International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

Why Enterprise Agent Deployments Fail: A Field Taxonomy

Karan Nisar

Independent Researcher, USA

ABSTRACT: Enterprise agents retrieve privileged data, select workflow steps, invoke tools, and modify organizational state, yet public evidence conflates canceled pilots, incorrect runs, unsafe actions, and uneconomic services. This paper develops an evidence-grounded field taxonomy of enterprise-agent deployment failure from a structured synthesis of academic research, technical benchmarks, enterprise surveys, standards, and public incidents available by June 30, 2025. Failure is defined at four nested levels (run, service, workflow, program) and attributed across eight causal domains spanning mission and value logic, control specification, state integrity, deliberation, tool execution, coordination and supervision, runtime reliability, and organizational absorption. The evidence does not support a universal failure rate, but it shows that short-task model competence is insufficient evidence for reliable, repeated, policy-constrained, tool-using work, and that consequential failures emerge at system boundaries that lack proportional assurance. The paper contributes a causal cascade, incident codebook, testable propositions, and deployment gates linking evaluation to authority, reversibility, observability, economics, and organizational learning.

KEYWORDS: AI agents, agentic AI, enterprise AI, human-AI systems, reliability, AI governance, AgentOps, sociotechnical systems, evaluation, incident analysis

I. INTRODUCTION

The defining change in enterprise generative AI is not better text generation. A model can now select workflow steps, retrieve internal context, invoke software tools, coordinate with other agents or people, and alter records outside the model itself. This shift from generation to delegated action changes the unit of risk: a wrong answer can become a wrong refund, a leaked record, a duplicate transaction, or a silent change to an operational system.

By mid-2025, enterprise interest in agents had accelerated while the public evidence remained uneven. Gartner forecast that more than 40 percent of agentic AI projects would be canceled by the end of 2027 because of escalating cost, unclear business value, or inadequate risk controls [1]. S&P Global reported that respondents' organizations scrapped an average of 46 percent of AI projects between proof of concept and broad adoption [2]. McKinsey found that 21 percent of respondents reporting organizational generative AI use said their organizations had fundamentally redesigned at least some workflows [3], and BCG classified only 4 percent of surveyed firms as consistently generating significant AI value [4].

Benchmarks expose a related capability gap. The best reported model completed 12.24 percent of OSWorld tasks against 72.36 percent for humans; GPT-4o completed 2.1 percent of a matched WorkArena++ subset against 93.9 percent for humans; and leading systems completed fewer than half of the tool-agent-user tasks in tau-bench [5, 6, 7]. These are controlled demonstrations that short-task competence does not automatically survive longer horizons, changing interfaces, policy constraints, or repeated use.

The common explanation is hallucination, but it is too narrow. A deployment can fail with a perfectly behaving model when the economics are not viable, the workflow does not change, or no team owns the service after the pilot; conversely, a model error need not become a deployment failure if independent controls prevent, detect, contain, and reverse it. This paper therefore treats agent deployment as a sociotechnical control problem and asks which control function first failed, why the deviation propagated, and which barriers failed. It contributes a four-level definition of failure, an eight-domain field taxonomy with coding rules, a causal cascade model, and measurable deployment gates. The evidence cutoff is June 30, 2025.

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

II. DEFINITIONS AND SCOPE

A. What counts as an enterprise agent

An enterprise agent is a software system in which one or more generative models dynamically select among actions, tools, or workflow transitions in pursuit of an organizational goal, with state maintained across at least part of the task. The definition covers agents operating through APIs, enterprise applications, browsers, code environments, or delegated subagents, and excludes deterministic workflows that use a model only inside a fixed step. Product labels such as assistant or copilot matter less than whether the generative component can decide what happens next and whether that decision can affect enterprise data, systems, people, or customers [8, 9, 10].

B. What counts as deployment failure

A deployment fails when it does not meet its declared value, service, safety, compliance, or sustainability criteria, or when it appears to meet them only because compensating human work or risk is left unmeasured. A planned decision not to deploy an unsafe or uneconomic system is a governance success. Failure is defined at four nested units:

1) Run failure: a single trajectory does not complete the task or violates an invariant.
2) Service failure: the deployed agent misses a service-level objective across a population of runs.
3) Workflow failure: the surrounding human and technical process does not improve the target outcome despite acceptable component metrics.
4) Program failure: the initiative is abandoned, rolled back, cannot scale, or produces unacceptable risk-adjusted value.

These units should not be combined without denominators: claims about “the failure rate” are not interpretable unless the unit, population, window, and success threshold are specified.

C. Failure cause, outcome, and exposure

The taxonomy separates cause (which control function first left the admissible operating envelope), outcome (what kind of loss occurred), and exposure (what made the deviation more likely to propagate or more severe). A security breach, for example, is an outcome class, not necessarily a root-cause class: prompt injection may exploit an authority boundary, a state-integrity failure, an unsafe tool interface, or several in sequence. Separating these dimensions makes the diagnosis actionable.

III. RESEARCH DESIGN

A. Structured theory-building evidence synthesis

The paper uses a structured theory-building evidence synthesis rather than a systematic review; the available evidence is too heterogeneous for a pooled failure estimate. Five source classes are combined: peer-reviewed studies and dated preprints; enterprise surveys with disclosed field dates, samples, or methods; government, standards, and security guidance; public decisions and incident disclosures; and practitioner guidance with disclosed implementation experience. Sources were eligible when publicly available by June 30, 2025 and relevant to agent performance, reliability, security, adoption, supervision, or organizational control. Primary and official records receive greatest weight: benchmarks support claims about measured tasks rather than enterprise prevalence, surveys support claims about respondents rather than causal mechanisms, and public incidents illustrate mechanisms because disclosure is selective.

B. Historical cutoff control

The date of first public availability, not a later revision, determines eligibility. For preprints, the version available by the cutoff is the relevant record; for regulations, enactment is distinguished from provisions actually applicable by the cutoff; for vulnerabilities, disclosure is distinguished from evidence of exploitation. This prevents later knowledge from being projected backward into the first half of 2025.

C. Comparative taxonomy refinement

The taxonomy was developed in two passes: a theory-informed pass drawing control functions from sociotechnical systems, task-technology fit, safety engineering, human factors, and MLOps [11, 12, 13, 14, 15], and a comparative pass testing those functions against failure signatures reported in agent benchmarks, security evaluations, multi-agent studies, enterprise surveys, and public cases. Candidate domains were merged when no observable boundary test could distinguish them and split when they implied different owners or controls. The paper does not claim domain prevalence or intercoder reliability; the term field taxonomy means the instrument is designed for field incident analysis, and Section X specifies how it can be validated.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

D. Originality and attribution

The eight-domain taxonomy, boundary-exposure measurement family, autonomy-assurance deficit, causal-cascade adaptation, coding rules, and deployment gates are original syntheses, not established empirical scales. Empirical findings, benchmarks, standards, and public cases are attributed at the point of use. The paper makes no claim to proprietary interviews, undisclosed incident data, or original prevalence estimates.

IV. THE EVIDENCE BASE

A. The enterprise production and value gap

The enterprise record is best read as constrained scaling rather than a single failure statistic. Table I summarizes several high-signal observations; the studies differ in population, wording, and unit, so their percentages are not directly comparable.

TABLE I. ENTERPRISE EVIDENCE ON PRODUCTION, VALUE, AND CONTROL

Source	Population and timing	Reported observation	Interpretation and caveat
Deloitte [16]	2,770 director-to-C-suite leaders involved in GenAI piloting or implementation; May-June 2024, 14 countries	68% said their organizations had moved 30% or fewer GenAI experiments fully into production	Implementation-active sample; not representative and not agent-specific
BCG [4]	1,000 CxOs and senior executives across 59 countries; published October 2024	4% consistently generating significant AI value; 22% beginning to realize substantial gains; 74% no tangible value yet	Proprietary classification from respondents' self-assessment
McKinsey [3]	1,491 participants across 101 nations; fielded July 2024, GDP-weighted	21% of GenAI users reported fundamentally redesigned workflows; more than 80% reported no tangible enterprise-level EBIT impact	Redesign associated with impact; cross-sectional, not causal
S&P Global [2]	1,006 mid- and senior-level respondents, North America and Europe; published May 2025	Average of 46% of AI projects scrapped between proof of concept and broad adoption; 42% abandoned the majority before production	Respondent reports; AI broadly, not agents only
Gartner [17]	644 respondents in the US, Germany, and the UK; fielded Q4 2023	Average of 48% of AI projects reached production, averaging eight months from prototype	Self-reported; an observed survey result, not a forecast
Gartner [18]	432 respondents in six countries; fielded Q4 2024; proprietary maturity model	45% of high-maturity organizations kept AI in production at least three years, versus 20% at low maturity	Cross-sectional association at maturity extremes; broad AI
Gartner [1]	Analyst forecast, June 2025; a January 2025 poll of 3,412 webinar attendees measured investment posture, not cancellations	Forecast that more than 40% of agentic AI projects would be canceled by end of 2027	Forward-looking forecast, not an observed failure rate

Four themes recur. First, deployment volume is not enterprise value: only 1 percent of 238 C-level respondents in a separate McKinsey survey described their generative AI rollouts as mature [19], and fewer than one fifth of State of AI participants said their organizations tracked KPIs for generative AI [3]. Second, workflow redesign was the practice most strongly associated with EBIT impact [3]. Third, data readiness, integration, governance, and skills remain constraints after model access is solved [20, 21]; RAND's interviews likewise identified leadership-driven causes such as persistent data problems and technology-first problem selection, and its familiar statement that more than 80 percent of AI projects fail was a cited estimate, not a measured rate [22]. Fourth, operational controls remain incomplete: 44 percent of respondents had experienced at least one negative consequence from generative AI, while only 18 percent reported an enterprise-wide responsible-AI body with decision authority [23], and LangChain's vendor survey paired 51 percent claiming production agents with only about a third mentioning evaluation [24].

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

B. The composition gap in agent benchmarks

Benchmarks available by June 2025 repeatedly showed performance falling as agents had to compose more steps, maintain state, obey policy, recover from errors, or interact with real interfaces. Table II reports original evaluation snapshots; they compare different models, tasks, and dates and are not a leaderboard.

TABLE II. HISTORICAL BENCHMARK SNAPSHOTS SHOWING THE COMPOSITION GAP

Benchmark	Environment	Reported result in the original evaluation	Failure signal
GAIA [25]	Realistic assistant questions requiring reasoning and tools	Humans 92%; GPT-4 with plugins 15%	Tool use and multi-step reasoning far below human performance
OSWorld [6]	369 tasks in real computer environments	Humans 72.36%; best evaluated model 12.24%	GUI grounding, operational knowledge, long action sequences
WorkArena ++ [5]	682 compositional tasks in an enterprise application	Humans 93.9% vs GPT-4o 2.1% on a matched subset; all agents 0% on the most implicit level	Atomic competence did not survive compositional work
tau-bench [7]	Policy-constrained retail and airline interactions	Leading agents completed fewer than half of tasks; retail pass^8 below 25%	One-run averages concealed poor repeated-use consistency
CRMArena-Pro [26]	4,280 enterprise CRM queries, single- and multi-turn	Best single-turn 58.3% (B2C); best multi-turn 35.1% (B2B); near-zero inherent confidentiality awareness	Settings not a paired comparison; interaction and confidentiality exposed control challenges
MAST [27]	More than 200 execution traces across seven multi-agent systems	Failure rates across six systems ranged 41.0%-86.7% on differing benchmarks; 14 failure modes clustered around specification, alignment, and verification	Cross-system rates not directly comparable
TheAgentCompany v2 [28]	175 consequential office-work tasks across enterprise-style tools	Best model fully completed 30.3%, averaging 27.2 steps and $4.20 per task	Cost, long trajectories, tool errors, partial completion

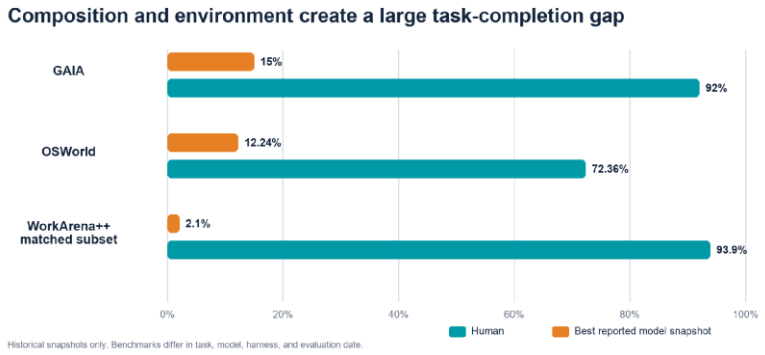

Fig. 1. Historical model-human task-completion gaps in three original benchmark evaluations. Scores are not directly comparable because tasks, models, harnesses, and dates differ.

WorkArena's atomic tasks showed that models could sometimes operate enterprise interfaces, while WorkArena++ showed that composing those skills into realistic knowledge work produced a sharp collapse [5, 29]. AgentBoard found that tasks with more subgoals produced substantial losses, and OSWorld identified grounding and operational knowledge as persistent constraints [30, 6]. TheAgentCompany placed an agent in a simulated software company; the strongest evaluated model fully completed 30.3 percent of 175 tasks with substantial step count and cost [28]. Tau-bench's pass^k metric, which requires all k repeated trials to succeed, shows that a system can look useful when measured once and be

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

unreliable at enterprise volume [7]. CRMArena-Pro reported a best single-turn result of 58.3 percent and a best multi-turn result of 35.1 percent, with near-zero inherent confidentiality awareness [26].

Security evaluations add an adversarial version of the same composition problem. AgentDojo found agents failing many benign tasks while indirect prompt injection redirected behavior, with defenses trading off utility [31]. InjecAgent and earlier work showed that retrieved or tool-returned content can act as an instruction channel unless the system enforces a boundary outside the model [32, 33], and ToolEmu demonstrated risky tool trajectories even without an adversary [34]. Together these establish a narrow proposition: capability measured on short or static tasks is insufficient evidence for the reliability of a long-running, policy-constrained, tool-using service.

C. Public cases as mechanism evidence

Public cases reveal mechanisms, not prevalence. In Moffatt v. Air Canada, a tribunal held the airline responsible for incorrect fare information from its website chatbot; the system predates this paper's agent definition but shows that an enterprise cannot treat an automated representative as legally separate [35]. Slack disclosed and patched a vulnerability under which a malicious workspace member could phish users for certain data, reporting no evidence of unauthorized access [36]. CVE-2025-32711 described an AI command-injection flaw associated with Microsoft 365 Copilot that could enable information disclosure; a disclosed vulnerability is not evidence of exploitation [37]. In each case the consequential question is whether the organization separated trusted policy from untrusted content, limited authority, mediated actions, and retained the ability to contain and recover.

V. THEORETICAL FOUNDATIONS

A. Joint optimization and task-technology fit

Sociotechnical systems theory holds that organizational performance depends on jointly designing technical and social subsystems, and task-technology fit predicts that a technology improves performance only when its capabilities match the work as actually used [12]. A technically successful pilot can therefore still fail: if the selected task is not a bottleneck, exceptions dominate, incentives oppose adoption, or verification work remains in place, better model output does not create value. Anthropic and OpenAI both advised builders to prefer simple workflows and to add agentic autonomy only when task variability justifies it [8, 9].

B. Safety as a control problem

Safety engineering shifts analysis away from single broken components. Leveson locates losses in control structures that fail to enforce constraints through adequate feedback [13]; Reason distinguishes active failures from latent conditions and failed defenses [38]; Rasmussen emphasizes risk emerging across organizational levels with different incentives and feedback delays [14]. A hallucinated plan is an active deviation; excess permissions, missing approval rules, incomplete traces, and an overloaded review queue are latent conditions or weakened barriers. The same model error may be harmless in a read-only sandbox and consequential in a production account with irreversible tools.

C. Human supervision and the ironies of automation

Human oversight is not a universal safeguard. Automation can remove routine practice while leaving people responsible for rare, ambiguous, time-sensitive exceptions [11]; trust should support appropriate reliance rather than maximal acceptance [39]; and levels of automation should be designed by function [40]. A confirmation button does not create independent control if the reviewer cannot reconstruct the plan, inspect sources, or reverse the action. McKinsey's 2025 survey found that only about 27 percent of respondents reviewed all generative AI outputs before use [3], and a study of 319 knowledge workers found that greater confidence in generative AI was associated with less self-reported critical-thinking effort [41]. Nominal human involvement should therefore not be treated as reliable verification.

D. From MLOps and SRE to AgentOps

MLOps research established that hidden technical debt, data dependencies, configuration changes, and undeclared consumers can dominate the lifecycle of machine-learning systems [42, 15], and data cascades propagate upstream quality and ownership problems into downstream harms [43]. Agents add dynamic prompts, plans, memory, tool schemas, policies, and multi-step execution to that debt surface. Site reliability practices such as service-level objectives, canaries, and postmortems are necessary but insufficient: a trajectory can return an HTTP 200 response while violating policy or changing the wrong record, so agent observability must connect prompts, retrieved context, versions, plans, tool calls, state changes, approvals, costs, and evaluation results into a replayable trace [44].

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

VI. THE EIGHT-DOMAIN ENTERPRISE AGENT FAILURE TAXONOMY

The taxonomy contains eight mutually distinguishable domains (Table III). For run or service incidents, D1 through D6 normally identify execution-time control failures, while D0 and D7 usually describe latent conditions or failed barriers. An incident receives one primary code only when the evidence supports attribution to the earliest evidenced failed invariant within the selected unit of analysis; otherwise it receives DX, indeterminate, with compound flags, attribution confidence, and competing explanations recorded separately.

TABLE III. ENTERPRISE AGENT FAILURE TAXONOMY

Code	Primary domain	Boundary test	Typical signatures	Control owner most likely to act
D0	Mission, task fit, and value logic	Would the deployment still fail to create acceptable net value if the agent behaved exactly as specified?	No KPI movement, low adoption, automation of a non-bottleneck, review cost erases savings	Business owner, product, finance, operations
D1	Control specification and authority boundaries	Were goals, policies, permissions, tools, or approval conditions ambiguous, excessive, conflicting, or unenforced?	Policy breach, approval bypass, prompt injection becomes action, unauthorized scope	Product, security, risk, platform
D2	Knowledge, context, and state integrity	Was the agent's representation of relevant state false, stale, incomplete, conflated, or poisoned?	Wrong entity, stale record, retrieval miss, unsupported claim, contaminated memory	Data, knowledge, product, platform
D3	Deliberation, planning, and termination	Given a valid goal, authority, and available state, did the agent choose an invalid plan or next action?	Skipped precondition, loop, goal drift, wrong sequence, premature stop	Agent engineering, evaluation, product
D4	Tool, interface, and transaction execution	Was the recorded pre-execution intent valid but serialized, bound, adapted, retried, or executed into the wrong external-system effect?	Schema or target misbinding, duplicate action, partial update, GUI drift, silent tool error	Application engineering, platform, SRE
D5	Multi-actor coordination and human supervision	Did failure occur at a handoff, shared-state update, conflict-resolution point, or human escalation?	Lost constraint, duplicated work, stale shared state, rubber-stamp approval, late escalation	Workflow owner, UX, operations, agent engineering
D6	Runtime, dependency, and change reliability	Did the operational substrate fail its availability, latency, capacity, consistency, cost, or change envelope?	Timeout, rate-limit cascade, cost spike, provider regression, retry storm, trace gap	SRE, platform, vendor management
D7	Organizational absorption, assurance, and learning	Did the enterprise fail to own, integrate, evaluate, govern, improve, or retire the system over time?	Shadow use, workarounds, recurring incidents, stale inventory, unclosed actions, ownerless pilot	Executive sponsor, governance, risk, operations
DX	Indeterminate primary domain	Is the available evidence insufficient to identify one earliest failed invariant?	Sparse disclosure, missing trace, tied failure interval, competing explanations	Incident lead assigns evidence owner before causal action

A. D0: Mission, task fit, and value logic

D0 applies when the system could perform exactly as designed and still fail to produce acceptable risk-adjusted value: automating a task that is not the process bottleneck, using model-centric proxy metrics instead of business outcomes, underestimating exception handling, or omitting integration and review costs from the business case. Leading indicators are economic and behavioral: adoption, correction minutes per completed task, verification burden against gross time

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

saved, and cost per successful outcome. The main control is whole-workflow redesign with a counterfactual baseline and an explicit kill criterion [4, 3].

B. D1: Control specification and authority boundaries

D1 applies when the agent's allowable goals, policies, data access, tools, permissions, or approval conditions are ambiguous, conflicting, excessive, or enforced only through prompts. Signatures include policy violations, approval bypass, excessive tool scope, and prompt injection that becomes an authorized action. A generative model should not be the sole interpreter and enforcer of the constraints that govern its own authority. Controls include typed policy outside the model, deny-by-default tools, least privilege, risk-tiered action gates, and separation of trusted instructions from untrusted content; OWASP's 2025 list distinguishes excessive functionality, permissions, and autonomy as useful subtypes [45].

C. D2: Knowledge, context, and state integrity

D2 applies when the agent's representation of the relevant world is false, stale, incomplete, conflated, or poisoned: retrieval omissions, incorrect entity resolution, hidden conflicts between authoritative systems, context truncation, or persistent-memory contamination. Long-context access does not guarantee effective use: models often perform worst when relevant information sits in the middle of context [46], and a single end-answer score can conceal distinct retrieval and generation failures [47]. Controls include source-of-truth APIs, provenance and freshness metadata, entity confirmation, bounded memory, and explicit abstention policies.

D. D3: Deliberation, planning, and termination

D3 applies when the goal, authority, and available state are adequate but the agent chooses an invalid plan or next action: skipped preconditions, wrong sequencing, goal drift, repeated no-progress loops, premature termination, or failure to terminate. MAST identified specification, coordination, verification, and termination failures across multi-agent systems [27]; AgentBench attributed substantial difficulty to long-term reasoning, decision making, and instruction following [48]; and intrinsic self-correction degraded reasoning performance in tested settings [49]. Reflection and extra agents can add cost and failure surfaces without progress. Controls shorten the unverified action horizon: deterministic workflows for known subproblems, checkpoints and stop conditions, and independent verification of consequential steps.

E. D4: Tool, interface, and transaction execution

D4 applies when the recorded pre-execution intent is valid but serialization, schema binding, an adapter, retry behavior, or downstream execution produces an incorrect external effect: target or parameter misbinding, duplicate messages or refunds after a retry, partial updates, GUI drift, or orphaned state after a timeout. Function-call accuracy is not enough; the system must validate preconditions, postconditions, and resulting state. Controls include typed schemas, contract tests, idempotency keys, read-after-write verification, two-phase confirmation for consequential actions, compensating transactions, and structured errors that cannot be mistaken for success.

F. D5: Multi-actor coordination and human supervision

D5 applies when individually plausible components fail at a handoff, role boundary, shared-state update, conflict-resolution point, or escalation to a person; D5a covers machine-to-machine coordination and D5b covers human supervisory control. Signatures include duplicated or contradictory work, lost constraints, stale shared state, late escalation, review-queue saturation, and rubber-stamping. MultiAgentBench found in a research ablation that increasing team size reduced its overall KPI, and MAST documented role-specification and inter-agent misalignment failures across seven systems [27, 50]. Controls include single-agent-first design, explicit role and handoff contracts, canonical shared state, evidence-rich approval interfaces, and reviewer-capacity planning.

G. D6: Runtime, dependency, and change reliability

D6 applies when the operational substrate fails to deliver the agent loop within its availability, latency, capacity, consistency, cost, or change envelope: timeouts, rate-limit cascades, retry storms, cost spikes, provider or model regression, configuration drift, or fallback failure. Agent services need two classes of objective: conventional objectives covering traffic, errors, saturation, and latency, and semantic objectives covering task completion, policy adherence, correct state change, and cost per successful task. A service may be technically available while semantically failing. Controls include end-to-end traces and replay, semantic error budgets, canary releases, version pinning, circuit breakers, step and cost limits, and change-triggered reevaluation; privacy, safety, authorization, and legal constraints are hard invariants, not spendable error budgets.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

H. D7: Organizational absorption, assurance, and learning

D7 applies when the enterprise fails to integrate, own, evaluate, govern, improve, or retire the system over time: shadow deployment, workarounds, stale agent inventories, audit gaps, unclosed postmortem actions, value decay, and ownerless services after the pilot team moves on. NIST's Generative AI Profile and ISO/IEC 42001 frame governance, measurement, incident response, and deactivation as lifecycle practices rather than launch-time checks [51, 52]. D7 often appears as a failed barrier after another domain triggers an incident, but it is primary when the earliest evidenced failed invariant is organizational, such as deploying without ownership or operating indefinitely without a valid control process.

VII. FROM LOCAL DEVIATION TO ENTERPRISE LOSS

A. Causal cascade

The taxonomy is embedded in a causal sequence: external pressure and latent conditions, a trigger, the earliest evidenced failed invariant within the selected unit, an unsafe or ineffective control action, propagation, failure of prevention, detection, containment, or recovery barriers, and finally enterprise loss followed by learning or recurrence (Fig. 2). This sequence avoids two attribution errors: model monocausality, where every incident is labeled a hallucination, and governance vagueness, where every failure is assigned to a lack of governance without identifying the broken control. Unsafe control actions take five observable forms:

1) A required action was omitted.
2) An unsafe action was issued.
3) A valid action occurred at the wrong time or in the wrong order.
4) An action continued too long or stopped too soon.
5) An action targeted the wrong entity, scope, or magnitude.

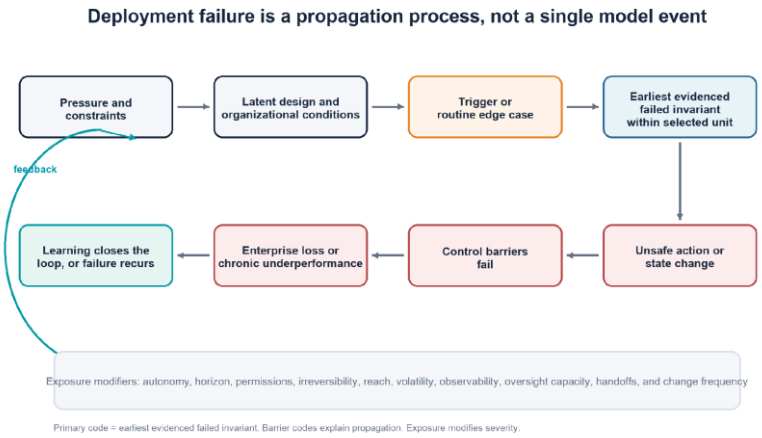

Fig. 2. Causally structured cascade for enterprise agent deployment failure. When evidence is sufficient, the primary taxonomy code marks the earliest evidenced failed invariant within the selected unit; barrier and exposure codes explain propagation and severity.

B. Reliability compounds across horizons

If a task requires n steps, the general chain rule gives:

$$P(\text{sequence success}) = \prod P(S_i \mid S_1, \ldots, S_{i-1})$$

Under an illustrative assumption of homogeneous independent steps, 95 percent per-step reliability across 20 steps yields sequence success of about 35.8 percent, while 99 percent yields about 81.8 percent. Real steps are not independent; correlated errors, recovery, and checkpoints move performance in either direction. The point is mechanism, not estimation: small atomic error rates can become material over long horizons, so services should report end-to-end success, invariant compliance, and repeated-run reliability rather than inferring reliability from component accuracy [7].

C. Boundary exposure

Boundary exposure is proposed as a measurement family, not a validated scalar: transition count, trust-level change, privilege escalation, externalization of effect, and loss of reversibility across policy, data, tool, identity, human, and organizational boundaries. It describes where a local deviation may encounter an ambiguous contract or weak barrier; a

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

well-specified, independently enforced boundary can be safer than a nominally internal but ambiguous transition. Other exposure modifiers include autonomy level, action horizon, permission breadth, irreversibility, external reach, oversight latency and capacity, change frequency, and adversarial input.

D. The autonomy-assurance deficit

Autonomy is often increased because the model can complete more steps, but assurance does not scale automatically with capability. The autonomy-assurance deficit is a proposed construct comparing delegated exposure with control coverage across authority, verification, observability, reversibility, containment, and recovery. The deficit grows when permissions broaden faster than policy enforcement, action horizons lengthen without checkpoints, approvals are added without reviewer capacity, or production changes outpace evaluation refresh.

E. Hidden labor, trust, and value decay

Deployments can appear successful while shifting work rather than removing it: people rewrite prompts, curate context, correct outputs, and reconcile systems. Unmeasured, this hidden labor lets gross automation metrics overstate net value, and cost pressure then erodes verification further. Trust follows a related loop: early errors increase checking and reduce adoption, while high apparent reliability reduces vigilance and weakens the human fallback for rare events; appropriate reliance requires calibrated evidence [39].

VIII. APPLYING THE TAXONOMY

A. Coding rule

For each incident or chronic underperformance case, select the unit of analysis, identify the earliest evidenced failed invariant, assign one primary domain only when the record supports it, and code up to three secondary contributors together with failed barriers, exposure modifiers, loss classes, attribution confidence, and the smallest credible counterfactual control. In a hypothetical case where stale CRM data causes an agent to select the wrong customer and a timeout then produces a duplicate message, D2 is primary because incorrect state caused the first divergence, D4 is secondary because a non-idempotent retry duplicated the effect, the timeout is a D6 contributor, and missing entity confirmation and idempotency are failed preventive barriers.

B. Worked public cases

Applied to the public cases in Section IV, sparse disclosures support only candidate domains: the Air Canada case reads as a D2 and D7 analogue, while the Slack and Copilot disclosures each admit D1, D2, or D7 readings and would be coded DX without fuller traces. A leaked record is not automatically a security root cause, and a duplicate payment is not automatically a tool root cause; coding the earliest evidenced failed invariant makes the remedy specific and reduces the tendency to blame the last component that touched the loss.

IX. MEASUREMENT AND DEPLOYMENT GATES

A. A minimum evidence package

Before production, an enterprise agent should have a minimum evidence package covering all eight domains:

1) Value: baseline process performance, target outcome, adoption assumption, risk-adjusted unit economics, and kill rule.
2) Authority: tool and data inventory, permission map, policy tests, action-risk tiers, and approval conditions.
3) State: source-of-truth map, provenance and freshness guarantees, retrieval tests, entity checks, and memory lifecycle.
4) Reliability: end-to-end success, invariant compliance, repeated-run stability, edge-case strata, and deterministic baseline comparison.
5) Transactions: contract tests, idempotency, reversibility, compensating actions, fault injection, and postcondition checks.
6) Supervision: role and handoff contracts, reviewer capacity, escalation precision and recall, and fallback drills.
7) Operations: service and semantic objectives, error budgets, cost and step limits, trace completeness, canary plan, and rollback criteria.
8) Learning: named owners, change-control triggers, production-evaluation plan, incident process, and retirement rule.

This package changes the deployment question from whether the model is accurate enough to whether the complete work system is controlled enough for the proposed exposure.

International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

B. An autonomy ladder

Deployment should progress through an autonomy ladder tied to evidence (Table IV). Movement up the ladder is earned by production evidence of stable end-to-end performance under realistic load, exceptions, attacks, dependency changes, and recovery drills. A higher-capability model is not, by itself, evidence for broader authority.

TABLE IV. AUTONOMY LADDER AND REQUIRED ASSURANCE BY DELEGATION LEVEL

Level	Agent role	Typical action rights	Required assurance
A0	Observe and evaluate	No user-facing output or state change	Representative traces, baseline, privacy controls
A1	Read-only assistance	Retrieve, summarize, recommend	Provenance, uncertainty display, quality monitoring
A2	Draft or propose	Prepare actions for human execution	Evidence-rich review, reviewer capacity, audit trail
A3	Act with approval	Execute bounded actions after confirmation	Independent policy checks, typed effects, idempotency, rollback
A4	Bounded autonomous action	Execute predefined reversible actions within limits	Semantic SLOs, least privilege, canaries, circuit breakers, continuous evaluation
A5	Broad or irreversible autonomy	External or high-impact action without per-action review	Presumptively exceptional; requires a demonstrated necessity and substantially stronger independent controls

C. Metrics that resist gaming

No single metric can represent agent dependability. A balanced set spans outcomes (process KPI, adoption), reliability (end-to-end success, pass^k, invariant violations), control (unauthorized actions, permission utilization), state quality (retrieval recall, provenance), operations (latency, trace completeness, error-budget burn), economics (cost per successful task, correction minutes, reviewer load), the human system (override and acceptance rates, escalation precision and recall), and learning (recurrence, corrective-action closure). Metrics must be stratified by task type, risk tier, and version; aggregates can hide rare severe failures and reward an agent for declining difficult work.

X. RESEARCH PROPOSITIONS AND VALIDATION AGENDA

The taxonomy generates testable propositions rather than a universal failure-rate claim. P1: holding task type and model constant, longer uncheckpointed trajectories will show lower repeated-run reliability, with severity rising only when horizon interacts with broader permissions, irreversibility, or failed containment. P2: production failure will be predicted more strongly by exception diversity, state volatility, and dependency change than by static benchmark accuracy. P3: autonomy, permission breadth, and irreversibility will interact in predicting loss severity even when the model-error rate is unchanged. P4: controls enforced outside the generative model will reduce policy and security incidents more than semantically equivalent prompt-only controls. P5: adding agents will reduce performance when coordination overhead and contract defects exceed the benefit of decomposition. P6: human approval will reduce consequential loss only when reviewers have adequate expertise, evidence, time, and authority, with false acceptance rising as review utilization approaches capacity. P7: task-technology fit and workflow redesign will predict sustained adoption and net value after controlling for technical accuracy. P8: timely closure of measurable postmortem actions and conversion of production corrections into regression tests will reduce recurrence.

Field validation should use an abductive, mixed-method, multiple-case design sampling successes, failures, and near misses across industries, risk levels, and autonomy levels, with at least two independent coders reporting Cohen's kappa or Krippendorff's alpha, sequence analysis of recurring cascades, and mixed-effects models treating organization and use case as random effects. The corpus should contain production traces, tool calls, state changes, incident records, and interviews rather than hidden chain-of-thought.

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

XI. IMPLICATIONS

For enterprise leaders, agent portfolios are operating-model investments rather than model demonstrations: funding should require an outcome baseline, a process owner, a risk-adjusted cost model, and stop criteria, and the decisive question is whether the organization is redesigning the work and its controls.

For product and engineering teams, the architecture should minimize boundary exposure and unverified horizon length: deterministic flows for stable subproblems, single-agent designs before multi-agent decomposition, explicit state over conversational memory, typed tools over open-ended interfaces, and reversible actions before broad autonomy, with evaluation replaying realistic trajectories across model, prompt, policy, tool, and data versions.

For risk, security, and compliance teams, consequential policies should be enforceable outside the model through least privilege, independent mediation, content provenance, action gating, and complete auditability. The EU AI Act entered into force on August 1, 2024, and its first provisions became applicable on February 2, 2025; later obligations should not be projected into the first half of 2025 [53, 54].

For operations and workforce design, supervision must be capacity planned: correction effort, review queues, escalation quality, and workarounds should be measured, and workers who understand exceptions should participate in design and postmortems. The aim is a functioning control loop with adequate evidence, time, and authority, not a person nominally in the loop.

For vendors and procurement, contracts should cover task-stratified success, repeated-run reliability, full cost per successful task, permission architecture, incident notification, reevaluation rights, and version transparency, with vendor benchmark claims evaluated against the buyer's own workflow and exception distribution.

XII. LIMITATIONS

The evidence base is limited. Enterprise failures are underreported for legal, reputational, and security reasons; public cases overrepresent visible customer and security events; most surveys cover AI broadly rather than agents, rely on self-reported outcomes, and differ in sample and wording; and analyst forecasts are not observed outcomes. Benchmarks are controlled approximations that omit organizational integration, real permissions, and changing dependencies, and their scores are model- and version-specific; reviews have identified inadequate holdouts, cost blindness, and reproducibility problems [55], and LLM-based judges can introduce position, verbosity, or self-preference bias [56, 57].

The earliest-evidenced-invariant rule improves actionability but may simplify cases in which control functions fail nearly simultaneously; DX codes, compound flags, confidence scores, and competing explanations preserve uncertainty. Boundary exposure and the autonomy-assurance deficit are proposed constructs requiring measurement validation, and success thresholds are negotiated organizational choices that can be gamed. Finally, this paper is a secondary synthesis without proprietary production traces, new interviews, or a representative incident denominator; its contribution is a codeable theory and field instrument, not an estimate of how often each domain occurs.

XIII. CONCLUSION

Enterprise agents do not fail for one reason. They fail when a probabilistic controller is inserted into a work system without jointly redesigning mission, authority, state, interfaces, supervision, operations, and learning. A model error may trigger an incident, but a local deviation becomes enterprise loss when surrounding controls allow it to propagate. The evidence available through June 30, 2025 supports three conclusions: enterprise adoption and enterprise value are different milestones; atomic model competence does not reliably survive long, repeated, policy-constrained, tool-using work; and autonomy changes the consequence of error because outputs become actions that mutate organizational state.

The eight-domain taxonomy separates the first failed control function from the loss, the trigger, the failed barriers, and the exposure modifiers. The practical implication is simple. Capability alone does not make an agent dependable. Dependability comes from constrained authority, trustworthy state, short and observable horizons, safe transactions, functioning human control, resilient operations, measurable economics, and organizational learning.

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

REFERENCES

[1] Gartner, "Gartner predicts over 40% of agentic AI projects will be canceled by end of 2027," Gartner Newsroom, Jun. 25, 2025. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2025-06-25-gartner-predicts-over-40-percent-of-agentic-ai-projects-will-be-canceled-by-end-of-2027
[2] S&P Global Market Intelligence, "Generative AI experiences rapid adoption, but with mixed outcomes: Highlights from VotE: AI & Machine Learning," May 30, 2025. [Online]. Available: https://www.spglobal.com/market-intelligence/en/news-insights/research/ai-experiences-rapid-adoption-but-with-mixed-outcomes-highlights-from-vote-ai-machine-learning
[3] McKinsey & Company, "The state of AI: How organizations are rewiring to capture value," Mar. 12, 2025. [Online]. Available: https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-how-organizations-are-rewiring-to-capture-value
[4] Boston Consulting Group, "Where's the value in AI?" Oct. 24, 2024. [Online]. Available: https://www.bcg.com/publications/2024/wheres-value-in-ai
[5] L. Boisvert et al., "WorkArena++: Towards compositional planning and reasoning-based common knowledge work tasks," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 37, 2024.
[6] T. Xie et al., "OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 37, 2024. [Online]. Available: https://arxiv.org/abs/2404.07972
[7] S. Yao, N. Shinn, P. Razavi, and K. Narasimhan, "Tau-bench: A benchmark for tool-agent-user interaction in real-world domains," arXiv preprint arXiv:2406.12045, 2024.
[8] Anthropic, "Building effective agents," Dec. 19, 2024. [Online]. Available: https://www.anthropic.com/engineering/building-effective-agents
[9] OpenAI, "A practical guide to building agents," Apr. 2025. [Online]. Available: https://cdn.openai.com/business-guides-and-resources/a-practical-guide-to-building-agents.pdf
[10] L. Wang et al., "A survey on large language model based autonomous agents," Front. Comput. Sci., vol. 18, art. no. 186345, 2024.
[11] L. Bainbridge, "Ironies of automation," Automatica, vol. 19, no. 6, pp. 775-779, 1983.
[12] D. L. Goodhue and R. L. Thompson, "Task-technology fit and individual performance," MIS Quart., vol. 19, no. 2, pp. 213-236, 1995.
[13] N. G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety. Cambridge, MA, USA: MIT Press, 2011.
[14] J. Rasmussen, "Risk management in a dynamic society: A modelling problem," Safety Sci., vol. 27, no. 2-3, pp. 183-213, 1997.
[15] D. Sculley et al., "Hidden technical debt in machine learning systems," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 28, 2015.
[16] Deloitte, "The state of generative AI in the enterprise, Q3 report," Aug. 27, 2024. [Online]. Available: https://www.deloitte.com/uk/en/about/press-room/deloitte-ai-institute-state-of-generative-ai-in-the-enterprise-report.html
[17] Gartner, "Gartner survey finds generative AI is now the most frequently deployed AI solution in organizations," Gartner Newsroom, May 7, 2024. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2024-05-07-gartner-survey-finds-generative-ai-is-now-the-most-frequently-deployed-ai-solution-in-organizations
[18] Gartner, "Gartner survey finds 45% of organizations with high AI maturity keep AI projects operational for at least three years," Gartner Newsroom, Jun. 30, 2025. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2025-06-30-gartner-survey-finds-forty-five-percent-of-organizations-with-high-artificial-intelligence-maturity-keep-artificial-intelligence-projects-operational-for-at-least-three-years
[19] H. Mayer, L. Yee, M. Chui, and R. Roberts, "Superagency in the workplace: Empowering people to unlock AI's full potential," McKinsey & Company, Jan. 28, 2025. [Online]. Available: https://www.mckinsey.com/capabilities/tech-and-ai/our-insights/superagency-in-the-workplace-empowering-people-to-unlock-ais-full-potential-at-work/
[20] IBM, "Data suggests growth in enterprise adoption of AI is due to widespread deployment by early adopters," IBM Newsroom, Jan. 10, 2024. [Online]. Available: https://newsroom.ibm.com/2024-01-10-Data-Suggests-Growth-in-Enterprise-Adoption-of-AI-is-Due-to-Widespread-Deployment-by-Early-Adopters
[21] Gartner, "Lack of AI-ready data puts AI projects at risk," Gartner Newsroom, Feb. 26, 2025. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2025-02-26-lack-of-ai-ready-data-puts-ai-projects-at-risk
[22] J. Ryseff, B. De Bruhl, and S. Newberry, "The root causes of failure for artificial intelligence projects and how they can succeed," RAND Corporation, Santa Monica, CA, USA, Rep. RRA2680-1, 2024. [Online]. Available: https://www.rand.org/pubs/research_reports/RRA2680-1.html

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

[23] McKinsey & Company, "The state of AI in early 2024: Gen AI adoption spikes and starts to generate value," May 30, 2024. [Online]. Available: https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-2024
[24] LangChain, "State of AI agents," Nov. 14, 2024. [Online]. Available: https://www.langchain.com/resources/state-of-ai-agents-report
[25] G. Mialon, C. Fourrier, C. Swift, T. Wolf, Y. LeCun, and T. Scialom, "GAIA: A benchmark for general AI assistants," in Proc. Int. Conf. Learn. Represent. (ICLR), 2024.
[26] K.-H. Huang et al., "CRMArena-Pro: Holistic assessment of LLM agents across diverse business scenarios and interactions," arXiv preprint arXiv:2505.18878v1, 2025.
[27] M. Cemri et al., "Why do multi-agent LLM systems fail?" arXiv preprint arXiv:2503.13657v2, 2025.
[28] F. F. Xu et al., "TheAgentCompany: Benchmarking LLM agents on consequential real world tasks," arXiv preprint arXiv:2412.14161v2, 2025.
[29] A. Drouin et al., "WorkArena: How capable are web agents at solving common knowledge work tasks?" in Proc. 41st Int. Conf. Mach. Learn. (ICML), vol. 235, 2024, pp. 11642-11662.
[30] C. Ma et al., "AgentBoard: An analytical evaluation board of multi-turn LLM agents," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 37, 2024.
[31] E. Debenedetti, J. Zhang, M. Balunovic, L. Beurer-Kellner, M. Fischer, and F. Tramer, "AgentDojo: A dynamic environment to evaluate prompt injection attacks and defenses for LLM agents," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 37, 2024.
[32] K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, "Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection," in Proc. 16th ACM Workshop Artif. Intell. Secur. (AISec), 2023, pp. 79-90.
[33] Q. Zhan, Z. Liang, Z. Ying, and D. Kang, "InjecAgent: Benchmarking indirect prompt injections in tool-integrated large language model agents," in Findings Assoc. Comput. Linguistics: ACL 2024, 2024, pp. 10471-10506.
[34] Y. Ruan et al., "Identifying the risks of LM agents with an LM-emulated sandbox," in Proc. Int. Conf. Learn. Represent. (ICLR), 2024.
[35] Moffatt v. Air Canada, 2024 BCCRT 149, Civil Resolution Tribunal of British Columbia, 2024. [Online]. Available: https://www.canlii.org/en/bc/bccrt/doc/2024/2024bccrt149/2024bccrt149.html
[36] Slack, "Slack security update," Aug. 21, 2024. [Online]. Available: https://slack.com/blog/news/slack-security-update-082124
[37] National Vulnerability Database, "CVE-2025-32711," National Institute of Standards and Technology, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2025-32711
[38] J. Reason, "Human error: Models and management," BMJ, vol. 320, no. 7237, pp. 768-770, 2000.
[39] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," Human Factors, vol. 46, no. 1, pp. 50-80, 2004.
[40] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," IEEE Trans. Syst., Man, Cybern. A, Syst. Humans, vol. 30, no. 3, pp. 286-297, 2000.
[41] H.-P. Lee et al., "The impact of generative AI on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers," in Proc. 2025 CHI Conf. Human Factors Comput. Syst. (CHI), 2025, art. no. 1121, pp. 1-22.
[42] S. Amershi et al., "Software engineering for machine learning: A case study," in Proc. 41st Int. Conf. Softw. Eng.: Softw. Eng. Pract. (ICSE-SEIP), 2019, pp. 291-300.
[43] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. M. Aroyo, "'Everyone wants to do the model work, not the data work': Data cascades in high-stakes AI," in Proc. 2021 CHI Conf. Human Factors Comput. Syst. (CHI), 2021, art. no. 39.
[44] Y. Dong, Q. Lu, and L. Zhu, "AgentOps: Enabling observability of LLM agents," arXiv preprint arXiv:2411.05285, 2024.
[45] OWASP Foundation, "OWASP Top 10 for LLM applications 2025," Nov. 18, 2024. [Online]. Available: https://owasp.org/www-project-top-10-for-large-language-model-applications/assets/PDF/OWASP-Top-10-for-LLMs-v2025.pdf
[46] N. F. Liu et al., "Lost in the middle: How language models use long contexts," Trans. Assoc. Comput. Linguistics, vol. 12, pp. 157-173, 2024.
[47] D. Ru et al., "RAGChecker: A fine-grained framework for diagnosing retrieval-augmented generation," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 37, 2024.
[48] X. Liu et al., "AgentBench: Evaluating LLMs as agents," in Proc. Int. Conf. Learn. Represent. (ICLR), 2024.
[49] J. Huang et al., "Large language models cannot self-correct reasoning yet," in Proc. Int. Conf. Learn. Represent. (ICLR), 2024.

| ISSN: 2320-0081 | www.ijctece.com | A Peer-Reviewed, Refereed, a Bimonthly Journal|

|| Volume 8, Issue 5, September – October 2025 ||

DOI: 10.15680/IJCTECE.2025.0805037

[50] K. Zhu et al., "MultiAgentBench: Evaluating the collaboration and competition of LLM agents," arXiv preprint arXiv:2503.01935v1, 2025.
[51] C. Autio et al., "Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile," National Institute of Standards and Technology, Gaithersburg, MD, USA, NIST AI 600-1, 2024.
[52] Information Technology, Artificial Intelligence, Management System, ISO/IEC 42001:2023, International Organization for Standardization, Geneva, Switzerland, 2023.
[53] European Commission, "AI Act enters into force," Aug. 1, 2024. [Online]. Available: https://commission.europa.eu/news-and-media/news/ai-act-enters-force-2024-08-01_en
[54] European Commission, "First rules of the Artificial Intelligence Act are now applicable," Feb. 2, 2025. [Online]. Available: https://ai-watch.ec.europa.eu/news/first-rules-artificial-intelligence-act-are-now-applicable-2025-02-02_en
[55] S. Kapoor, B. Stroebl, Z. S. Siegel, N. Nadgir, and A. Narayanan, "AI agents that matter," arXiv preprint arXiv:2407.01502v1, 2024.
[56] L. Shi, C. Ma, W. Liang, X. Diao, W. Ma, and S. Vosoughi, "Judging the judges: A systematic study of position bias in LLM-as-a-judge," arXiv preprint arXiv:2406.07791v8, 2024.
[57] L. Zheng et al., "Judging LLM-as-a-judge with MT-Bench and Chatbot Arena," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 36, 2023.