, C.; Xiong, Z.; Dutta, R.; Schaeffer, R.; et al. DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models 2024, doi:10.48550/arXiv.2306.11698.
35. Liang, P.; Bommasani, R.; Lee, T.; Tsipras, D.; Soylu, D.; Yasunaga, M.; Zhang, Y.; Narayanan, D.; Wu, Y.; Kumar, A.; et al. Holistic Evaluation of Language Models 2023, doi:10.48550/arXiv.2211.09110.
36. Mattern, J.; Mireshghallah, F.; Jin, Z.; Schölkopf, B.; Sachan, M.; Berg-Kirkpatrick, T. Membership Inference Attacks against Language Models via Neighbourhood Comparison 2023, doi:10.48550/arXiv.2305.18462.
37. Mienye, I.D.; Swart, T.G. Ensemble Large Language Models: A Survey. *Information* 2025, *16*, 688, doi:10.3390/info16080688.
38. Ataman, D.; Birch, A.; Habash, N.; Federico, M.; Koehn, P.; Cho, K. Machine Translation in the Era of Large Language Models:A Survey of Historical and Emerging Problems. *Information* 2025, *16*, 723, doi:10.3390/info16090723.
39. Askell, A.; Bai, Y.; Chen, A.; Drain, D.; Ganguli, D.; Henighan, T.; Jones, A.; Joseph, N.; Mann, B.; DasSarma, N.; et al. A General Language Assistant as a Laboratory for Alignment 2021, doi:10.48550/arXiv.2112.00861.
40. Dodge, J.; Sap, M.; Marasović, A.; Agnew, W.; Ilharco, G.; Groeneveld, D.; Mitchell, M.; Gardner, M. Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus 2021, doi:10.48550/arXiv.2104.08758.
41. Balkir, E.; Kiritchenko, S.; Nejadgholi, I.; Fraser, K. Challenges in Applying Explainability Methods to Improve the Fairness of NLP Models. In Proceedings of the Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022); Association for Computational Linguistics: Seattle, U.S.A., 2022; pp. 80–92, doi:10.18653/v1/2022.trustnlp-1.8.

42. Cho, I.; Wesslen, R.; Karduni, A.; Santhanam, S.; Shaikh, S.; Dou, W. The Anchoring Effect in Decision-Making with Visual Analytics. In Proceedings of the 2017 IEEE Conference on Visual Analytics Science and Technology (VAST); IEEE: Phoenix, AZ, USA, October 2017; pp. 116–126, doi:10.1109/VAST.2017.8585665.
43. Elaraby, M.; Lu, M.; Dunn, J.; Zhang, X.; Wang, Y.; Liu, S.; Tian, P.; Wang, Y.; Wang, Y. Halo: Estimation and Reduction of Hallucinations in Open-Source Weak Large Language Models 2023, doi:10.48550/arXiv.2308.11764.
44. He, X.; Lyu, L.; Chen, C.; Xu, Q. Extracted BERT Model Leaks More Information than You Think! In Proceedings of the Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Abu Dhabi, United Arab Emirates, 2022; pp. 1530–1537, doi:10.18653/v1/2022.emnlp-main.99.
45. Christiano, P.; Leike, J.; Brown, T.B.; Martic, M.; Legg, S.; Amodei, D. Deep Reinforcement Learning from Human Preferences 2023, doi:10.48550/arXiv.1706.03741.
46. Understanding the MITRE ATT&CK Framework: A Guide for Security Teams | Fidelis Security Available online: https://fidelissecurity.com/cybersecurity-101/learn/mitre-attack-framework/ (accessed on 5 March 2026).
47. Zero Trust Maturity Model | CISA Available online: https://www.cisa.gov/zero-trust-maturity-model (accessed on 5 March 2026).
48. Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H.B.; Mironov, I.; Talwar, K.; Zhang, L. Deep Learning with Differential Privacy. In Proceedings of the Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security; October 24 2016; pp. 308–318, doi:10.1145/2976749.2978318.
49. Borgeaud, S.; Mensch, A.; Hoffmann, J.; Cai, T.; Rutherford, E.; Millican, K.; Driessche, G. van den; Lespiau, J.-B.; Damoc, B.; Clark, A.; et al. Improving Language Models by Retrieving from Trillions of Tokens 2022, doi:10.48550/arXiv.2112.04426.
50. Carlini, N.; Hayes, J.; Nasr, M.; Jagielski, M.; Sehwag, V.; Tramèr, F.; Balle, B.; Ippolito, D.; Wallace, E. Extracting Training Data from Diffusion Models 2023, doi:10.48550/arXiv.2301.13188.
51. Mozes, M.; He, X.; Kleinberg, B.; Griffin, L.D. Use of LLMs for Illicit Purposes: Threats, Prevention Measures, and Vulnerabilities 2023, doi:10.48550/arXiv.2308.12833.
52. Aghakhani, H.; Dai, W.; Manoel, A.; Fernandes, X.; Kharkar, A.; Kruegel, C.; Vigna, G.; Evans, D.; Zorn, B.; Sim, R. TrojanPuzzle: Covertly Poisoning Code-Suggestion Models 2024, doi:10.48550/arXiv.2301.02344.
53. Gu, T.; Liu, K.; Dolan-Gavitt, B.; Garg, S. BadNets: Evaluating Backdooring Attacks on Deep Neural Networks. *IEEE Access* 2019, *7*, 47230–47244, doi:10.1109/ACCESS.2019.2909068.
54. Li, Z.; Zhang, S.; Zhao, H.; Yang, Y.; Yang, D. BatGPT: A Bidirectional Autoregessive Talker from Generative Pre-Trained Transformer 2023, doi:10.48550/arXiv.2307.00360.
55. Motoki, F.; Pinho Neto, V.; Rodrigues, V. More Human than Human: Measuring ChatGPT Political Bias. *Public Choice* 2024, *198*, 3–23, doi:10.1007/s11127-023-01097-2.
56. Nasr, M.; Carlini, N.; Hayase, J.; Jagielski, M.; Cooper, A.F.; Ippolito, D.; Choquette-Choo, C.A.; Wallace, E.; Tramèr, F.; Lee, K. Scalable Extraction of Training Data from (Production) Language Models 2023, doi:10.48550/arXiv.2311.17035.
57. Crothers, E.N.; Japkowicz, N.; Viktor, H.L. Machine-Generated Text: A Comprehensive Survey of Threat Models and Detection Methods. *IEEE Access* 2023, *11*, 70977–71002, doi:10.1109/ACCESS.2023.3294090.
58. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.* 2025, *43*, 1–55, doi:10.1145/3703155.

59. Kumar, A.; Agarwal, C.; Srinivas, S.; Li, A.J.; Feizi, S.; Lakkaraju, H. Certifying LLM Safety against Adversarial Prompting 2025, doi:10.48550/arXiv.2309.02705.
60. Liu, X.; Xu, N.; Chen, M.; Xiao, C. AutoDAN: Generating Stealthy Jailbreak Prompts on Aligned Large Language Models 2024, doi:10.48550/arXiv.2310.04451.
61. Jaffal, N.O.; Alkhanafseh, M.; Mohaisen, D. Large Language Models in Cybersecurity: A Survey of Applications, Vulnerabilities, and Defense Techniques. *AI* 2025, *6*, 216, doi:10.3390/ai6090216.
62. Salem, A.; Paverd, A.; Köpf, B. Maatphor: Automated Variant Analysis for Prompt Injection Attacks 2023, doi:10.48550/arXiv.2312.11513.
63. Phute, M.; Helbling, A.; Hull, M.; Peng, S.; Szyller, S.; Cornelius, C.; Chau, D.H. LLM Self Defense: By Self Examination, LLMs Know They Are Being Tricked 2024, doi:10.48550/arXiv.2308.07308.
64. Liu, Y.; Deng, G.; Li, Y.; Wang, K.; Wang, Z.; Wang, X.; Zhang, T.; Liu, Y.; Wang, H.; Zheng, Y.; et al. Prompt Injection Attack against LLM-Integrated Applications 2025, doi:10.48550/arXiv.2306.05499.
65. Perez, F.; Ribeiro, I. Ignore Previous Prompt: Attack Techniques For Language Models 2022, doi:10.48550/arXiv.2211.09527.
66. Greshake, K.; Abdelnabi, S.; Mishra, S.; Endres, C.; Holz, T.; Fritz, M. Not What You've Signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection 2023, doi:10.48550/arXiv.2302.12173.
67. Jiang, S.; Chen, X.; Tang, R. Prompt Packer: Deceiving LLMs through Compositional Instruction with Hidden Attacks 2023, doi:10.48550/arXiv.2310.10077.
68. Pelrine, K.; Taufeeque, M.; Zając, M.; McLean, E.; Gleave, A. Exploiting Novel GPT-4 APIs 2024, doi:10.48550/arXiv.2312.14302.
69. Zhang, C.; Jin, M.; Yu, Q.; Liu, C.; Xue, H.; Jin, X. Goal-Guided Generative Prompt Injection Attack on Large Language Models 2024, doi:10.48550/arXiv.2404.07234.
70. Shen, X.; Chen, Z.; Backes, M.; Shen, Y.; Zhang, Y. "Do Anything Now": Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models 2024, doi:10.48550/arXiv.2308.03825.
71. Liyanage, V.; Buscaldi, D. Detecting Artificially Generated Academic Text: The Importance of Mimicking Human Utilization of Large Language Models. In Natural Language Processing and Information Systems; Métais, E., Meziane, F., Sugumaran, V., Manning, W., Reiff-Marganiec, S., Eds.; Lecture Notes in Computer Science, Vol. 13913; Springer: Cham, Switzerland, 2023; pp. 558–565.
72. Chern, I.-C.; Chern, S.; Chen, S.; Yuan, W.; Feng, K.; Zhou, C.; He, J.; Neubig, G.; Liu, P. FacTool: Factuality Detection in Generative AI -- A Tool Augmented Framework for Multi-Task and Multi-Domain Scenarios 2023, doi:10.48550/arXiv.2307.13528.
73. Jin, H.; Chen, R.; Zhang, P.; Zhou, A.; Wang, H. GUARD: Role-Playing to Generate Natural-Language Jailbreakings to Test Guideline Adherence of Large Language Models 2025, doi:10.48550/arXiv.2402.03299.
74. Lapid, R.; Langberg, R.; Sipper, M. Open Sesame! Universal Black Box Jailbreaking of Large Language Models 2024, doi:10.48550/arXiv.2309.01446.
75. Song, Y.; Li, C.; Xing, W.; Lyu, B.; Zhu, W. Investigating Perceived Fairness of AI Prediction System for Math Learning: A Mixed-Methods Study with College Students. *The Internet and Higher Education* 2025, *65*, 101000, doi:10.1016/j.iheduc.2025.101000.
76. Deng, G.; Liu, Y.; Li, Y.; Wang, K.; Zhang, Y.; Li, Z.; Wang, H.; Zhang, T.; Liu, Y. MasterKey: Automated Jailbreak Across Multiple Large Language Model Chatbots. In Proceedings of the Proceedings 2024 Network and Distributed System Security Symposium; 2024, doi:10.14722/ndss.2024.24188.

77. Ding, P.; Kuang, J.; Ma, D.; Cao, X.; Xian, Y.; Chen, J.; Huang, S. A Wolf in Sheep's Clothing: Generalized Nested Jailbreak Prompts Can Fool Large Language Models Easily 2024, doi:10.48550/arXiv.2311.08268.
78. Mehrotra, A.; Zampetakis, M.; Kassianik, P.; Nelson, B.; Anderson, H.; Singer, Y.; Karbasi, A. Tree of Attacks: Jailbreaking Black-Box LLMs Automatically 2024, doi:10.48550/arXiv.2312.02119.
79. Robey, A.; Wong, E.; Hassani, H.; Pappas, G.J. SmoothLLM: Defending Large Language Models Against Jailbreaking Attacks 2024, doi:10.48550/arXiv.2310.03684.
80. Deng, B.; Wang, W.; Feng, F.; Deng, Y.; Wang, Q.; He, X. Attack Prompt Generation for Red Teaming and Defending Large Language Models 2023, doi:10.48550/arXiv.2310.12505.
81. Guo, X.; Yu, F.; Zhang, H.; Qin, L.; Hu, B. COLD-Attack: Jailbreaking LLMs with Stealthiness and Controllability 2024, doi:10.48550/arXiv.2402.08679.
82. Zhou, A.; Li, B.; Wang, H. Robust Prompt Optimization for Defending Language Models Against Jailbreaking Attacks 2024, doi:10.48550/arXiv.2401.17263.
83. Chao, P.; Robey, A.; Dobriban, E.; Hassani, H.; Pappas, G.J.; Wong, E. Jailbreaking Black Box Large Language Models in Twenty Queries 2024, doi:10.48550/arXiv.2310.08419.
84. Jain, N.; Schwarzschild, A.; Wen, Y.; Somepalli, G.; Kirchenbauer, J.; Chiang, P.; Goldblum, M.; Saha, A.; Geiping, J.; Goldstein, T. Baseline Defenses for Adversarial Attacks Against Aligned Language Models 2023, doi:10.48550/arXiv.2309.00614.
85. Cao, B.; Cao, Y.; Lin, L.; Chen, J. Defending Against Alignment-Breaking Attacks via Robustly Aligned LLM. In Proceedings of the Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Association for Computational Linguistics: Bangkok, Thailand, 2024; pp. 10542–10560, doi:10.18653/v1/2024.acl-long.568.
86. Kang, D.; Li, X.; Stoica, I.; Guestrin, C.; Zaharia, M.; Hashimoto, T. Exploiting Programmatic Behavior of LLMs: Dual-Use Through Standard Security Attacks 2023, doi:10.48550/arXiv.2302.05733.
87. Dong, X.; Tuan, L.A.; Lin, M.; Yan, S.; Zhang, H. How Should Pre-Trained Language Models Be Fine-Tuned Towards Adversarial Robustness? 2021, doi:10.48550/arXiv.2112.11668.
88. Guo, C.; Sablayrolles, A.; Jégou, H.; Kiela, D. Gradient-Based Adversarial Attacks against Text Transformers 2021, doi:10.48550/arXiv.2104.13733.
89. Mangaokar, N.; Hooda, A.; Choi, J.; Chandrashekaran, S.; Fawaz, K.; Jha, S.; Prakash, A. PRP: Propagating Universal Perturbations to Attack Large Language Model Guard-Rails 2024, doi:10.48550/arXiv.2402.15911.
90. Zou, A.; Wang, Z.; Carlini, N.; Nasr, M.; Kolter, J.Z.; Fredrikson, M. Universal and Transferable Adversarial Attacks on Aligned Language Models 2023, doi:10.48550/arXiv.2307.15043.
91. Li, Y.; Li, T.; Chen, K.; Zhang, J.; Liu, S.; Wang, W.; Zhang, T.; Liu, Y. BadEdit: Backdooring Large Language Models by Model Editing 2024, doi:10.48550/arXiv.2403.13355.
92. Zhao, S.; Jia, M.; Tuan, L.A.; Pan, F.; Wen, J. Universal Vulnerabilities in Large Language Models: Backdoor Attacks for In-Context Learning 2024, doi:10.48550/arXiv.2401.05949.
93. Cui, G.; Yuan, L.; He, B.; Chen, Y.; Liu, Z.; Sun, M. A Unified Evaluation of Textual Backdoor Learning: Frameworks and Benchmarks 2022, doi:10.48550/arXiv.2206.08514.
94. Kurita, K.; Michel, P.; Neubig, G. Weight Poisoning Attacks on Pre-Trained Models 2020, doi:10.48550/arXiv.2004.06660.
95. Ganguli, D.; Lovitt, L.; Kernion, J.; Askell, A.; Bai, Y.; Kadavath, S.; Mann, B.; Perez, E.; Schiefer, N.; Ndousse, K.; et al. Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned 2022, doi:10.48550/arXiv.2209.07858.

96. Geiping, J.; Fowl, L.; Somepalli, G.; Goldblum, M.; Moeller, M.; Goldstein, T. What Doesn't Kill You Makes You Robust(Er): How to Adversarially Train against Data Poisoning 2022, doi:10.48550/arXiv.2102.13624.
97. Qi, X.; Zeng, Y.; Xie, T.; Chen, P.-Y.; Jia, R.; Mittal, P.; Henderson, P. Fine-Tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! 2023, doi:10.48550/arXiv.2310.03693.
98. Talat, Z.; Névéol, A.; Biderman, S.; Clinciu, M.; Dey, M.; Longpre, S.; Luccioni, S.; Masoud, M.; Mitchell, M.; Radev, D.; et al. You Reap What You Sow: On the Challenges of Bias Evaluation Under Multilingual Settings. In Proceedings of the Proceedings of BigScience Episode #5 -- Workshop on Challenges & Perspectives in Creating Large Language Models; Association for Computational Linguistics: virtual+Dublin, 2022; pp. 26–41, doi:10.18653/v1/2022.bigscience-1.3.
99. Kandpal, N.; Wallace, E.; Raffel, C. Deduplicating Training Data Mitigates Privacy Risks in Language Models 2022, doi:10.48550/arXiv.2202.06539.
100. Ozdayi, M.S.; Peris, C.; FitzGerald, J.; Dupuy, C.; Majmudar, J.; Khan, H.; Parikh, R.; Gupta, R. Controlling the Extraction of Memorized Data from Large Language Models via Prompt-Tuning 2023, doi:10.48550/arXiv.2305.11759.
101. Huang, J.; Shao, H.; Chang, K.C.-C. Are Large Pre-Trained Language Models Leaking Your Personal Information? 2022, doi:10.48550/arXiv.2205.12628.
102. Vakili, T.; Lamproudis, A.; Henriksson, A.; Dalianis, H. Downstream Task Performance of BERT Models Pre-Trained Using Automatically De-Identified Clinical Data. In Proceedings of the 13th Language Resources and Evaluation Conference (LREC 2022), Marseille, France, 20–25 June 2022; pp. 4247–4254.
103. Das, B.C.; Amini, M.H.; Wu, Y. Security and Privacy Challenges of Large Language Models: A Survey 2024, doi:10.48550/arXiv.2402.00888.
104. Balunović, M.; Dimitrov, D.I.; Jovanović, N.; Vechev, M. LAMP: Extracting Text from Gradients with Language Model Priors 2022, doi:10.48550/arXiv.2202.08827.
105. Song, C.; Raghunathan, A. Information Leakage in Embedding Models 2020, doi:10.48550/arXiv.2004.00053.
106. Shokri, R.; Stronati, M.; Song, C.; Shmatikov, V. Membership Inference Attacks against Machine Learning Models 2017, doi:10.48550/arXiv.1610.05820.
107. Chen, X.; Tang, S.; Zhu, R.; Yan, S.; Jin, L.; Wang, Z.; Su, L.; Zhang, Z.; Wang, X.; Tang, H. The Janus Interface: How Fine-Tuning in Large Language Models Amplifies the Privacy Risks 2024, doi:10.48550/arXiv.2310.15469.
108. Staab, R.; Vero, M.; Balunović, M.; Vechev, M. Beyond Memorization: Violating Privacy Via Inference with Large Language Models 2024, doi:10.48550/arXiv.2310.07298.
109. Deng, J.; Wang, Y.; Li, J.; Shang, C.; Liu, H.; Rajasekaran, S.; Ding, C. TAG: Gradient Attack on Transformer-Based Language Models 2021, doi:10.48550/arXiv.2103.06819.
110. Feng, Q.; Kasa, S.R.; Kasa, S.K.; Yun, H.; Teo, C.H.; Bodapati, S.B. Exposing Privacy Gaps: Membership Inference Attack on Preference Data for LLM Alignment 2025, doi:10.48550/arXiv.2407.06443.
111. Fu, W.; Wang, H.; Gao, C.; Liu, G.; Li, Y.; Jiang, T. Practical Membership Inference Attacks against Fine-Tuned Large Language Models via Self-Prompt Calibration 2024, doi:10.48550/arXiv.2311.06062.
112. Geiping, J.; Bauermeister, H.; Dröge, H.; Moeller, M. Inverting Gradients -- How Easy Is It to Break Privacy in Federated Learning? 2020, doi:10.48550/arXiv.2003.14053.
113. Ross, R.; Pillitteri, V.; Dempsey, K.; Riddle, M.; Guissanie, G. *Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations*; National Institute of Standards and Technology: Gaithersburg, MD, 2020; p. NIST SP 800-171r2;, doi:10.6028/NIST.SP.800-171r2.

114. Fredrikson, M.; Jha, S.; Ristenpart, T. Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. In Proceedings of the Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security; ACM: Denver Colorado USA, October 12 2015; pp. 1322–1333, doi:10.1145/2810103.2813677.
115. Jayaraman, B.; Wang, L.; Knipmeyer, K.; Gu, Q.; Evans, D. Revisiting Membership Inference Under Realistic Assumptions 2021, doi:10.48550/arXiv.2005.10881.
116. Huang, Y.; Sun, L.; Wang, H.; Wu, S.; Zhang, Q.; Li, Y.; Gao, C.; Huang, Y.; Lyu, W.; Zhang, Y.; et al. TrustLLM: Trustworthiness in Large Language Models 2024, doi:10.48550/arXiv.2401.05561.
117. Atallah, M.J.; Raskin, V.; Hempelmann, C.F.; Karahan, M.; Sion, R.; Topkara, U.; Triezenberg, K.E. Natural Language Watermarking and Tamperproofing. In Information Hiding; Petitcolas, F.A.P., Ed.; Lecture Notes in Computer Science, Vol. 2578; Springer: Berlin, Heidelberg, 2003; pp. 196–212.
118. Jiang, C.; Qi, B.; Hong, X.; Fu, D.; Cheng, Y.; Meng, F.; Yu, M.; Zhou, B.; Zhou, J. On Large Language Models' Hallucination with Regard to Known Facts 2024, doi:10.48550/arXiv.2403.20009.
119. Liu, A.; Pan, L.; Hu, X.; Meng, S.; Wen, L. A Semantic Invariant Robust Watermark for Large Language Models 2024, doi:10.48550/arXiv.2310.06356.
120. Meng, K.; Bau, D.; Andonian, A.; Belinkov, Y. Locating and Editing Factual Associations in GPT 2023, doi:10.48550/arXiv.2202.05262.
121. Duan, H.; Dziedzic, A.; Papernot, N.; Boenisch, F. Flocks of Stochastic Parrots: Differentially Private Prompt Learning for Large Language Models 2023, doi:10.48550/arXiv.2305.15594.
122. Yao, H.; Lou, J.; Ren, K.; Qin, Z. PromptCARE: Prompt Copyright Protection by Watermark Injection and Verification 2023, doi:10.48550/arXiv.2308.02816.
123. Qammar, A.; Wang, H.; Ding, J.; Naouri, A.; Daneshmand, M.; Ning, H. Chatbots to ChatGPT in a Cybersecurity Space: Evolution, Vulnerabilities, Attacks, Challenges, and Future Recommendations 2023, doi:10.48550/arXiv.2306.09255.
124. Glukhov, D.; Shumailov, I.; Gal, Y.; Papernot, N.; Papyan, V. LLM Censorship: A Machine Learning Challenge or a Computer Security Problem? 2023, doi:10.48550/arXiv.2307.10719.
125. Lee, K.; Ippolito, D.; Nystrom, A.; Zhang, C.; Eck, D.; Callison-Burch, C.; Carlini, N. Deduplicating Training Data Makes Language Models Better 2022, doi:10.48550/arXiv.2107.06499.
126. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks 2021, doi:10.48550/arXiv.2005.11401.
127. Christ, M.; Gunn, S.; Zamir, O. Undetectable Watermarks for Language Models 2023, doi:10.48550/arXiv.2306.09194.
128. Kirchenbauer, J.; Geiping, J.; Wen, Y.; Katz, J.; Miers, I.; Goldstein, T. A Watermark for Large Language Models 2024, doi:10.48550/arXiv.2301.10226.
129. Dwork, C.; Roth, A. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science* 2014, *9*, 211–487, doi:10.1561/0400000042.
130. Nguyen, T.T.; Huynh, T.T.; Ren, Z.; Nguyen, P.L.; Liew, A.W.-C.; Yin, H.; Nguyen, Q.V.H. A Survey of Machine Unlearning 2024, doi:10.48550/arXiv.2209.02299.
131. Dhamala, J.; Sun, T.; Kumar, V.; Krishna, S.; Pruksachatkun, Y.; Chang, K.-W.; Gupta, R. BOLD: Dataset and Metrics for Measuring Biases in Open-Ended Language Generation. In Proceedings of the Proceedings of the

2021 ACM Conference on Fairness, Accountability, and Transparency; March 3 2021; pp. 862–872, doi:10.1145/3442188.3445924.

132. Peykani, P.; Ramezanlou, F.; Tanasescu, C.; Ghanidel, S. Large Language Models: A Structured Taxonomy and Review of Challenges, Limitations, Solutions, and Future Directions. *Applied Sciences* 2025, *15*, 8103, doi:10.3390/app15148103.
133. Azamfirei, R.; Kudchadkar, S.R.; Fackler, J. Large Language Models and the Perils of Their Hallucinations. *Crit Care* 2023, *27*, 120, doi:10.1186/s13054-023-04393-x.
134. Bianchi, F.; Suzgun, M.; Attanasio, G.; Röttger, P.; Jurafsky, D.; Hashimoto, T.; Zou, J. Safety-Tuned LLaMAs: Lessons From Improving the Safety of Large Language Models That Follow Instructions 2024, doi:10.48550/arXiv.2309.07875.
135. Attanasio, G.; Nozza, D.; Hovy, D.; Baralis, E. Entropy-Based Attention Regularization Frees Unintended Bias Mitigation from Lists 2022, doi:10.48550/arXiv.2203.09192.
136. Min, S.; Krishna, K.; Lyu, X.; Lewis, M.; Yih, W.; Koh, P.W.; Iyyer, M.; Zettlemoyer, L.; Hajishirzi, H. FActScore: Fine-Grained Atomic Evaluation of Factual Precision in Long Form Text Generation 2023, doi:10.48550/arXiv.2305.14251.
137. Bergmair, R. Towards Linguistic Steganography: A Systematic Investigation of Approaches, Systems, and Issues. Ph.D. Thesis, University of Cambridge, Cambridge, UK, 2004. Available online: https://richard.bergmair.eu/dwn/Bergmair2004Bb5.pdf (accessed on 5 March 2026).
138. Dev, S.; Sheng, E.; Zhao, J.; Amstutz, A.; Sun, J.; Hou, Y.; Sanseverino, M.; Kim, J.; Nishi, A.; Peng, N.; et al. On Measures of Biases and Harms in NLP 2022, doi:10.48550/arXiv.2108.03362.
139. Chen, C.; Shu, K. Combating Misinformation in the Age of LLMs: Opportunities and Challenges 2023, doi:10.48550/arXiv.2311.05656.
140. Jin, X.; Larson, J.; Yang, W.; Lin, Z. Binary Code Summarization: Benchmarking ChatGPT/GPT-4 and Other Large Language Models 2023, doi:10.48550/arXiv.2312.09601.
141. Rozado, D. The Political Biases of ChatGPT. *Social Sciences* 2023, *12*, 148, doi:10.3390/socsci12030148.
142. Lin, S.; Hilton, J.; Evans, O. TruthfulQA: Measuring How Models Mimic Human Falsehoods. In Proceedings of the Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Association for Computational Linguistics: Dublin, Ireland, 2022; pp. 3214–3252, doi:10.18653/v1/2022.acl-long.229.
143. Pan, Y.; Pan, L.; Chen, W.; Nakov, P.; Kan, M.-Y.; Wang, W.Y. On the Risk of Misinformation Pollution with Large Language Models 2023, doi:10.48550/arXiv.2305.13661.
144. Sun, H.; Pei, J.; Choi, M.; Jurgens, D. Sociodemographic Prompting Is Not Yet an Effective Approach for Simulating Subjective Judgments with LLMs 2025, doi:10.48550/arXiv.2311.09730.
145. Sun, Z.; Shen, Y.; Zhou, Q.; Zhang, H.; Chen, Z.; Cox, D.; Yang, Y.; Gan, C. Principle-Driven Self-Alignment of Language Models from Scratch with Minimal Human Supervision 2023, doi:10.48550/arXiv.2305.03047.
146. Nadeem, M.; Bethke, A.; Reddy, S. StereoSet: Measuring Stereotypical Bias in Pretrained Language Models. In Proceedings of the Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers); Association for Computational Linguistics: Online, 2021; pp. 5356–5371, doi:10.18653/v1/2021.acl-long.416.
147. Lee, N.; Ping, W.; Xu, P.; Patwary, M.; Fung, P.; Shoeybi, M.; Catanzaro, B. Factuality Enhanced Language Models for Open-Ended Text Generation 2023, doi:10.48550/arXiv.2206.04624.

148. Kumar, S.; Balachandran, V.; Njoo, L.; Anastasopoulos, A.; Tsvetkov, Y. Language Generation Models Can Cause Harm: So What Can We Do About It? An Actionable Survey. In Proceedings of the Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics; Association for Computational Linguistics: Dubrovnik, Croatia, 2023; pp. 3299–3321, doi:10.18653/v1/2023.eacl-main.241.
149. Chuang, Y.-S.; Xie, Y.; Luo, H.; Kim, Y.; Glass, J.; He, P. DoLa: Decoding by Contrasting Layers Improves Factuality in Large Language Models 2024, doi:10.48550/arXiv.2309.03883.
150. Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.J.; Madotto, A.; Fung, P. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.* 2023, *55*, 1–38, doi:10.1145/3571730.
151. Shuster, K.; Poff, S.; Chen, M.; Kiela, D.; Weston, J. Retrieval Augmentation Reduces Hallucination in Conversation. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021; Association for Computational Linguistics: Punta Cana, Dominican Republic, 2021; pp. 3784–3803, doi:10.18653/v1/2021.findings-emnlp.320.
152. Li, J.; Cheng, X.; Zhao, X.; Nie, J.-Y.; Wen, J.-R. HaluEval: A Large-Scale Hallucination Evaluation Benchmark for Large Language Models. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Singapore, 2023; pp. 6449–6464, doi:10.18653/v1/2023.emnlp-main.397.
153. Májovský, M.; Černý, M.; Kasal, M.; Komarc, M.; Netuka, D. Artificial Intelligence Can Generate Fraudulent but Authentic-Looking Scientific Medical Articles: Pandora's Box Has Been Opened. *J Med Internet Res* 2023, *25*, e46924, doi:10.2196/46924.
154. Lawyer Cites Fake Cases Generated by ChatGPT in Legal Brief | Legal Dive Available online: https://www.legaldive.com/news/chatgpt-fake-legal-cases-generative-ai-hallucinations/651557/ (accessed on 22 February 2026).
155. Melbourne Lawyer Referred to Complaints Body after AI Generated Made-up Case Citations in Family Court | Australian Law | The Guardian Available online: https://www.theguardian.com/law/2024/oct/10/melbourne-lawyer-referred-to-complaints-body-after-ai-generated-made-up-case-citations-in-family-court-ntwnfb (accessed on 22 February 2026).
156. Jha, S.; Jha, S.; Lincoln, P.; Bastian, N.D.; Velasquez, A.; Neema, S. Dehallucinating Large Language Models Using Formal Methods Guided Iterative Prompting. In Proceedings of the 2023 IEEE International Conference on Assured Autonomy (ICAA), Laurel, MD, USA, 6–8 June 2023; IEEE: Piscataway, NJ, USA, 2023; https://doi.org/10.1109/ICAA58325.2023.00029
157. Manakul, P.; Liusie, A.; Gales, M.J.F. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models 2023, doi:10.48550/arXiv.2303.08896.
158. Ferrara, E. Should ChatGPT Be Biased? Challenges and Risks of Bias in Large Language Models. *FM* 2023, doi:10.5210/fm.v28i11.13346.
159. Blodgett, S.L.; Barocas, S.; Daumé, H.; Wallach, H. Language (Technology) Is Power: A Critical Survey of "Bias" in NLP 2020, doi:10.48550/arXiv.2005.14050.
160. Caliskan, A.; Bryson, J.J.; Narayanan, A. Semantics Derived Automatically from Language Corpora Contain Human-like Biases. *Science* 2017, *356*, 183–186, doi:10.1126/science.aal4230.
161. Schramowski, P.; Turan, C.; Andersen, N.; Rothkopf, C.A.; Kersting, K. Large Pre-Trained Language Models Contain Human-like Biases of What Is Right and Wrong to Do 2022, doi:10.48550/arXiv.2103.11790.
162. Kotek, H.; Dockum, R.; Sun, D.Q. Gender Bias and Stereotypes in Large Language Models. In Proceedings of the Proceedings of The ACM Collective Intelligence Conference; November 6 2023; pp. 12–24, doi:10.1145/3582269.3615599.

163. Zhao, J.; Ding, Y.; Jia, C.; Wang, Y.; Qian, Z. Gender Bias in Large Language Models across Multiple Languages 2024, doi:10.48550/arXiv.2403.00277.
164. Ahn, J.; Oh, A. Mitigating Language-Dependent Ethnic Bias in BERT. In Proceedings of the Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Online and Punta Cana, Dominican Republic, 2021; pp. 533–549, doi:10.18653/v1/2021.emnlp-main.42.
165. Hardt, M.; Price, E.; Srebro, N. Equality of Opportunity in Supervised Learning 2016, doi:10.48550/arXiv.1610.02413.
166. Hendrycks, D.; Burns, C.; Basart, S.; Critch, A.; Li, J.; Song, D.; Steinhardt, J. Aligning AI With Shared Human Values 2023, doi:10.48550/arXiv.2008.02275.
167. Maudslay, R.H.; Gonen, H.; Cotterell, R.; Teufel, S. It's All in the Name: Mitigating Gender Bias with Name-Based Counterfactual Data Substitution 2020, doi:10.48550/arXiv.1909.00871.
168. Schick, T.; Udupa, S.; Schütze, H. Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP 2021, doi:10.48550/arXiv.2103.00453.
169. Gupta, M.; Akiri, C.; Aryal, K.; Parker, E.; Praharaj, L. From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy 2023, doi:10.48550/arXiv.2307.00691.
170. Hazell, J. Spear Phishing With Large Language Models 2023, doi:10.48550/arXiv.2305.06972.
171. Pearce, H.; Tan, B.; Ahmad, B.; Karri, R.; Dolan-Gavitt, B. Examining Zero-Shot Vulnerability Repair with Large Language Models 2022, doi:10.48550/arXiv.2112.02125.
172. De Angelis, L.; Baglivo, F.; Arzilli, G.; Privitera, G.P.; Ferragina, P.; Tozzi, A.E.; Rizzo, C. ChatGPT and the Rise of Large Language Models: The New AI-Driven Infodemic Threat in Public Health. *Front. Public Health* 2023, *11*, 1166120, doi:10.3389/fpubh.2023.1166120.
173. Lucas, J.; Uchendu, A.; Yamashita, M.; Lee, J.; Rohatgi, S.; Lee, D. Fighting Fire with Fire: The Dual Role of LLMs in Crafting and Detecting Elusive Disinformation. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Singapore, 2023; pp. 14279–14305, doi:10.18653/v1/2023.emnlp-main.883.
174. Liu, F.; Jiang, J.; Lu, Y.; Huang, Z.; Jiang, J. The Ethical Security of Large Language Models: A Systematic Review. *Front. Eng. Manag.* 2025, *12*, 128–140, doi:10.1007/s42524-025-4082-6.
175. Davidson, T.; Warmsley, D.; Macy, M.; Weber, I. Automated Hate Speech Detection and the Problem of Offensive Language 2017, doi:10.48550/arXiv.1703.04009.
176. Gehman, S.; Gururangan, S.; Sap, M.; Choi, Y.; Smith, N.A. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models 2020, doi:10.48550/arXiv.2009.11462.
177. Jin, D.; Pan, E.; Oufattole, N.; Weng, W.-H.; Fang, H.; Szolovits, P. What Disease Does This Patient Have? A Large-Scale Open Domain Question Answering Dataset from Medical Exams 2020, doi:10.48550/arXiv.2009.13081.
178. Jin, Q.; Dhingra, B.; Liu, Z.; Cohen, W.; Lu, X. PubMedQA: A Dataset for Biomedical Research Question Answering. In Proceedings of the Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP); Association for Computational Linguistics: Hong Kong, China, 2019; pp. 2567–2577, doi:10.18653/v1/D19-1259.
179. Ren, J.; Luo, J.; Zhao, Y.; Krishna, K.; Saleh, M.; Lakshminarayanan, B.; Liu, P.J. Out-of-Distribution Detection and Selective Generation for Conditional Language Models 2023, doi:10.48550/arXiv.2209.15558.
180. Whitty, M.T. How to Conduct AI-Assisted (Large Language Model-Assisted) Content Analysis in Information Science and Cyber Security Research. *Electronics* 2025, *14*, 4104, doi:10.3390/electronics14204104.

181. Inan, H.; Upasani, K.; Chi, J.; Rungta, R.; Iyer, K.; Mao, Y.; Tontchev, M.; Hu, Q.; Fuller, B.; Testuggine, D.; et al. Llama Guard: LLM-Based Input-Output Safeguard for Human-AI Conversations 2023, doi:10.48550/arXiv.2312.06674.
182. Hou, A.; Zhang, J.; He, T.; Wang, Y.; Chuang, Y.-S.; Wang, H.; Shen, L.; Van Durme, B.; Khashabi, D.; Tsvetkov, Y. SemStamp: A Semantic Watermark with Paraphrastic Robustness for Text Generation. In Proceedings of the Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers); Association for Computational Linguistics: Mexico City, Mexico, 2024; pp. 4067–4082, doi:10.18653/v1/2024.naacl-long.226.
183. Li, X.; Tramèr, F.; Liang, P.; Hashimoto, T. Large Language Models Can Be Strong Differentially Private Learners 2022, doi:10.48550/arXiv.2110.05679.
184. Yuan, Y.; Jiao, W.; Wang, W.; Huang, J.; He, P.; Shi, S.; Tu, Z. GPT-4 Is Too Smart To Be Safe: Stealthy Chat with LLMs via Cipher 2024, doi:10.48550/arXiv.2308.06463.
185. Biderman, S.; Schoelkopf, H.; Anthony, Q.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M.A.; Purohit, S.; Prashanth, U.S.; Raff, E.; et al. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling 2023, doi:10.48550/arXiv.2304.01373.
186. Chakrabarty, T.; Laban, P.; Agarwal, D.; Muresan, S.; Wu, C.-S. Art or Artifice? Large Language Models and the False Promise of Creativity 2024, doi:10.48550/arXiv.2309.14556.
187. Cavalcanti, A.P.; Barbosa, A.; Carvalho, R.; Freitas, F.; Tsai, Y.-S.; Gašević, D.; Mello, R.F. Automatic Feedback in Online Learning Environments: A Systematic Literature Review. *Computers and Education: Artificial Intelligence* 2021, *2*, 100027, doi:10.1016/j.caeai.2021.100027.
188. Xie, Y.; Yi, J.; Shao, J.; Curl, J.; Lyu, L.; Chen, Q.; Xie, X.; Wu, F. Defending ChatGPT against Jailbreak Attack via Self-Reminders. *Nat Mach Intell* 2023, *5*, 1486–1496, doi:10.1038/s42256-023-00765-8.
189. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A. y Communication-Efficient Learning of Deep Networks from Decentralized Data 2023, doi:10.48550/arXiv.1602.05629.
190. Gonen, H.; Iyer, S.; Blevins, T.; Smith, N.A.; Zettlemoyer, L. Demystifying Prompts in Language Models via Perplexity Estimation 2024, doi:10.48550/arXiv.2212.04037.
191. Ren, J.; Xu, H.; Liu, Y.; Cui, Y.; Wang, S.; Yin, D.; Tang, J. A Robust Semantics-Based Watermark for Large Language Model against Paraphrasing. In Proceedings of the Findings of the Association for Computational Linguistics: NAACL 2024; Association for Computational Linguistics: Mexico City, Mexico, 2024; pp. 613–625, doi:10.18653/v1/2024.findings-naacl.40.
192. Ge, S.; Zhou, C.; Hou, R.; Khabsa, M.; Wang, Y.-C.; Wang, Q.; Han, J.; Mao, Y. MART: Improving LLM Safety with Multi-Round Automatic Red-Teaming 2023, doi:10.48550/arXiv.2311.07689.
193. Perez, E.; Huang, S.; Song, F.; Cai, T.; Ring, R.; Aslanides, J.; Glaese, A.; McAleese, N.; Irving, G. Red Teaming Language Models with Language Models 2022, doi:10.48550/arXiv.2202.03286.
194. Ji, J.; Liu, M.; Dai, J.; Pan, X.; Zhang, C.; Bian, C.; Zhang, C.; Sun, R.; Wang, Y.; Yang, Y. BeaverTails: Towards Improved Safety Alignment of LLM via a Human-Preference Dataset 2023, doi:10.48550/arXiv.2307.04657.
195. Wu, T.; Zhu, B.; Zhang, R.; Wen, Z.; Ramchandran, K.; Jiao, J. Pairwise Proximal Policy Optimization: Harnessing Relative Feedback for LLM Alignment 2023, doi:10.48550/arXiv.2310.00212.
196. Amabile, T.M. Social Psychology of Creativity: A Consensual Assessment Technique. *Journal of Personality and Social Psychology* 1982, *43*, 997–1013, doi:10.1037/0022-3514.43.5.997.
197. Jiang, L.; Zhou, H.; Lin, Y.; Li, P.; Zhou, J.; Jiang, R. ROSE: Robust Selective Fine-Tuning for Pre-Trained Language Models 2022, doi:10.48550/arXiv.2210.09658.

198. Hasan, A.; Rugina, I.; Wang, A. Pruning for Protection: Increasing Jailbreak Resistance in Aligned LLMs Without Fine-Tuning 2024, doi:10.48550/arXiv.2401.10862.
199. Hinton, G.; Vinyals, O.; Dean, J. Distilling the Knowledge in a Neural Network 2015, doi:10.48550/arXiv.1503.02531.
200. Detecting and Preventing Distillation Attacks Available online: https://www.anthropic.com/news/detecting-and-preventing-distillation-attacks (accessed on 6 March 2026).
201. Chen, B.; Paliwal, A.; Yan, Q. Jailbreaker in Jail: Moving Target Defense for Large Language Models 2023, doi:10.48550/arXiv.2310.02417.
202. Dale, D.; Voronov, A.; Dementieva, D.; Logacheva, V.; Kozlova, O.; Semenov, N.; Panchenko, A. Text Detoxification Using Large Pre-Trained Neural Models 2021, doi:10.48550/arXiv.2109.08914.
203. Alon, G.; Kamfonas, M. Detecting Language Model Attacks with Perplexity 2023, doi:10.48550/arXiv.2308.14132.
204. Zhang, Z.; Yang, J.; Ke, P.; Mi, F.; Wang, H.; Huang, M. Defending Large Language Models Against Jailbreaking Attacks Through Goal Prioritization 2024, doi:10.48550/arXiv.2311.09096.
205. Zhang, Y.; Ding, L.; Zhang, L.; Tao, D. Intention Analysis Makes LLMs A Good Jailbreak Defender 2024, doi:10.48550/arXiv.2401.06561.
206. Cohen, J.M.; Rosenfeld, E.; Kolter, J.Z. Certified Adversarial Robustness via Randomized Smoothing 2019, doi:10.48550/arXiv.1902.02918.
207. Xiao, Y.; Jin, Y.; Bai, Y.; Wu, Y.; Yang, X.; Luo, X.; Yu, W.; Zhao, X.; Liu, Y.; Gu, Q.; et al. PrivacyMind: Large Language Models Can Be Contextual Privacy Protection Learners 2024, doi:10.48550/arXiv.2310.02469.
208. Anil, R.; Dai, A.M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. PaLM 2 Technical Report 2023, doi:10.48550/arXiv.2305.10403.
209. Wen, J.; Ke, P.; Sun, H.; Zhang, Z.; Li, C.; Bai, J.; Huang, M. Unveiling the Implicit Toxicity in Large Language Models. In Proceedings of the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Singapore, 2023; pp. 1322–1338, doi:10.18653/v1/2023.emnlp-main.84.
210. Cramer, R.; Damgård, I.B.; Nielsen, J.B. *Secure Multiparty Computation and Secret Sharing*; 1st ed.; Cambridge University Press: Cambridge, UK, 2015; ISBN 978-1-107-33775-6.
211. Plant, R.; Giuffrida, V.; Gkatzia, D. You Are What You Write: Preserving Privacy in the Era of Large Language Models 2022, doi:10.48550/arXiv.2204.09391.
212. Gao, C.A.; Howard, F.M.; Markov, N.S.; Dyer, E.C.; Ramesh, S.; Luo, Y.; Pearson, A.T. Comparing Scientific Abstracts Generated by ChatGPT to Real Abstracts with Detectors and Blinded Human Reviewers. *npj Digit. Med.* 2023, *6*, 75, doi:10.1038/s41746-023-00819-6.
213. Mitchell, E.; Lee, Y.; Khazatsky, A.; Manning, C.D.; Finn, C. DetectGPT: Zero-Shot Machine-Generated Text Detection Using Probability Curvature 2023, doi:10.48550/arXiv.2301.11305.
214. Kandpal, N.; Jagielski, M.; Tramèr, F.; Carlini, N. Backdoor Attacks for In-Context Learning with Language Models 2023, doi:10.48550/arXiv.2307.14692.
215. Huang, X.; Ruan, W.; Huang, W.; Jin, G.; Dong, Y.; Wu, C.; Bensalem, S.; Mu, R.; Qi, Y.; Zhao, X.; et al. A Survey of Safety and Trustworthiness of Large Language Models through the Lens of Verification and Validation 2023, doi:10.48550/arXiv.2305.11391.
216. Hacker, P.; Engel, A.; Mauer, M. Regulating ChatGPT and Other Large Generative AI Models 2023, doi:10.48550/arXiv.2302.02337.

217. Smajić, A.; Karlović, R.; Bobanović Dasko, M.; Lorencin, I. Large Language Models for Structured and Semi-Structured Data, Recommender Systems and Knowledge Base Engineering: A Survey of Recent Techniques and Architectures. *Electronics* 2025, *14*, 3153, doi:10.3390/electronics14153153.
218. Guha, N.; Nyarko, J.; Ho, D.E.; Ré, C.; Chilton, A.; Narayana, A.; Chohlas-Wood, A.; Peters, A.; Waldon, B.; Rockmore, D.N.; et al. LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models 2023, doi:10.48550/arXiv.2308.11462.
219. Sarsa, S.; Denny, P.; Hellas, A.; Leinonen, J. Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models. In Proceedings of the Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 1; August 3 2022; pp. 27–43, doi:10.1145/3501385.3543957.
220. Agathokleous, E.; Saitanis, C.J.; Fang, C.; Yu, Z. Use of ChatGPT: What Does It Mean for Biology and Environmental Science? *Science of The Total Environment* 2023, *888*, 164154, doi:10.1016/j.scitotenv.2023.164154.
221. Cascella, M.; Montomoli, J.; Bellini, V.; Bignami, E. Evaluating the Feasibility of ChatGPT in Healthcare: An Analysis of Multiple Clinical and Research Scenarios. *J Med Syst* 2023, *47*, 33, doi:10.1007/s10916-023-01925-4.
222. Chaudhry, M.A.; Cukurova, M.; Luckin, R. A Transparency Index Framework for AI in Education 2022, doi:10.48550/arXiv.2206.03220.
223. Geller, S.A.; Gal, K.; Segal, A.; Sripathi, K.; Kim, H.G.; Facciotti, M.T.; Igo, M.; Hoernle, N.; Karger, D. New Methods for Confusion Detection in Course Forums: Student, Teacher, and Machine. *IEEE Trans. Learning Technol.* 2021, *14*, 665–679, doi:10.1109/TLT.2021.3123266.
224. Cibu, B.-R.; Crăciun, L.; Molănescu, A.G.; Cotfas, L.-A. Exploring the Educational Applications of Large Language Models: A Systematic Review and Topic Analysis. *Electronics* 2025, *14*, 4683, doi:10.3390/electronics14234683.
225. Kasneci, E.; Sessler, K.; Küchemann, S.; Bannert, M.; Dementieva, D.; Fischer, F.; Gasser, U.; Groh, G.; Günnemann, S.; Hüllermeier, E.; et al. ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education. *Learning and Individual Differences* 2023, *103*, 102274, doi:10.1016/j.lindif.2023.102274.
226. Birhane, A.; Kasirzadeh, A.; Leslie, D.; Wachter, S. Science in the Age of Large Language Models. *Nat Rev Phys* 2023, *5*, 277–280, doi:10.1038/s42254-023-00581-4.
227. Alimardani, A. Borderline Disaster: An Empirical Study on Student Usage of GenAI in a Law Assignment. *IEEE Trans. Technol. Soc.* 2025, *6*, 210–219, doi:10.1109/TTS.2025.3540978.
228. Borges, B.; Foroutan, N.; Bayazit, D.; Sotnikova, A.; Montariol, S.; Nazaretzky, T.; Banaei, M.; Sakhaeirad, A.; Servant, P.; Neshaei, S.P.; et al. Could ChatGPT Get an Engineering Degree? Evaluating Higher Education Vulnerability to AI Assistants. *Proc. Natl. Acad. Sci. U.S.A.* 2024, *121*, e2414955121, doi:10.1073/pnas.2414955121.
229. Lund, B.; Wang, T.; Mannuru, N.R.; Nie, B.; Shimray, S.; Wang, Z. ChatGPT and a New Academic Reality: Artificial Intelligence-Written Research Papers and the Ethics of the Large Language Models in Scholarly Publishing. *Asso for Info Science & Tech* 2023, *74*, 570–581, doi:10.1002/asi.24750.
230. Fan, Y.; Tang, L.; Le, H.; Shen, K.; Tan, S.; Zhao, Y.; Shen, Y.; Li, X.; Gašević, D. Beware of Metacognitive Laziness: Effects of Generative Artificial Intelligence on Learning Motivation, Processes, and Performance. *Brit J Educational Tech* 2025, *56*, 489–530, doi:10.1111/bjet.13544.
231. Caines, A.; Benedetto, L.; Taslimipoor, S.; Davis, C.; Gao, Y.; Andersen, O.; Yuan, Z.; Elliott, M.; Moore, R.; Bryant, C.; et al. On the Application of Large Language Models for Language Teaching and Assessment Technology 2023, doi:10.48550/arXiv.2307.08393.

232. Prather, J.; Denny, P.; Leinonen, J.; Becker, B.A.; Albluwi, I.; Craig, M.; Keuning, H.; Kiesler, N.; Kohn, T.; Luxton-Reilly, A.; et al. The Robots Are Here: Navigating the Generative AI Revolution in Computing Education. In Proceedings of the Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education; December 22 2023; pp. 108–159, doi:10.1145/3623762.3633499.
233. Deng, Y.; Xia, C.S.; Peng, H.; Yang, C.; Zhang, L. Large Language Models Are Zero-Shot Fuzzers: Fuzzing Deep-Learning Libraries via Large Language Models 2023, doi:10.48550/arXiv.2212.14834.
234. Lee, D.K.C.; Guan, C.; Yu, Y.; Ding, Q. A Comprehensive Review of Generative AI in Finance. *FinTech* 2024, *3*, 460–478, doi:10.3390/fintech3030025.
235. Joshi, S. Review of Gen AI Models for Financial Risk Management: Architectural Frameworks and Implementation Strategies. *IJISEM* 2025, 207–222, doi:10.69968/ijisem.2025v4i2207-222.
236. Saha, B.; Rani, N.; Shukla, S.K. Generative AI in Financial Institution: A Global Survey of Opportunities, Threats, and Regulation 2025, doi:10.48550/arXiv.2504.21574.
237. Billah, M.M.; Hamjaya, H.S.; Shiralizade, H.; Singh, V.; Inam, R. Large Language Models' Trustworthiness in the Light of the EU AI Act—A Systematic Mapping Study. *Applied Sciences* 2025, *15*, 7640, doi:10.3390/app15147640.
238. Raji, I.D.; Smart, A.; White, R.N.; Mitchell, M.; Gebru, T.; Hutchinson, B.; Smith-Loud, J.; Theron, D.; Barnes, P. Closing the AI Accountability Gap: Defining an End-to-End Framework for Internal Algorithmic Auditing 2020, doi:10.48550/arXiv.2001.00973.
239. European Parliament and Council. Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, 2024, L, 12 July 2024. Available online: https://eur-lex.europa.eu/eli/reg/2024/1689/oj (accessed on 5 March 2026).
240. Haag, S.; Eckhardt, A. Shadow IT. *Bus Inf Syst Eng* 2017, *59*, 469–473, doi:10.1007/s12599-017-0497-x.
241. Silic, M.; Back, A. Shadow IT – A View from behind the Curtain. *Computers & Security* 2014, *45*, 274–283, doi:10.1016/j.cose.2014.06.007.
242. Davenport, T.H.; Ronanki, R. Artificial Intelligence for the Real World. Harvard Business Review 2018, 96, 108–116. Available online: https://hbr.org/2018/01/artificial-intelligence-for-the-real-world (accessed on 5 March 2026).
243. Market Guide for AI Governance Platforms Available online: https://www.gartner.com/en/documents/7145930 (accessed on 7 March 2026).
244. Yao, D.; Zhang, J.; Harris, I.G.; Carlsson, M. FuzzLLM: A Novel and Universal Fuzzing Framework for Proactively Discovering Jailbreak Vulnerabilities in Large Language Models. In Proceedings of the ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); April 14 2024; pp. 4485–4489, doi:10.1109/ICASSP48485.2024.10448041.
245. Shu, M.; Wang, J.; Zhu, C.; Geiping, J.; Xiao, C.; Goldstein, T. On the Exploitability of Instruction Tuning 2023, doi:10.48550/arXiv.2306.17194.
246. Zhou, C.; Liu, P.; Xu, P.; Iyer, S.; Sun, J.; Mao, Y.; Ma, X.; Efrat, A.; Yu, P.; Yu, L.; et al. LIMA: Less Is More for Alignment 2023, doi:10.48550/arXiv.2305.11206.
247. Brandao, P.R. The Impact of Artificial Intelligence on Modern Society. *AI* 2025, *6*, 190, doi:10.3390/ai6080190.