CORRELATED SYSTEMIC RISK IN LARGE-SCALE AI DEPLOYMENT

An Evidence Synthesis of Dyadic Human–AI Architectures Without Independent Verification Layers

Author: Brian K Rasmussen

Retrieval-Enabled Research Synthesis • February 2026

EVIDENCE CLASSIFICATION SYSTEM

[EMPIRICAL] = Sourced to measured data or verified incident
[THEORETICAL] = Formal proof or logical argument without direct measurement
[UNCERTAINTY] = Evidence is incomplete, contested, or non-transferable

This document contains no speculative projections, no policy advocacy, and no invented statistics.

I. DEFINITIONS

1.1 Dyadic Architecture

A system configuration in which a single human user interacts with a single AI model instance, without a structurally independent verification, audit, or retrieval-gating layer between the model's output and the user's consumption of that output. This is the default deployment architecture of consumer-facing LLM products and many enterprise integrations.

1.2 Independent Verification Layer

A component that evaluates model outputs using processes, data, or computational substrates that are architecturally distinct from the generating model. Examples include: separate audit models with different training distributions; retrieval-augmented generation (RAG) systems that ground outputs against external corpora; multi-agent verification where independent model instances challenge outputs; and non-LLM classifiers (e.g., fine-tuned BERT models) that evaluate outputs using different architectures.

1.3 Correlated Failure

A failure mode in which a single cause produces simultaneous failures across multiple system instances, in contrast to independent failures where each instance fails for unrelated reasons. Correlated failures are more dangerous than independent failures because they defeat redundancy: N copies of the same system do not provide N-fold safety improvement if all copies fail from the same cause.

1.4 Structural Limitation vs. Moral Failure

A structural limitation is a failure mode arising from system architecture rather than negligent or malicious human decisions. It persists even when all parties act competently and in good faith. A moral failure is attributable to identifiable actors who had the knowledge, resources, and opportunity to prevent harm but did not. The distinction is operationally important: structural limitations require architectural remedies; moral failures require accountability mechanisms. Many real-world incidents involve both.

II. EMPIRICAL EVIDENCE OF CORRELATED SOFTWARE FAILURES

This section documents verified cases where a single software fault caused simultaneous, correlated failures across multiple deployed instances. These cases are drawn from aviation, automotive, and cloud computing domains to establish that correlated software failure is an empirically observed phenomenon, not a theoretical concern.

2.1 Aviation: Boeing 737 MAX MCAS

> **[EMPIRICAL]** The Boeing 737 MAX's Maneuvering Characteristics Augmentation System (MCAS) relied on data from a single Angle of Attack (AoA) sensor to trigger automatic nose-down stabilizer trim [1]. When the AoA sensor provided erroneous data, MCAS repeatedly forced the aircraft's nose down. This identical failure mode caused two crashes: Lion Air Flight 610 (October 29, 2018, 189 fatalities) and Ethiopian Airlines Flight 302 (March 10, 2019, 157 fatalities) [1, 2]. Total: 346 deaths from a single software design flaw replicated across the fleet.

The system's structural deficiency was the absence of independent verification: MCAS accepted a single sensor input without cross-checking against the second available AoA sensor. The two aircraft that crashed did not contain the optional disagree light that would have alerted pilots to conflicting sensor readings [2]. Boeing's safety analysis assumed pilots would serve as the backstop if MCAS malfunctioned, but pilots were not informed of MCAS's existence [1, 3]. After the crashes, the redesigned MCAS cross-checks both AoA sensors—an independent verification layer that was absent in the original design [1].

This case is directly analogous to dyadic AI architectures: a single computational system made consequential decisions based on a single data source without independent verification, and the same flaw caused identical failures across every deployed instance.

2.2 Cloud Computing: CrowdStrike Incident (July 19, 2024)

> **[EMPIRICAL]** On July 19, 2024, CrowdStrike distributed a faulty configuration update (Channel File 291) to its Falcon Sensor security software. The update contained a logic error caused by a mismatch between the number of input fields in the IPC Template Type (21 fields) and the inputs provided by the sensor code (20 fields), combined with a missing runtime array bounds check [4]. Approximately 8.5 million Windows systems crashed globally, representing less than 1% of all Windows systems but causing disruptions across airlines, banks, hospitals, emergency services, and government operations [4, 5].
> **[EMPIRICAL]** Estimated financial damage: Fortune 500 companies alone faced losses exceeding $5 billion in direct losses. Healthcare and banking sectors were hardest hit, with estimated losses of $1.94 billion and $1.15 billion respectively. Delta Air Lines claimed approximately $500 million in damages and disruption of 7,000 flights affecting 1.3 million customers over five days [5, 6].

This incident demonstrates perfect correlated failure: a single code defect propagated simultaneously to all subscribers because CrowdStrike's software did not provide a mechanism for

subscribers to delay installation of content updates [4]. The failure was identical across all affected systems. Recovery required manual intervention on each individual machine, taking organizations days to fully restore operations [4].

The former general counsel of the NSA stated after the incident that such cascading outages would continue because of the "underlying reality that everything is extraordinarily interconnected" [7]. The Congressional Research Service noted the incident highlighted "risks related to market domination by a few providers and reliance on cloud-services," since CrowdStrike maintained approximately one-fifth of endpoint security market share [8].

2.3 Automotive: Tesla Autopilot Software Recalls

> **[EMPIRICAL]** NHTSA's investigation into Tesla's Autopilot system (EA22002) reviewed 956 total crashes where Autopilot was alleged to have been in use between January 2018 and August 2023. Of these, 29 were fatal. NHTSA found that "drivers involved in the crashes were not sufficiently engaged in the driving task and that the warnings provided by Autopilot when Autosteer was engaged did not adequately ensure that drivers maintained their attention" [9].
> **[EMPIRICAL]** In December 2023, Tesla filed Recall 23V838, applicable to all Tesla models equipped with any version of Autopilot—approximately 2 million vehicles. Tesla's filing stated that "the prominence and scope of the system's controls may be insufficient for a driver assistance system that requires constant supervision by a human driver" [10]. Following the recall remedy (an OTA software update), NHTSA identified an additional 13 fatal crashes in which foreseeable driver misuse appeared to play a role, and opened a further investigation into the remedy's effectiveness [11].

Tesla's largest recall to date (24V051000) affected an estimated 2,193,869 vehicles for an incorrect font size on instrument panel warning indicators [12]. The recall was addressed via OTA software update, demonstrating that in software-defined vehicles, a single code change can simultaneously affect millions of units—the same correlated-failure property observed in the CrowdStrike incident.

2.4 Pattern Summary

Three properties are consistent across these cases: (a) a single point of failure in software or configuration; (b) simultaneous propagation to all deployed instances; (c) the absence or inadequacy of an independent verification layer. The MCAS system lacked sensor cross-checking. The CrowdStrike update lacked staged deployment or subscriber-controlled gating. Tesla's OTA update architecture means a single software change affects millions of vehicles simultaneously. In each case, the correlated nature of the failure converted what would have been isolated incidents into systemic events.

III. AI-SPECIFIC EVIDENCE

This section reviews primary research on failure modes specific to AI systems, focusing on empirically measured effects rather than theoretical projections.

3.1 Guardrail Bypass Research

> **[EMPIRICAL]** Hackett et al. (2025) conducted the first systematic empirical analysis of evasion attacks against six prominent LLM guardrail systems, including Microsoft's Azure Prompt Shield, Meta's Prompt Guard, NVIDIA's NeMo Guard, and Protect AI. Results: character injection techniques (zero-width characters, Unicode tags, homoglyphs, emoji smuggling) achieved up to 100% evasion success against multiple guardrails. Some attacks fully bypassed all detection across several systems [13]. Responsible disclosure was completed April 2025 with agreement from all parties.
> **[EMPIRICAL]** HiddenLayer (April 2025) disclosed "Policy Puppetry," described as the first post-instruction-hierarchy alignment bypass that works across virtually all frontier AI models. The technique reformulates prompts to resemble policy configuration files (XML, JSON, INI), bypassing safety

alignment across models and organizations [14]. HiddenLayer describes the technique as "easy to adapt, highly scalable, and difficult to patch" because it "exploits a systemic weakness in how many LLMs are trained on instruction or policy-related data" [14].
[EMPIRICAL] In October 2025, HiddenLayer demonstrated that OpenAI's Guardrails framework (released October 6, 2025), which uses an LLM "judge" to evaluate inputs and outputs, could be bypassed through prompt injection that manipulates the judge's confidence scoring. Because both the generating model and the safety judge are LLM-based, prompt injection that affects the base model also affects the judge—a "cascade failure where the security mechanism becomes part of the attack vector" [15].

Relevance to correlated failure: guardrail bypasses are correlated across all instances of the same model. A technique that bypasses GPT-4's safety alignment works on every GPT-4 instance simultaneously. Unlike hardware failures (which are independently distributed across units), software vulnerabilities are perfectly correlated across all deployments of the same software.

3.2 Hallucination Persistence

[THEORETICAL] Xu et al. (2024) provided a formal proof that hallucination is mathematically inevitable for LLMs used as general problem solvers. Using results from learning theory, they demonstrate that LLMs cannot learn all computable functions and will therefore produce outputs inconsistent with ground truth [16]. This is a theoretical result establishing a lower bound on hallucination, not a measurement of hallucination rates.

[EMPIRICAL] OpenAI (2025) published "Why Language Models Hallucinate," arguing that hallucinations persist because current training objectives and evaluation benchmarks reward confident guessing over calibrated uncertainty. A model that guesses has a nonzero probability of being scored correct; a model that says "I don't know" scores zero on accuracy metrics [17].
[EMPIRICAL] AIMultiple benchmarked 37 LLMs with 60 questions and found that even latest-generation models exhibit hallucination rates exceeding 15% when analyzing provided statements [18]. This is a measured rate on a specific benchmark; hallucination rates on other tasks and benchmarks may differ substantially.
[EMPIRICAL] Anh-Hoang et al. (2025, Frontiers in AI) introduced a probabilistic attribution framework classifying hallucinations as prompt-dominant or model-dominant. Key finding: some models (e.g., DeepSeek 67B) show low Prompt Sensitivity but high Model Variability, meaning hallucinations persist regardless of prompt structure—indicating fundamental model-intrinsic limitations [19].
[EMPIRICAL] A Scientific Reports (2025) study analyzing 3 million user reviews from 90 AI-powered mobile apps estimated that approximately 1.75% of reviews initially flagged as relevant contained indicators of user-reported hallucination experiences [20]. This is a lower bound: users can only report hallucinations they detect.

[UNCERTAINTY] *Hallucination rates are task-dependent, model-dependent, and configuration-dependent. The 15% figure applies to one specific benchmark. The 1.75% user-report figure is a lower bound on user-detected hallucinations in one app category. No single number characterizes a model's hallucination rate across all conditions.*

3.3 Epistemic Authority and Overreliance

[EMPIRICAL] A 2025 study in Economic Computation and Economic Cybernetics Studies and Research documents "confidence inflation": repeated exposure to confidently presented LLM outputs recalibrates human epistemic standards through heuristic reliance on fluency, reinforcement of implicit trust, authority transfer, and error blindness [21].
[EMPIRICAL] Pelrine et al. (2024–2025) measured epistemic miscalibration in LLMs, finding that models frequently express high external linguistic assertiveness even when their internal uncertainty (measured by token probability distributions) is high [22]. This represents a measurable gap between internal model state and external expression.
[EMPIRICAL] Research on AI-assisted decision-making found that human-AI teams sometimes perform worse than AI alone because humans either follow incorrect AI advice or ignore correct AI

> advice. Displaying reliable uncertainty estimations can help humans recognize error boundaries and calibrate trust [23].

Relevance to correlated risk: overreliance is not independently distributed across users. If a model's confidence calibration is systematically miscalibrated (expressing certainty about claims where it should express uncertainty), every user of that model receives the same miscalibrated signal. This creates a correlated epistemic failure analogous to correlated technical failures.

IV. SCALING RISK ANALYSIS

This section examines what changes when AI deployment scales, drawing strictly on documented evidence and sourced analogies. Where evidence is insufficient to support quantitative claims, this is stated explicitly.

4.1 Software Failures Are Correlated; Hardware Failures Are Not

A central finding from the empirical evidence in Section II is that software failures exhibit perfect correlation across deployed instances, whereas hardware failures are independently distributed. When CrowdStrike's Channel File 291 contained a logic error, every system that received the update failed identically [4]. When a brake pad on a single vehicle wears out, other vehicles' brake pads are unaffected.

This distinction is critical for AI systems because AI failures are software failures. A hallucination pattern, a guardrail bypass, or a confidence miscalibration in a given model version affects every instance of that model simultaneously. Scaling the number of instances does not diversify risk; it amplifies exposure to correlated failures.

4.2 Documented Incident Volume at Current Scale

> **[EMPIRICAL]** The Stanford AI Index Report 2025 documented 233 AI safety incidents in 2024, a 56.4% increase from 149 incidents in 2023 [24]. This is the highest annual total recorded. While not all incidents involved dyadic LLM systems, the growth trajectory reflects expanding deployment.
> **[EMPIRICAL]** Adversa AI's 2025 report found that 35% of all real-world AI security incidents were caused by simple prompts, with some leading to losses exceeding $100,000 without writing a single line of code [25].
> **[EMPIRICAL]** ISACA (2025) documented patterns including: wrongful arrests linked to facial recognition with misplaced certainty, young people turning to chatbots for emotional support with fatal outcomes, a hiring platform exposing 64 million application records through default credentials, and chatbots providing confident but incorrect advice [26].

4.3 Does AI Introduce Higher Correlated Failure Risk Than Other Software?

The evidence suggests two properties that may make AI systems more prone to correlated failure than conventional software:

Non-determinism in outputs: Conventional software bugs are deterministic (the same input always produces the same erroneous output). AI system failures are stochastic (the same input may produce different outputs depending on sampling), making them harder to detect through conventional testing but no less correlated when they manifest as systematic biases or calibration errors [17, 19].

Self-referential safety architecture: When LLM-based safety judges are used to monitor LLM-based generators, the safety system inherits the generator's vulnerability class. This is empirically

demonstrated by the OpenAI Guardrails bypass [15]. Conventional software safety systems (watchdog timers, hardware interlocks) typically use architecturally distinct mechanisms.

> **[UNCERTAINTY]** *Whether AI systems exhibit quantitatively higher correlated failure rates than conventional enterprise software has not been empirically measured in a controlled comparative study. The evidence supports a qualitative argument that the mechanisms are present, but no peer-reviewed study directly measures the comparative rate.*

V. MITIGATION ANALYSIS

This section reports measured improvements from documented mitigation approaches. Only effect sizes from published research are included.

5.1 Retrieval-Augmented Generation (RAG)

> **[EMPIRICAL]** A NAACL 2024 Industry Track paper demonstrated that RAG "significantly reduces hallucination and allows generalization to out-of-domain settings," and that using a small, well-trained retriever can reduce required LLM size at no loss in performance [27].
> **[EMPIRICAL]** A 2025 meta-analysis reported that hybrid RAG architectures show consistent 35–60% error reduction in hallucination rates. Systems combining RAG with statistical validation (e.g., AWS Bedrock contextual grounding integrated with NVIDIA NeMo guardrails) achieved 97% hallucination detection rates at sub-200ms latency [28].
> **[EMPIRICAL]** Stanford's 2025 legal RAG reliability work found that even well-curated retrieval pipelines can fabricate citations, and the most promising systems now add span-level verification: each generated claim is matched against retrieved evidence and flagged if unsupported [29].
>
> **[UNCERTAINTY]** *The 35–60% error reduction and 97% detection figures come from a meta-analysis of varied experimental conditions (source [28] is not independently peer-reviewed; it is a preprint). Actual reduction in specific deployments depends on retrieval quality, corpus coverage, and task complexity.*

5.2 Multi-Agent Verification

> **[EMPIRICAL]** ACL Findings 2025 demonstrated that Best-of-N reranking—generating multiple candidate responses and selecting the most faithful one using a lightweight factuality metric—significantly lowers hallucination rates without model retraining [29].
> **[EMPIRICAL]** An MDPI Information (2025) review of multi-agent debate and verification frameworks found that these approaches reduce hallucination compared to single-model systems by introducing diversity of perspective and iterative challenge [30].

The structural advantage of multi-agent verification is that it breaks the self-referential circularity of single-model systems. However, if agents share the same training distribution, their errors may be correlated rather than independent, reducing the effective diversity.

5.3 Independent Audit Models

> **[EMPIRICAL]** Non-LLM classifiers (e.g., fine-tuned BERT-based detectors) avoid the vulnerability inheritance problem documented in the OpenAI Guardrails bypass because they are architecturally distinct from the system they monitor [15]. However, Hackett et al. (2025) found these classifiers also have vulnerabilities: they can be evaded through character injection because their training data may not cover the same input space as the LLM [13].

No guardrail system tested in the Hackett et al. study consistently outperformed others across all attack types. Each showed significant weaknesses depending on the technique and threat model applied [13]. This indicates that no single verification layer provides comprehensive protection; layered defenses across architecturally diverse mechanisms are required.

5.4 Regulatory Requirements

> **[EMPIRICAL]** The EU AI Act (effective August 2025 for general-purpose AI obligations) requires human oversight mechanisms for high-risk AI systems [31]. U.S. OMB Memo M-26-04 (December 2025) requires federal agencies purchasing LLMs to request model cards, evaluation artifacts, and acceptable use policies [32]. California SB 53 (signed September 2025) requires frontier developers to publish risk frameworks and report critical safety incidents [33]. Texas TRAIGA (signed June 2025) prohibits AI systems designed for restricted purposes and provides penalties ranging from $10,000 to $200,000 per violation [33].

These frameworks implicitly recognize that self-governance is insufficient by requiring external documentation, testing, and oversight independent of the model's self-assessment.

5.5 Measured Mitigation Summary

Mitigation	Measured Effect	Source	Evidence Quality
RAG (basic)	Significant hallucination reduction (qualitative)	[27] NAACL 2024	Peer-reviewed industry track
RAG + statistical validation	35–60% error reduction; 97% detection	[28] Preprint 2025	Not peer-reviewed
Best-of-N reranking	Significant error reduction	[29] ACL Findings 2025	Peer-reviewed
Multi-agent debate	Reduced hallucination vs single model	[30] MDPI 2025	Peer-reviewed review
Non-LLM classifiers	Architecturally avoids vulnerability inheritance	[13, 15]	Peer-reviewed + industry
Span-level verification	Flags unsupported claims in RAG	[29]	Benchmark (REFIND SemEval 2025)

VI. UNCERTAINTY AND OPEN QUESTIONS

6.1 Where Evidence Is Thin

No peer-reviewed study directly compares correlated failure rates between AI systems and conventional software systems at equivalent deployment scale. The analogy between software correlated failure (CrowdStrike, MCAS) and AI correlated failure (hallucination patterns, guardrail bypasses) is structurally sound but has not been empirically validated through controlled measurement.

Hallucination rates at scale—specifically, how hallucination patterns correlate across millions of concurrent users of the same model—have not been systematically measured. Individual-level hallucination rates are documented; fleet-level correlation structure is not.

The long-term effects of epistemic authority transfer (confidence inflation) have been studied in short-term experimental settings but not in longitudinal studies tracking users' critical evaluation capacity over months or years of daily AI assistant use.

6.2 Where Analogies May Not Transfer

The aviation and automotive analogies in Section II involve deterministic software operating on physical systems with directly measurable consequences. AI systems operating in information and advisory domains produce harms that are harder to measure (epistemic degradation, decision quality reduction, subtle misinformation). The consequence pathways differ: a software bug in MCAS produces immediate physical harm; a systematic hallucination pattern produces diffuse informational harm that may be individually minor but cumulatively significant. Whether the risk management frameworks developed for safety-critical physical systems transfer to information systems is an open question.

Multi-agent verification's effectiveness depends on the independence of errors between agents. If all agents in a verification system are fine-tuned from the same base model or trained on overlapping data, their errors may be correlated rather than independent, reducing the effective safety margin. The degree to which different LLMs produce correlated versus independent errors on the same inputs has not been comprehensively measured.

6.3 What Remains Untested

Fleet-level correlated failure of AI systems: no incident has yet demonstrated simultaneous correlated failure across all instances of a model producing identical harmful outputs to all users at once (comparable to CrowdStrike). Whether this is because such events are structurally unlikely, or because we have not yet observed them, cannot be determined from current evidence.

Effectiveness of layered verification at scale: while RAG, multi-agent verification, and audit models show measured improvements in controlled settings, their effectiveness when deployed across millions of concurrent users in production environments has not been systematically validated in peer-reviewed literature.

Comparative safety: whether AI systems with verification layers achieve safety levels comparable to, better than, or worse than conventional software safety-critical systems (which have decades of established engineering practice) is unknown.

APPENDIX: QUANTITATIVE CLAIM VERIFICATION

Every quantitative claim in this document is listed below with its exact source. Claims that could not be verified against a primary source have been removed from the document.

8.5 million systems crashed (CrowdStrike): Wikipedia citing CrowdStrike post-incident report and Microsoft estimate [4]. Also confirmed by IBM [5], CNN [6], HBR [6], and U.S. Congressional Research Service [8].

$5+ billion in Fortune 500 losses (CrowdStrike): Parametrix analysis cited by CNN [6] and Harvard Business Review [6].

$1.94B healthcare / $1.15B banking losses (CrowdStrike): Parametrix analysis cited by CNN [6].

346 deaths (Boeing 737 MAX): 189 (Lion Air Flight 610) + 157 (Ethiopian Airlines Flight 302). Widely documented including PMC/NIH [2] and FAA review [1].

MCAS relied on single AoA sensor: FAA Summary of Review [1]; PMC case study [2]; Seattle Times investigation [3].

956 crashes reviewed / 29 fatal (Tesla Autopilot): NHTSA INCR-EA22002-14496 report [9].

2 million vehicles recalled (Tesla Autopilot 23V838): NHTSA recall filing and CNBC reporting [10, 11].

2,193,869 vehicles (Tesla recall 24V051): NHTSA recall report 24V051 [12].

100% evasion success (guardrail bypass): Hackett et al. (2025), arXiv:2504.11168, Section 5.1 [13].

233 AI safety incidents in 2024: Stanford AI Index Report 2025, cited by Responsible AI Labs [24].

56.4% increase from 2023: Same source, 149 (2023) to 233 (2024) [24].

35% of AI security incidents from prompts: Adversa AI 2025 report [25].

>15% hallucination rate (frontier models): AIMultiple benchmark of 37 LLMs with 60 questions [18].

1.75% user-reported hallucination rate: Scientific Reports (2025), analysis of 3 million reviews from 90 apps [20].

35–60% error reduction (hybrid RAG): Preprint meta-analysis [28]. NOTE: Not peer-reviewed.

97% detection rate (RAG + statistical validation): Same preprint [28]. NOTE: Not peer-reviewed; not independently replicated.

CONCLUSION: EVIDENCE-SUPPORTED FINDINGS ONLY

Finding 1: Correlated software failure is an empirically documented phenomenon. The CrowdStrike incident (8.5 million simultaneous system failures from one code defect), the Boeing 737 MAX crashes (346 deaths from one software design flaw replicated across the fleet), and Tesla's fleet-wide software recalls demonstrate that software failures propagate simultaneously across all deployed instances.

Finding 2: AI guardrail systems are empirically vulnerable to evasion. Testing of six prominent guardrail systems found up to 100% evasion success rates using character injection techniques. A universal bypass technique (Policy Puppetry) works across virtually all frontier models. LLM-based safety judges inherit the same vulnerability class as the models they monitor.

Finding 3: Hallucination is formally proven to be an inherent property of LLMs used as general problem solvers and empirically measured at rates exceeding 15% on specific benchmarks. Some hallucination patterns are model-intrinsic and persist regardless of prompt design.

Finding 4: Confidence inflation and automation bias are documented in LLM interaction contexts. LLMs exhibit measurable epistemic miscalibration—expressing high linguistic confidence when internal certainty is low. This creates a correlated epistemic failure across all users of the same model.

Finding 5: Independent verification layers (RAG, multi-agent verification, audit models) produce measured improvements: 35–60% error reduction for hybrid RAG (preprint, not peer-reviewed), significant hallucination reduction in peer-reviewed RAG studies, and architectural disruption of vulnerability inheritance with non-LLM classifiers. No verification approach achieves comprehensive protection.

Finding 6: Whether large-scale deployment of AI systems without independent verification layers introduces correlated systemic risk comparable to the documented cases in aviation, automotive, and cloud computing cannot be definitively answered with current evidence. The structural mechanisms for correlated failure are present and empirically documented (software vulnerability correlation, shared training distributions, self-referential safety architecture). The quantitative magnitude of the risk at scale remains unmeasured.

This document does not advocate for specific policies. The evidence presented is intended to inform risk assessment.

SOURCE LIST

[1] FAA. Summary of the FAA's Review of the Boeing 737 MAX. https://www.faa.gov/sites/faa.gov/files/2022-08/737_RTS_Summary.pdf

[2] Herkert et al. (2020). The Boeing 737 MAX: Lessons for Engineering Ethics. PMC. https://pmc.ncbi.nlm.nih.gov/articles/PMC7351545/

[3] Gates (2019). The Inside Story of MCAS. Seattle Times / AFA-CWA. https://afacwa.org/the_inside_story_of_mcas_seattle_times/

[4] 2024 CrowdStrike-related IT outages. Wikipedia. https://en.wikipedia.org/wiki/2024_CrowdStrike-related_IT_outages

[5] IBM (2024). Recent CrowdStrike Outage: What You Should Know. https://www.ibm.com/think/news/recent-crowdstrike-outage-what-you-should-know

[6] CNN (2024). CrowdStrike Outage Cost and Cause. https://www.cnn.com/2024/07/24/tech/crowdstrike-outage-cost-cause; HBR (2025). What the 2024 CrowdStrike Glitch Can Teach Us About Cyber Risk. https://hbr.org/2025/01/what-the-2024-crowdstrike-glitch-can-teach-us-about-cyber-risk

[7] NPR (2024). The CrowdStrike Outage Showed the Vulnerability of the Cloud. https://www.npr.org/2024/07/27/nx-s1-5049863/

[8] Congressional Research Service. The July 19th Global IT Outages. https://www.congress.gov/crs-product/IN12392

[9] NHTSA. Additional Information Regarding EA22002. https://static.nhtsa.gov/odi/inv/2022/INCR-EA22002-14496.pdf

[10] NHTSA. Recall Query RQ24009. https://static.nhtsa.gov/odi/inv/2024/INOA-RQ24009-12046.pdf

[11] Repairer Driven News (2024). NHTSA Investigates Tesla Recall Remedy. https://www.repairerdrivennews.com/2024/05/21/nhtsa-investigates-tesla-recall-remedy/

[12] BRC Legal (2025). Tesla Recall Statistics. https://brclegal.com/tesla-recall-statistics/

[13] Hackett et al. (2025). Bypassing LLM Guardrails. arXiv:2504.11168. https://arxiv.org/abs/2504.11168

[14] HiddenLayer (2025). Policy Puppetry: Universal AI Bypass. https://hiddenlayer.com/innovation-hub/novel-universal-bypass-for-all-major-llms

[15] Cyber Security News (2025). OpenAI Guardrails Bypassed. https://cybersecuritynews.com/openai-guardrails-bypassed/

[16] Xu et al. (2024). Hallucination is Inevitable. arXiv:2401.11817. https://arxiv.org/abs/2401.11817

[17] OpenAI (2025). Why Language Models Hallucinate. https://openai.com/index/why-language-models-hallucinate/

[18] AIMultiple (2025). AI Hallucination Benchmark. https://research.aimultiple.com/ai-hallucination/

[19] Anh-Hoang et al. (2025). Survey and Analysis of Hallucinations in LLMs. Frontiers in AI 8. https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2025.1622292/full

[20] "My AI is Lying to Me" (2025). Scientific Reports. https://www.nature.com/articles/s41598-025-15416-8

[21] Trust at First Reply (2025). Economic Computation and Economic Cybernetics Studies. https://store.ectap.ro/articole/1867.pdf

[22] Pelrine et al. (2024–2025). Epistemic Calibration in LLMs. arXiv:2411.06528. https://arxiv.org/pdf/2411.06528

[23] Understanding Effects of Miscalibrated AI Confidence (2025). arXiv:2402.07632. https://arxiv.org/html/2402.07632v4

[24] Responsible AI Labs (2025). AI Safety Incidents of 2024. Stanford AI Index cited. https://responsibleailabs.ai/knowledge-hub/articles/ai-safety-incidents-2024

[25] Adversa AI (2025). Top AI Security Incidents 2025. https://adversa.ai/blog/adversa-ai-unveils-explosive-2025-ai-security-incidents-report/

[26] ISACA (2025). Avoiding AI Pitfalls in 2026. https://www.isaca.org/resources/news-and-trends/isaca-now-blog/2025/avoiding-ai-pitfalls-in-2026-lessons-learned-from-top-2025-incidents

[27] Ayala & Bechard (2024). Reducing Hallucination via RAG. NAACL Industry Track. https://aclanthology.org/2024.naacl-industry.19/

[28] Mitigating LLM Hallucinations (2025). Preprints.org. NOT PEER-REVIEWED. https://www.preprints.org/manuscript/202505.1955/v1/download

[29] Lakera (2025). LLM Hallucinations in 2025. https://www.lakera.ai/blog/guide-to-hallucinations-in-large-language-models

[30] Multi-Agent Framework for Hallucination Mitigation (2025). MDPI Information. https://www.mdpi.com/2078-2489/16/7/517

[31] EU AI Act. Regulation (EU) 2024/1689. General-purpose AI obligations effective August 2025.

[32] Promptfoo (2025). How AI Regulation Changed in 2025. https://www.promptfoo.dev/blog/ai-regulation-2025/

[33] Baker Botts (2026). U.S. AI Law Update. https://www.bakerbotts.com/thought-leadership/publications/2026/january/us-ai-law-update