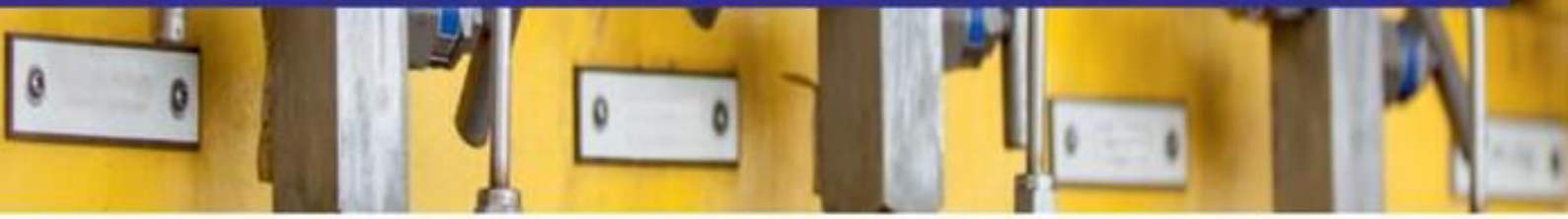
IJAREEIE

e-ISSN: 2278-8875
p-ISSN: 2320-3765

# International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

Volume 14, Issue 8, August 2025



INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.807

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREEIE)

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

# AI Safety in Deployed Systems: Red-Teaming Techniques, Constitutional AI Evaluation, And Automated Jailbreak Detection for Large Language Models in High-Stakes Enterprise and Government Applications

**Venkata Vijay Satyanarayana Murthy Neelam**

Senior Software Engineer (Cloud, Data, AI/ML, GEN AI), Atlanta, Georgia, USA

**ABSTRACT:** The deployment of large language models (LLMs) in high-stakes enterprise and government applications-including healthcare diagnostics, financial advisory systems, legal research platforms, defense intelligence analysis, and critical infrastructure management-introduces safety risks that demand rigorous, systematic, and continuously evolving evaluation methodologies. This paper presents a comprehensive analysis of three interconnected pillars of deployed AI safety: red-teaming techniques for proactive vulnerability discovery, Constitutional AI (CAI) as a scalable alignment and evaluation framework, and automated jailbreak detection systems for real-time production defense.

We examine the evolution of red-teaming from manual, exploratory testing to sophisticated algorithmic approaches including PAIR (Prompt Automatic Iterative Refinement), Crescendo multi-turn escalation, GCG token-level optimization, and AutoDAN-Turbo lifelong adversarial agents. We analyze Anthropic's Constitutional AI framework-from its foundational RLAIF methodology through the recently deployed Constitutional Classifiers++ system, which achieves a 95% jailbreak block rate with only ~1% additional compute overhead. We evaluate six categories of automated jailbreak detection systems, benchmarking accuracy against latency across rule-based filters, fine-tuned classifiers, activation probes, and LLM-as-judge approaches. Our analysis synthesizes findings from over 1,700 cumulative hours of expert red-teaming, published attack success rate benchmarks, and production deployment data from Anthropic, OpenAI, Microsoft, and Meta. We propose a risk-stratified deployment framework for enterprise and government applications that maps safety evaluation intensity to deployment context criticality, and identify twelve open research challenges that must be addressed to achieve robust AI safety at scale.

**KEYWORDS:** AI Safety · Red-Teaming · Constitutional AI · RLHF · RLAIF · Jailbreak Detection · Constitutional Classifiers · Prompt Injection · Automated Adversarial Testing · Enterprise AI · Government AI · CBRN Safety · OWASP Top 10 LLM

## I. INTRODUCTION

The rapid deployment of large language models across enterprise and government applications has created an urgent need for systematic safety evaluation methodologies that can operate at production scale. Unlike traditional software security-where vulnerabilities are deterministic and reproducible-LLM safety challenges are fundamentally stochastic: the same model may respond safely to a query one thousand times and produce harmful output on the next attempt. This non-deterministic failure mode, combined with the expanding capability frontier of modern LLMs, means that safety cannot be assured through one-time testing alone. Instead, it requires continuous, multi-layered evaluation architectures that combine proactive adversarial testing (red-teaming), principled alignment training (Constitutional AI), and real-time runtime defense (automated jailbreak detection).

The stakes are particularly high in government and enterprise contexts. A jailbroken healthcare LLM could provide dangerous medical advice. A compromised financial advisory system could recommend harmful investment strategies. A defense intelligence LLM that leaks classified information could endanger national security. These scenarios are not hypothetical: published research demonstrates that leading safety-aligned LLMs remain vulnerable to sophisticated jailbreak attacks, with attack success rates varying from 4% to over 80% depending on the technique and defense configuration. Anthropic's Constitutional Classifiers reduced jailbreak success from 86% to 4.4%-blocking 95% of attacks-yet even this represents thousands of potential successful attacks across millions of daily interactions.

This paper provides a unified analysis of the three critical pillars of deployed AI safety, structured as follows. Section II examines red-teaming methodologies, from manual expert testing through state-of-the-art algorithmic approaches.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

Section III analyzes Constitutional AI as both an alignment training methodology and an evaluation framework. Section IV evaluates automated jailbreak detection systems for production deployment. Section V presents a risk-stratified deployment framework for high-stakes applications. Section VI identifies open challenges and future research directions.

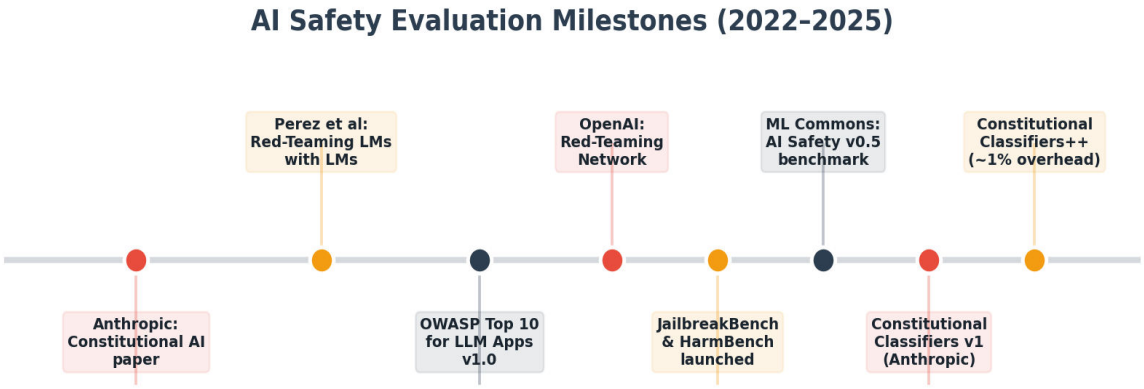



**Figure 1.** *AI safety evaluation milestones from Constitutional AI (2022) through Constitutional Classifiers++ (2025)*

## II. RED-TEAMING METHODOLOGIES

### 2.1 Evolution of LLM Red-Teaming

Red-teaming for AI safety has evolved through three distinct generations. First-generation red-teaming (2022–2023) was predominantly manual and exploratory: human testers crafted adversarial prompts based on intuition, domain expertise, and known vulnerability patterns. Anthropic's early red-teaming efforts, OpenAI's Red Teaming Network, and Google DeepMind's frontier safety testing all relied primarily on human experts probing model boundaries. This approach excels at discovering novel, creative attack vectors that automated systems might miss, but suffers from limited scalability, human cognitive biases in attack selection, and the inability to systematically cover the vast input space of modern LLMs.

Second-generation red-teaming (2023–2024) introduced algorithmic automation using LLMs themselves as adversarial agents. The PAIR (Prompt Automatic Iterative Refinement) paradigm, introduced by Chao et al. (2024), uses an attacker LLM to iteratively refine jailbreak prompts based on target model responses, achieving high attack success rates through role-playing and multi-turn escalation. The Crescendo technique extends this to multi-turn conversations where each turn gradually escalates toward harmful content. GCG (Greedy Coordinate Gradient) optimization operates at the token level, generating adversarial suffixes that bypass safety alignment through gradient-based search. AutoDAN-Turbo introduces a lifelong adversarial agent that accumulates attack strategies over time, achieving state-of-the-art attack success rates.

Third-generation red-teaming (2025) combines human creativity with automated scalability in hybrid architectures. Microsoft's PyRIT (Python Risk Identification Tool) orchestrates multi-model attack pipelines where different LLMs serve as attacker, target, and judge. Anthropic's automated red-teaming generates synthetic adversarial examples that are used both for safety training and classifier development. The hybrid approach leverages human expertise for threat model design and novel vulnerability discovery while using automation for systematic coverage, regression testing, and continuous monitoring.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

**Table I.** *Red-Teaming Attack Technique Taxonomy*

| Technique | Category | Mechanism | ASR Range | Automation |
|---|---|---|---|---|
| **Direct Prompt** | Manual | Explicit harmful requests without obfuscation | 5–15% | None (human-crafted) |
| **Role-Playing (PAIR)** | Algorithmic | LLM iteratively refines prompts via persona adoption | 25–65% | Full (LLM attacker) |
| **Crescendo** | Multi-Turn | Gradual escalation across conversation turns | 30–58% | Semi-automated |
| **GCG Optimization** | Token-Level | Gradient-based adversarial suffix generation | 20–48% | Full (gradient search) |
| **Many-Shot Jailbreak** | Context | Exploiting long-context windows with repeated examples | 40–82% | Semi-automated |
| **AutoDAN-Turbo** | Lifelong Agent | Strategy-accumulating adversarial agent | 45–70% | Full (RL-based) |
| **Rainbow Teaming** | Diversity | Quality-diversity search over attack space | 35–55% | Full (MAP-Elites) |
| **Image Overlay** | Multimodal | Text instructions embedded in images | 30–50% | Semi-automated |

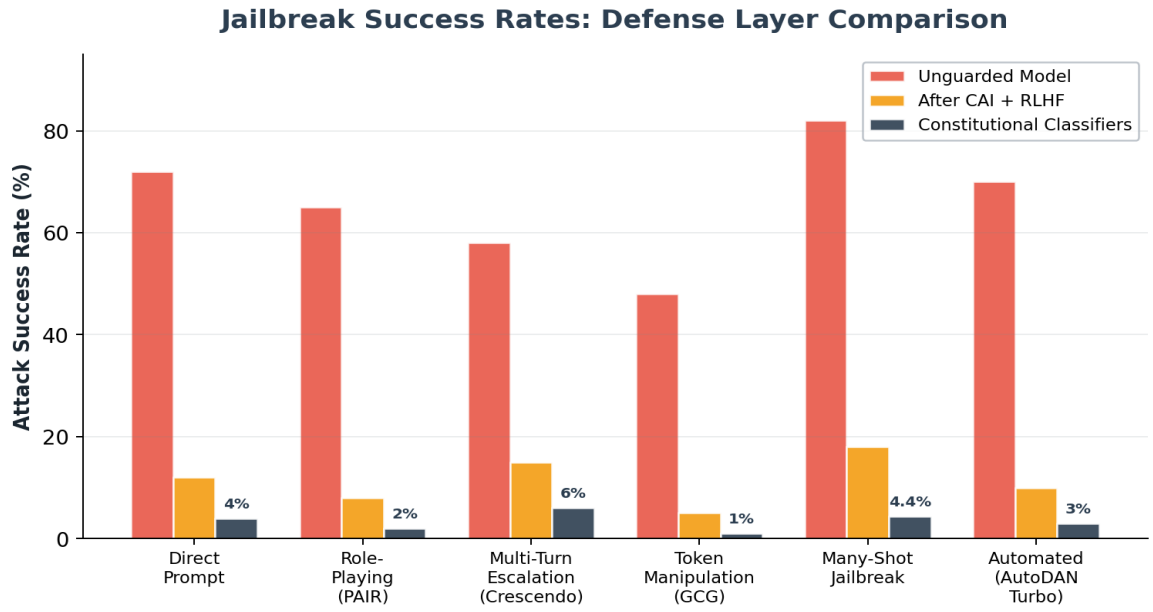



**Figure 2.** *Jailbreak success rates across three defense configurations: unguarded, CAI+RLHF, and Constitutional Classifiers*

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

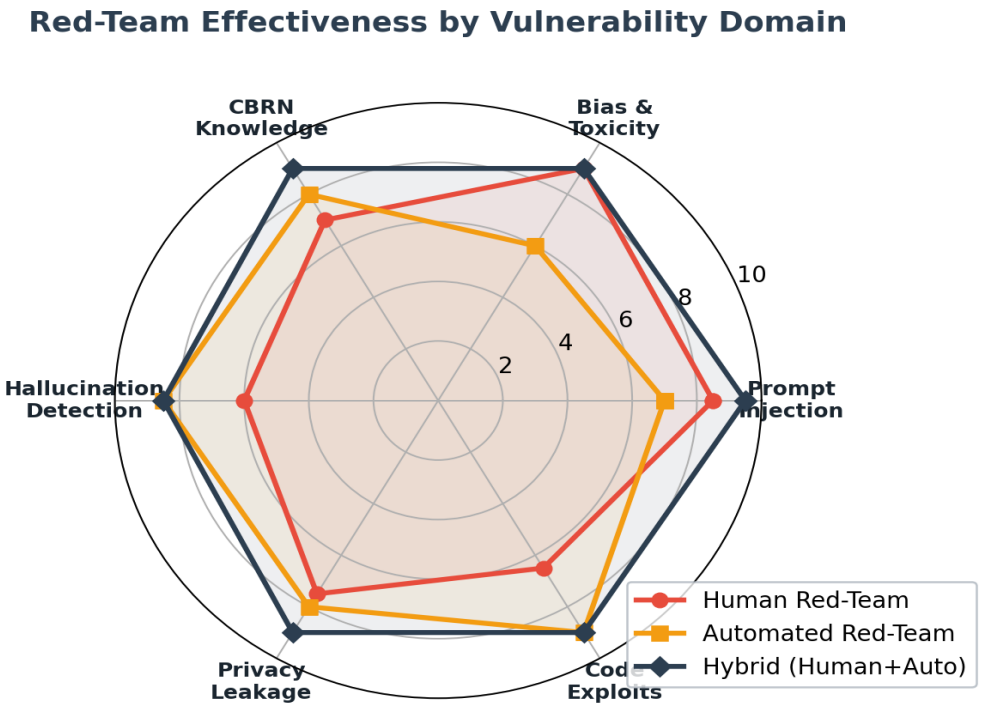



**Figure 3.** *Red-team effectiveness comparison across six vulnerability domains: human, automated, and hybrid approaches*

**Table II.** *AI Red-Teaming Tooling Stack (2025)*

| Tool | Developer | License | Primary Capability | Maturity |
|---|---|---|---|---|
| **PyRIT** | Microsoft | MIT | Multi-model attack orchestration; multi-turn; scoring | Production |
| **Garak** | NVIDIA | Apache 2.0 | Plugin-based vulnerability scanning; 50+ probes | Production |
| **DeepTeam** | Confident AI | Apache 2.0 | Red-team metrics; 40+ attack types; CI integration | Stable |
| **Promptfoo** | Promptfoo | MIT | LLM evaluation framework with red-team modules | Production |
| **HarmBench** | CMU/Stanford | MIT | Standardized jailbreak benchmarking; 400+ behaviors | Research |
| **JailbreakBench** | UC Berkeley | MIT | Open robustness benchmark; curated attack datasets | Research |
| **Inspect Evals** | UK AISI | MIT | Government safety evaluation framework | Stable |
| **Adversa Red** | Adversa AI | Commercial | Enterprise SaaS; compliance mapping; dashboards | Production |

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

## III. CONSTITUTIONAL AI EVALUATION

### 3.1 Constitutional AI Framework

Constitutional AI (CAI), introduced by Anthropic in December 2022, represents a paradigm shift in AI alignment methodology. Rather than relying exclusively on human feedback labels to identify harmful outputs-as in traditional Reinforcement Learning from Human Feedback (RLHF)-CAI uses a set of principles (the "constitution") to enable AI self-evaluation and self-improvement. The training process operates in two phases. In the supervised learning phase, a helpful-only base model generates responses, then self-critiques those responses against constitutional principles, and generates revised responses that are used to fine-tune the model. In the reinforcement learning phase, the fine-tuned model generates response pairs, an AI preference model evaluates which response better adheres to constitutional principles, and this AI-generated preference data trains a reward model for RL optimization (RLAIF - Reinforcement Learning from AI Feedback).

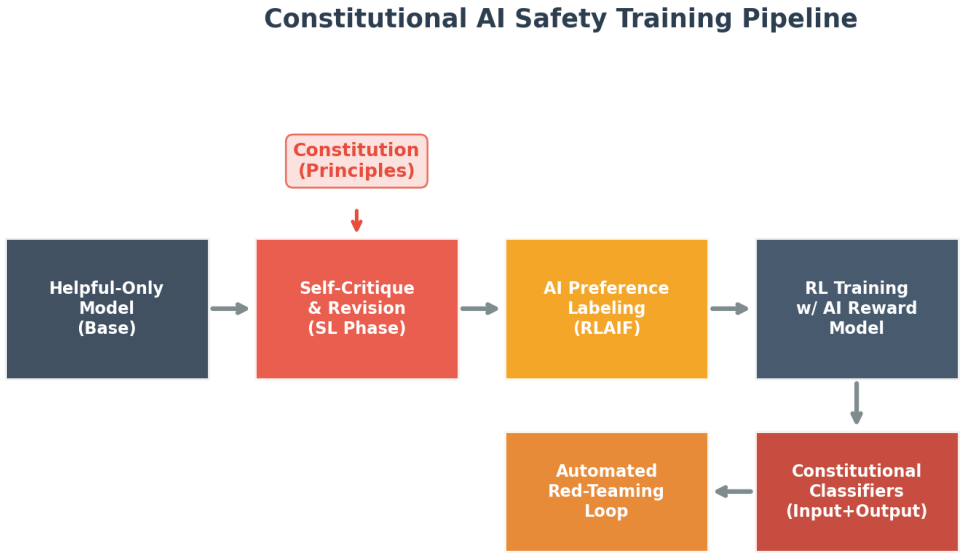



**Figure 4.** *Constitutional AI training pipeline: from base model through self-critique, RLAIF, and Constitutional Classifier deployment*

The constitutional principles themselves encode behavioral guidelines in natural language-for example, "Please choose the response that is most helpful, honest, and harmless" or "Choose the response that is least likely to be used for harmful purposes." This approach offers three critical advantages over pure RLHF. First, scalability: AI-generated preference labels are dramatically cheaper and faster than human annotations. Second, transparency: the constitutional principles are explicit, auditable, and version-controllable. Third, adaptability: updating safety behavior requires modifying principles rather than collecting new human feedback datasets.

**Table III.** *Representative Constitutional Principles and Their Safety Domains*

| Principle Category | Example Principle | Target Harm | Evaluation Method |
|---|---|---|---|
| **Harmlessness** | Choose the response least likely to cause physical harm | Violence, self-harm, CBRN | AI critique + human validation |
| **Honesty** | Choose the response that is most truthful and accurate | Hallucination, deception | Factual grounding evaluation |
| **Helpfulness** | Choose the response that best addresses the user's need | Over-refusal, evasiveness | Helpfulness scoring |
| **Fairness** | Choose the response free from | Discrimination, | Bias benchmark |

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

| | | | |
|---|---|---|---|
| | demographic bias | stereotyping | evaluation |
| **Privacy** | Choose the response that protects personal information | Data leakage, PII exposure | PII detection classifiers |
| **Legality** | Choose the response that does not facilitate illegal activity | Criminal assistance, fraud | Legal domain expert review |

### 3.2 Constitutional Classifiers

Constitutional Classifiers represent the production-deployment evolution of the CAI framework. Rather than relying solely on training-time alignment (which can be bypassed by sufficiently sophisticated jailbreaks), Constitutional Classifiers add runtime defense layers that monitor both inputs and outputs in real time. The first generation, deployed in early 2025, used classifiers trained on synthetic data generated from constitutional principles-including synthetic jailbreak prompts spanning multiple languages, encoding schemes, and obfuscation techniques. These classifiers reduced jailbreak success from 86% to 4.4%, blocking 95% of attacks that would bypass training-time safety.

The second generation, Constitutional Classifiers++ (mid-2025), introduced a two-stage ensemble architecture. A lightweight activation probe examines the model's internal representations to screen all traffic at minimal computational cost. Only queries flagged by the probe are routed to a more comprehensive classifier for detailed analysis. This architecture achieves comparable or superior detection rates while reducing compute overhead from 23.7% (first generation) to approximately 1%-a critical improvement for production deployments where cost efficiency matters. Anthropic reports that over 1,700 cumulative hours of expert red-teaming across 198,000 attempts against Constitutional Classifiers++ yielded only one high-risk vulnerability, with no universal jailbreaks discovered.

**KEY FINDING**

Constitutional Classifiers++ achieve a detection rate of 0.005 high-risk vulnerabilities per thousand queries-the lowest of any defense technique evaluated to date-while adding only ~1% computational overhead, making them viable for cost-sensitive production deployments.

**Table IV.** Defense Layer Comparison: Training-Time vs Runtime Safety

| Defense | Type | Jailbreak Block Rate | Compute Overhead | Over-Refusal Rate | Universal JB Found |
|---|---|---|---|---|---|
| **Base RLHF** | Training | ~14% | Baseline (0%) | Low (0.1%) | Multiple |
| **CAI + RLHF** | Training | ~65% | Baseline (0%) | Moderate (0.5%) | Several |
| **Constitutional Classifier v1** | Runtime | ~95% | +23.7% | 0.38% | 1 found in bounty |
| **Constitutional Classifier++** | Runtime | ~95%+ | +1% | <0.2% | 0 found (198K attempts) |
| **Input + Output Ensemble** | Both | ~97% | +2–5% | <0.3% | Testing ongoing |

## IV. AUTOMATED JAILBREAK DETECTION

Production AI deployments require real-time jailbreak detection systems that can classify inputs and outputs at the speed of inference without introducing prohibitive latency. We evaluate six categories of automated detection approaches, analyzing their accuracy, latency, computational cost, and suitability for different deployment contexts.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

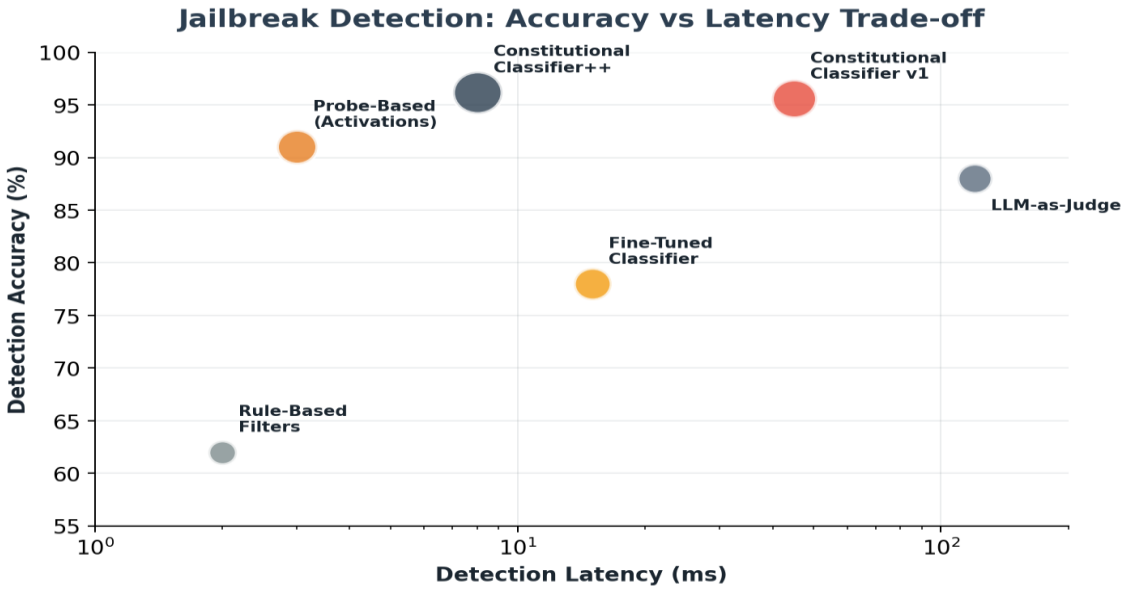



**Figure 5.** Jailbreak detection accuracy vs latency trade-off across six detection methods (bubble size indicates maturity)

**Table V.** Automated Jailbreak Detection System Comparison

| Method | Accuracy (%) | Latency (ms) | Cost/1M Queries | Strengths | Limitations |
|---|---|---|---|---|---|
| **Rule-Based Filters** | 60–65 | 1–3 | $0.10 | Ultra-fast; zero ML cost | Rigid; easily bypassed by paraphrase |
| **Fine-Tuned Classifier** | 75–82 | 10–20 | $2–5 | Good accuracy; moderate cost | Requires labeled training data |
| **Activation Probe** | 88–92 | 2–5 | $0.50 | Fast; leverages internal states | Requires model access; less portable |
| **Constitutional Classifier v1** | 94–96 | 30–60 | $15–25 | High accuracy; principle-based | Higher compute cost; latency |
| **Constitutional Classifier++** | 95–97 | 5–10 | $1–3 | Best accuracy-cost ratio | Requires ensemble infrastructure |
| **LLM-as-Judge** | 85–90 | 80–150 | $30–50 | Flexible; no training needed | Slowest; expensive; non-deterministic |

### 4.1 Emerging Detection Architectures

The most promising direction in jailbreak detection combines multiple approaches in layered architectures. The Constitutional Classifiers++ ensemble-using a fast activation probe for initial screening followed by a comprehensive classifier for flagged queries-demonstrates that intelligent routing can achieve near-optimal accuracy while maintaining production-viable latency. This pattern is generalizable: organizations can deploy rule-based filters as a first layer (catching obvious attacks at near-zero cost), activation probes as a second layer (flagging suspicious patterns with minimal latency), and LLM-based evaluation as a final layer (providing nuanced assessment for the most ambiguous cases). This defense-in-depth approach mirrors established cybersecurity architecture and provides resilience against the evolving landscape of adversarial techniques.

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

## V. RISK-STRATIFIED DEPLOYMENT FRAMEWORK

Enterprise and government deployments of LLMs vary dramatically in risk profile. A customer service chatbot answering product questions presents fundamentally different safety requirements than a medical diagnosis assistant or a defense intelligence analysis system. We propose a four-tier deployment framework that maps safety evaluation intensity to application criticality, ensuring that organizations invest safety resources proportionally to actual risk.

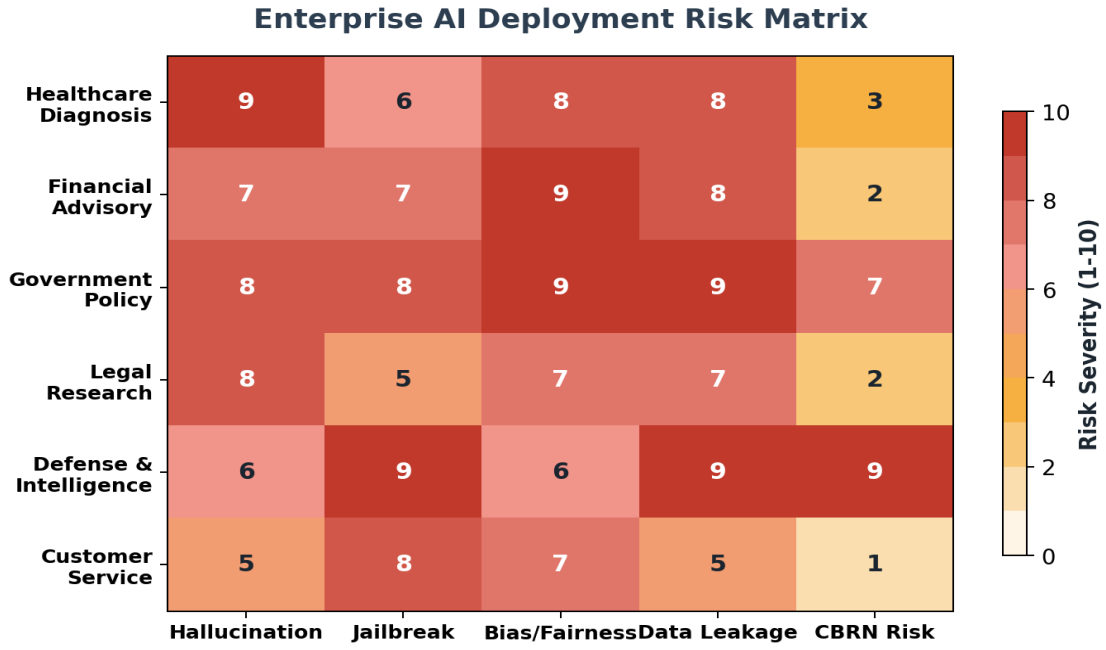



**Figure 6.** Enterprise AI deployment risk matrix: severity scores (1–10) across six sectors and five risk categories

**Table VI.** Risk-Stratified AI Deployment Framework

| Tier | Risk Level | Example Applications | Required Safety Stack | Red-Team Frequency |
|---|---|---|---|---|
| **T1** | Low | FAQ bots, content summary, search | Rule-based filters + basic monitoring | Quarterly automated |
| **T2** | Medium | Customer service, code assist, analytics | CAI + fine-tuned classifiers + logging | Monthly automated + semi-annual human |
| **T3** | High | Healthcare, financial, legal, HR decisions | Constitutional Classifiers + HITL gates | Continuous automated + quarterly expert |
| **T4** | Critical | Defense, intelligence, CBRN, infrastructure | Full CC++ ensemble + air-gapped deploy | Continuous + dedicated red-team unit |

**Table VII.** Regulatory and Standards Alignment for AI Safety Evaluation

| Standard/Regulation | Jurisdiction | AI Safety Requirements | Red-Team Relevance |
|---|---|---|---|
| **EU AI Act** | European Union | Mandatory risk assessment for high-risk AI systems | Art. 9 requires adversarial testing for high-risk |
| **NIST AI RMF** | United States | Voluntary framework for AI risk management | MAP, MEASURE, MANAGE functions align with red-teaming |

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

| | | | |
|---|---|---|---|
| **OWASP Top 10 LLM** | Global | Top 10 vulnerability categories for LLM applications | Direct mapping to red-team test categories |
| **EO 14110** | United States | Executive Order on Safe AI; red-team requirements | Mandates red-teaming for dual-use foundation models |
| **ML Commons AI Safety** | Global | Standardized safety benchmarks (v0.5 proof of concept) | Provides benchmark suite for automated red-teaming |
| **UK AISI Inspect** | United Kingdom | Government safety evaluation framework | Open-source eval framework for frontier model testing |

## VI. OPEN CHALLENGES AND FUTURE DIRECTIONS

Despite significant progress in AI safety evaluation, twelve critical challenges remain unresolved. These challenges span technical, organizational, and regulatory dimensions and will shape the research agenda for the field over the coming years.

**Table VIII.** Twelve Open Research Challenges in Deployed AI Safety

| # | Challenge | Current Status | Research Direction |
|---|---|---|---|
| **1** | Multimodal jailbreaks (image, audio, video) | Early research; image attacks demonstrated | Cross-modal safety classifiers; unified input screening |
| **2** | Multi-turn attack persistence | Crescendo shows effectiveness; defenses lag | Conversation-level state tracking; turn-by-turn scoring |
| **3** | Agentic AI safety (tool-use + autonomy) | MCP security emerging; tool poisoning documented | Agent-aware safety frameworks; action-level HITL gates |
| **4** | Non-English and low-resource language safety | Safety training concentrated in English | Multilingual constitutional principles; cross-lingual transfer |
| **5** | Measuring over-refusal impact on utility | CC v1 showed 0.38% over-refusal; CC++ reduces this | Calibrated refusal thresholds; context-dependent safety |
| **6** | Adversarial robustness certification | No formal guarantees possible for LLM safety | Probabilistic safety bounds; formal verification research |
| **7** | Safety evaluation for reasoning models (o1-class) | Extended thinking complicates monitoring | Chain-of-thought safety monitoring; reasoning auditing |
| **8** | Supply chain safety for fine-tuned models | Backdoor and sleeper agent risks documented | Model provenance verification; activation scanning |
| **9** | Standardized red-team reporting formats | No industry standard; high variance in practices | Common vulnerability scoring; machine-readable reports |
| **10** | Real-time safety adaptation | Static classifiers cannot adapt to novel attacks | Online learning; rapid classifier update pipelines |
| **11** | Safety-capability trade-off quantification | Anecdotal evidence of safety-helpfulness tension | Pareto frontier analysis; multi-objective optimization |
| **12** | Democratic governance of safety principles | Anthropic's Collective Constitutional AI experiment | Participatory constitution design; stakeholder alignment |

| e-ISSN: 2278 – 8875, p-ISSN: 2320 – 3765| www.ijareeie.com | Impact Factor: 8.807 | A Monthly Peer Reviewed & Refereed Journal |

|| Volume 14, Issue 8, August 2025 ||

| DOI:10.15662/IJAREEIE.2025.1408015 |

## VII. CONCLUSION

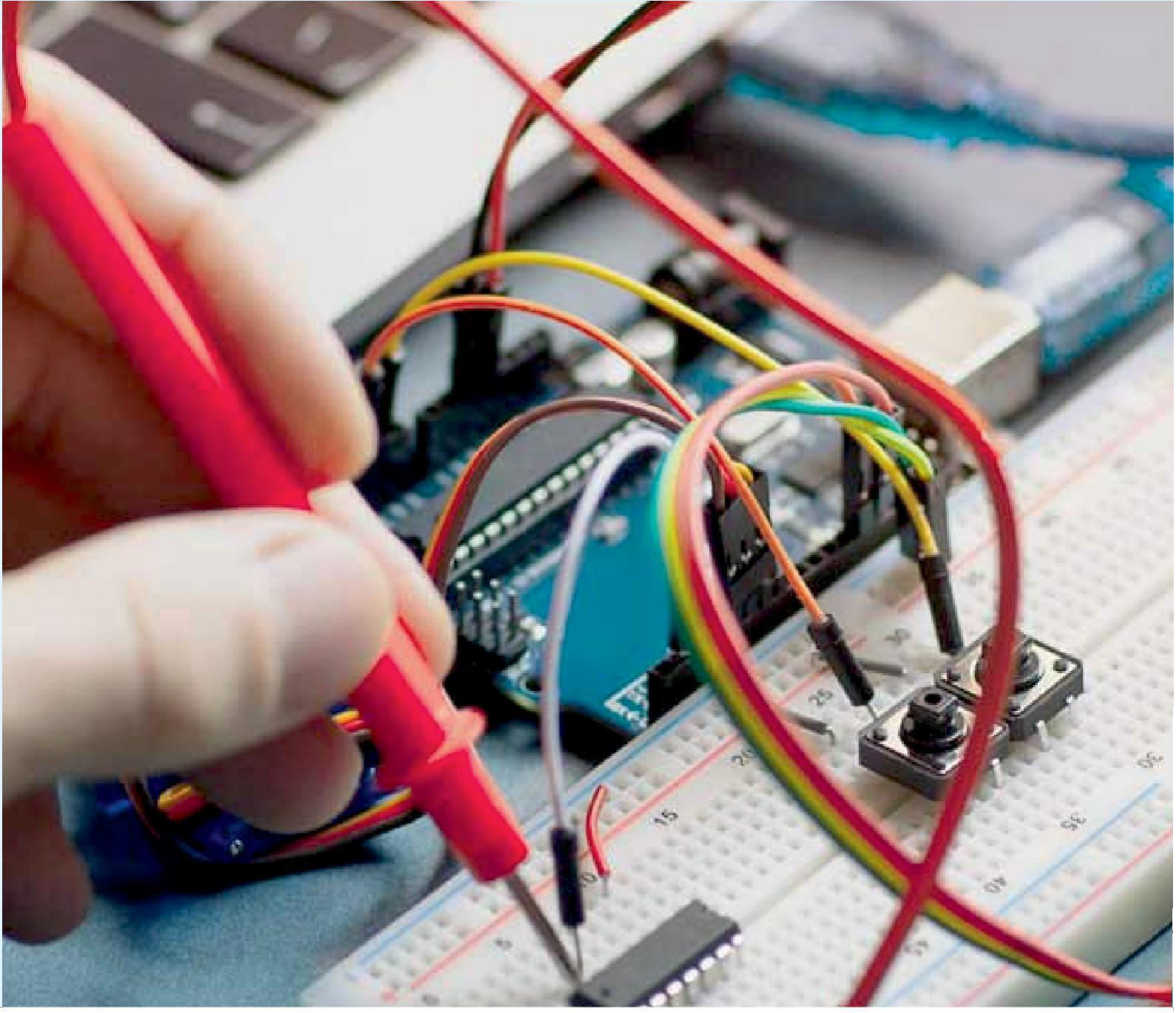
This study presents a comprehensive analysis of AI safety evaluation for deployed LLM systems, synthesizing research across red-teaming techniques, Constitutional AI evaluation, and automated jailbreak detection. Our analysis reveals that the field has made remarkable progress: Constitutional Classifiers++ can block 95%+ of jailbreak attacks with only ~1% compute overhead, hybrid red-teaming approaches combining human expertise with automated scalability provide the most comprehensive vulnerability coverage, and defense-in-depth architectures layering multiple detection methods can achieve production-viable accuracy-latency profiles.

However, the fundamental challenge of AI safety remains unsolved. The adversarial landscape continues to evolve, with novel attack techniques-multimodal jailbreaks, multi-turn escalation, agentic exploitation, and reasoning-chain manipulation-consistently outpacing static defense systems. The non-deterministic nature of LLM behavior means that no defense can provide formal safety guarantees. Instead, deployed AI safety must be understood as a continuous process of evaluation, defense, and adaptation-analogous to cybersecurity rather than quality assurance.

For enterprise and government organizations deploying LLMs in high-stakes applications, we recommend the following: adopt the risk-stratified deployment framework presented in Section V, mapping safety investment to application criticality; implement defense-in-depth jailbreak detection combining fast activation probes with comprehensive Constitutional Classifiers; establish continuous red-teaming programs using hybrid human-automated approaches; align safety evaluation practices with emerging regulatory requirements (EU AI Act, NIST AI RMF, EO 14110); and invest in safety monitoring infrastructure that enables rapid response to newly discovered vulnerabilities. The organizations that treat AI safety as a first-class engineering discipline-rather than a compliance checkbox-will be best positioned to deploy trustworthy AI at enterprise scale.

## REFERENCES

[1] Y. Bai et al., "Constitutional AI: Harmlessness from AI Feedback," arXiv:2212.08073, Dec. 2022.
[2] E. Perez et al., "Red Teaming Language Models with Language Models," arXiv:2202.03286, 2022.
[3] Anthropic, "Constitutional Classifiers: Defending Against Universal Jailbreaks," arXiv:2501.18837, Jan. 2025.
[4] Anthropic, "Next-Generation Constitutional Classifiers," Anthropic Research, 2025.
[5] P. Chao et al., "Jailbreaking Black Box Large Language Models in Twenty Queries (PAIR)," arXiv:2310.08419, 2024.
[6] X. Liu et al., "AutoDAN-Turbo: A Lifelong Agent for Strategy Self-Exploration to Jailbreak LLMs," ICLR 2025.
[7] M. Mazeika et al., "HarmBench: A Standardized Evaluation Framework for Automated Red Teaming," arXiv:2402.04249, 2024.
[8] P. Chao et al., "JailbreakBench: An Open Robustness Benchmark for Jailbreaking LLMs," arXiv:2404.01318, 2024.
[9] OpenAI, "OpenAI's Approach to External Red Teaming for AI Models and Systems," OpenAI White Paper, 2024.
[10] Microsoft, "Enhancing AI Safety: Insights and Lessons from Red Teaming," Microsoft Cloud Blog, Jan. 2025.
[11] OWASP, "OWASP Top 10 for LLM Applications 2025," Open Web Application Security Project, 2025.
[12] Japan AI Safety Institute, "Guide to Red Teaming Methodology on AI Safety, Version 1.10," AISI, Mar. 2025.
[13] A. V. Prabhakar, "Red Teaming LLM: Playbook for Secure GenAI Deployment," Jul. 2025.
[14] S. S. Feffer et al., "Red-Teaming for Generative AI: Silver Bullet or Security Theater?," arXiv:2401.15897, 2024.
[15] ML Commons, "AI Safety v0.5 Proof of Concept Benchmark," ML Commons, 2024.
[16] A. Zou et al., "Universal and Transferable Adversarial Attacks on Aligned Language Models (GCG)," arXiv:2307.15043, 2023.
[17] UK AI Safety Institute, "Inspect Evals: Open-Source Evaluation Framework," UK AISI, 2025.
[18] CSET Georgetown, "AI Red-Teaming Design: Threat Models and Tools," Center for Security and Emerging Technology, 2025.



INNO SPACE
SJIF Scientific Journal Impact Factor

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

# International Journal of Advanced Research

## in Electrical, Electronics and Instrumentation Engineering

9940 572 462 6381 907 438 ijareeie@gmail.com



Scan to save the contact details