

ORIGINAL RESEARCH

Open access

Can We Make a Trustworthiness Case for AI? A Methodological Framework for Claims, Evidence, Structured Arguments, Residual Risks, and Deployment Decisions

Thomas Andersen^{1*}, Sofia Lindqvist², David Okoro³

Abstract

Trustworthy artificial intelligence is frequently discussed through principles, performance measures, governance commitments, or general descriptors such as safe, fair, explainable, and responsible. These expressions are important but insufficient for determining whether a particular intelligent system is adequately justified for a defined use, environment, population, and decision context. This article proposes a conceptual AI assurance-case methodology for organizing trustworthiness reasoning through explicit assurance claims, decomposed subclaims, bounded acceptance criteria, evidence requirements, evidence-quality assessment, provenance, traceability, structured argumentation, defeater analysis, uncertainty, and residual-risk reasoning. The methodology responds to distinctive assurance challenges created by model opacity, distributional change, context dependency, conflicting trustworthiness objectives, incomplete evidence, human-oversight limitations, and lifecycle evolution. It distinguishes the existence of evidence from its relevance, validity, credibility, completeness, and sufficiency, while treating traceability as an aid to inspection rather than a guarantee of evidential quality. Structured arguments connect claims to evidence through warrants, assumptions, contextual conditions, and inference links, while counterevidence, argument gaps, evidence gaps, and unknown risks remain visible. Assurance cases may consequently support accountable deployment deliberation by clarifying decision gates, deployment boundaries, monitoring obligations, and conditions requiring reassessment. The principal contribution is an integrated methodology for constructing and maintaining bounded trustworthiness cases without treating assurance reasoning as proof of safety, fairness, legality, ethical sufficiency, compliance, certification, operational readiness, or deployment acceptability. The methodology supports transparent and challengeable reasoning, but requires domain-specific evaluation, governance review, stakeholder participation, and continuing lifecycle monitoring.

Keywords Trustworthy AI, Evidence traceability, AI assurance, Assurance case, Structured argumentation, Residual risk

*Correspondence:

Thomas Andersen

thomas.andersen@di.ku.dk

¹ Department of Explainable AI and Human-Computer Interaction, Faculty of Engineering, University of Copenhagen, Copenhagen, Denmark

² Department of AI and Society, Stockholm University, Stockholm, Sweden

³ Department of Computer Science, University of Lagos, Lagos, Nigeria

Introduction

Trustworthy AI scholarship has produced broad agreement around values such as transparency, fairness, accountability, safety, human agency, and social benefit, but these principles remain heterogeneous in interpretation and implementation. Trustworthiness is therefore better understood as a multidimensional and context-sensitive object of justification than as a general attribute that can be assigned to an intelligent system without qualification [1-3]. An assurance case responds to this problem by asking what is claimed, for which use, under which assumptions, with what evidence, and within which decision boundary.

Human-centered AI further shows that reliable intelligent systems cannot be assessed solely through model performance. Human authority, competence, intervention capacity, organizational responsibility, and the consequences of use remain part of the system being assured [4]. A technically accurate model may still be unsuitable when users cannot contest its output, operators cannot recognize failure, affected stakeholders are excluded, or responsibility is dispersed across development and deployment actors. Assurance reasoning must therefore address socio-technical arrangements rather than isolate the algorithm from its operational environment.

An AI assurance case is not synonymous with a safety case, audit report, compliance assessment, model card, risk register, or governance framework. It is a structured and reviewable argument that connects bounded claims to relevant evidence through explicit warrants, assumptions, contextual statements, and inference links. Its purpose is not to declare that an AI system is trustworthy, but to clarify why specified claims may be considered sufficiently supported for a particular deliberative purpose and where the reasoning remains incomplete, uncertain, contested, or conditional.

This article develops a conceptual methodology for constructing such cases. It integrates assurance-context definition, system and use-case boundaries, stakeholder and responsibility mapping, trustworthiness claims, acceptance criteria, evidence-quality appraisal, provenance, traceability, structured argumentation, defeaters, residual risks, decision boundaries, monitoring, and assurance updating. The contribution is methodological rather than prescriptive: it offers a disciplined architecture for assurance reasoning while maintaining explicit boundaries against trustworthiness proof, automated deployment approval, legal sufficiency, certification, and universal applicability.

Assurance Challenges in Intelligent Systems

Intelligent systems create lifecycle assurance problems because the relationship between past evidence and future behavior may change. Concept drift, data drift, evolving environments, altered user populations, and modified data-generation processes can reduce the relevance of earlier evaluations [5]. Model drift may also arise from retraining, updated dependencies, feedback loops, or operational adaptation. Consequently, assurance cannot be treated as a fixed predeployment judgment; it requires conditions for detecting material change and reopening affected claims, evidence, assumptions, and arguments.

Context dependency creates a second challenge. Strong internal performance does not necessarily establish external validity, workflow compatibility, appropriate human oversight, or reliable behavior in the intended environment [6]. A system boundary must therefore include relevant interfaces, data pipelines, operators, organizational controls, and downstream decisions. Distribution shift, uncertain generalization, incomplete monitoring, and unanticipated interaction effects create residual risks that cannot be resolved by model validation alone. The assurance context must specify where evidence applies and where its inferential reach ends.

Explainability and fairness create distinct assurance challenges because both depend on purpose, stakeholder context, definitions, and evaluation choices [7, 8]. An explanation may be technically faithful but incomprehensible to a user, or understandable while failing to represent the model's actual reasoning. Fairness claims likewise depend on selected groups, outcomes, metrics, data conditions, and normative judgments. Robustness, reliability, privacy, and security claims are similarly bounded by evaluated threats, scenarios, and assumptions rather than established as universal system properties.

Opacity, uncertainty, weak oversight, accountability gaps, evidence gaps, and argument gaps can interact. Missing evidence may be concealed by confident language, while a large evidence repository may still fail to support the stated claim. Unknown risks remain possible even after systematic analysis, especially when systems operate in changing socio-technical environments. The resulting methodology must therefore preserve a no-assurance-completeness boundary and a no-deployment-readiness boundary: organizing available reasoning cannot prove that every relevant failure, affected interest, or future operating condition has been identified.

Trustworthiness Claims and Acceptance Criteria

Trustworthiness claims should identify a specific property, system configuration, use case, population, environment, timeframe, and decision purpose. Autonomous-system assurance research demonstrates that technical, ethical, legal, and organizational concerns may create gaps between what engineering evidence establishes and what broader claims appear to assert [9]. A top-level assurance claim must therefore be decomposed into narrower subclaims so that unsupported transitions, omitted dimensions, and incompatible assumptions can be examined rather than absorbed into an undifferentiated statement that an AI system is trustworthy.

Claim decomposition should follow the lifecycle and architecture of the system. Development evidence, validation evidence, operational controls, human oversight, monitoring arrangements, and change-management processes may support different subclaims at different stages [10]. Decomposition should also distinguish properties such as explainability, fairness, robustness, reliability, accountability, and human control. These subclaims may interact, but they should not be merged merely because they are commonly grouped under trustworthy AI. Each requires its own scope, evidence needs, limitations, and relevant defeaters.

Acceptance criteria translate claims into observable conditions that can be evaluated. AI assurance encompasses multiple evidence classes and cannot be reduced to predictive accuracy or one technical validation activity [11]. Criteria may therefore concern test performance, external validity, documentation, human-review capability, monitoring coverage, escalation procedures, or the adequacy of governance controls. Acceptance thresholds may be used where they are defensible and contextually validated, but the methodology does not prescribe universal numerical limits. Criteria should instead state what evidence would count for, against, or as insufficient to support the relevant claim.

Responsibility must also be attached to claims and criteria. Safety and accountability remain institutional and human obligations even when decisions are heavily mediated by AI [12]. Claim owners should specify who formulates the claim, who supplies and reviews evidence, who can challenge assumptions, and who decides whether remaining uncertainty is acceptable. Stakeholder mapping should identify developers, operators, decision makers, users, reviewers, and affected groups whose knowledge or interests may reveal omitted risks or inappropriate acceptance criteria.

Every claim should include explicit limitations, assumptions, and generalization boundaries. Meeting a set of criteria may support a subclaim without establishing overall acceptability, and evidence collected for one model version, population, or decision cannot automatically justify another. Table 1 summarises how trustworthiness claims and acceptance criteria can be scoped, decomposed, bounded, and connected to evidence requirements in an AI assurance case.

Table 1. Trustworthiness Claims and Acceptance Criteria for AI Assurance Cases: Assurance Claims, Claim Decomposition, Subclaims, Scope Definition, Context Definition, System Boundaries, Use-Case Boundaries, Risk Context, Decision Context, Stakeholder Mapping, Acceptance Criteria, Acceptance Thresholds, Claim Assumptions, and Claim-Generalization Boundaries

Claim or criterion requirement	Core claim question	Required scope or context	Acceptance criterion function	Potential evidence type	Failure or overclaiming risk	Validation or governance need

Assurance claim	What proposition is the case intended to support?	Named system, use and decision	Defines the case objective	Integrated technical and governance evidence	Treating support as proof	Independent challenge
Trustworthiness claim	Which trustworthiness property is asserted?	Property, population, environment and time	Converts a general value into an assessable claim	Property-specific evidence	Global “trustworthy AI” label	Stakeholder review
Claim decomposition	Which lower-level propositions support the parent claim?	Explicit reasoning strategy	Makes complex claims reviewable	Evidence mapped to subclaims	Omitted dimensions or circularity	Structural review
Subclaim	What narrower assertion can be independently examined?	Component, actor, lifecycle stage or risk	Localizes evidence and uncertainty	Tests, evaluations or records	Fragmentation without integration	Dependency review
Scope definition	What is included and excluded?	Technical and socio-technical coverage	Limits interpretation	Architecture and process documentation	Hidden exclusions	Multidisciplinary review
Context definition	Under which conditions does the claim apply?	Environment, users, workflow and timeframe	Conditions claim applicability	Context records and external evaluation	Context-free generalization	Domain validation
System boundary	Which components and interfaces are included?	Model, data, software, hardware and humans	Identifies evidence and responsibility coverage	System documentation	Model-only assurance	Systems review
Use-case boundary	Which purpose and decisions are supported?	Intended, restricted and prohibited uses	Prevents transfer to unevaluated uses	Use-case documentation	Purpose expansion	Governance approval
Risk context	Which harms and affected interests matter?	Credible hazards, impacts and stakeholders	Links criteria to consequences	Risk and impact analyses	Narrow technical framing	Stakeholder challenge
Decision context	Which decision will the case inform?	Decision maker, alternatives and consequences	Aligns evidence with decision needs	Decision documentation	Treating evidence as decision-neutral	Decision-owner review

Stakeholder mapping	Whose knowledge and interests are relevant?	Developers, users, operators and affected groups	Informs claims and challenge	Participation and role evidence	Tokenistic inclusion	Transparent participation process
Acceptance criterion	What observable condition counts for or against the claim?	Claim-specific context	Makes evaluation explicit	Test, evaluation or process evidence	Vague or circular criteria	Domain review
Acceptance threshold	Where supported, what boundary distinguishes acceptable evidence?	Risk- and domain-specific conditions	Supports consistent assessment	Validated limits or expert judgment	Arbitrary universal threshold	Sensitivity and authority review
Claim assumption	Which condition is provisionally accepted?	Source, owner, duration and status	Makes dependence visible	Assumption evidence	Treating assumptions as facts	Verification and expiry review
Claim limitation	What does the claim not establish?	Evidential and inferential limits	Prevents overinterpretation	Limitation record	Caveats omitted or hidden	Mandatory review
Claim-generalization boundary	Can the claim extend beyond its evaluated context?	New versions, populations, purposes or environments	Blocks unsupported transfer	External validation	Universalizing local results	Reassessment after change
Acceptance-criteria sufficiency boundary	Does meeting listed criteria establish overall acceptability?	Whole system and decision context	Separates partial evidence from final judgment	Integrated review	Checklist-based approval	Governance deliberation

Evidence Quality and Traceability

Evidence traceability begins with identifying the origin, composition, intended use, processing history, and maintenance of relevant artefacts. Structured dataset documentation can expose collection choices, representation limits, recommended uses, and known exclusions [13]. Comparable records are needed for models, software versions, system configurations, evaluations, monitoring arrangements, and decisions. Documentation supports provenance and review, but its existence does not demonstrate that the recorded evidence is accurate, complete, unbiased, valid, or appropriate for the claim.

Evidence quality requires separate judgments about relevance, validity, credibility, completeness, uncertainty, and sufficiency. Auditing can organize the examination of criteria, processes, evidence, and institutional responsibilities, but its conclusions remain constrained by scope, access, competence, independence, and the assumptions built into the audit

design [14]. An assurance case should therefore record why evidence is admitted, what it establishes, what it does not establish, and how much confidence is justified. Auditability is a review capability, not an ethical or technical guarantee.

Transparency and traceability can strengthen accountability when they reveal how data, models, evaluations, and decisions are connected. However, transparency contributes to accountability only when identifiable actors possess obligations and authority to interpret records, answer challenges, correct failures, and provide remedies [15]. Decision provenance should consequently record who reviewed which case version, what evidence and alternatives were considered, which residual risks were acknowledged, and what conditions were attached to the decision. A traceable process without accountable ownership may remain practically unactionable.

Visibility must not be confused with knowledge. Revealing model artefacts or producing extensive documentation does not necessarily explain the social relationships, organizational incentives, downstream effects, or hidden dependencies that shape algorithmic outcomes [16]. Evidence trails can be complete in a technical sense while remaining substantively weak, irrelevant, or misleading. The methodology therefore includes a no-evidence-sufficiency guarantee boundary: neither documentation volume, provenance completeness, audit access, nor bidirectional traceability can prove that all necessary evidence exists or that a claim is adequately supported.

Evidence should be organized by function. Evaluation and test evidence address defined behavioral claims; simulation and benchmark evidence are admissible only when their assumptions and contextual relevance are justified. Monitoring and incident evidence support lifecycle reassessment. Human-oversight, explainability, fairness, robustness, reliability, and accountability evidence address distinct socio-technical claims and require appropriate technical, organizational, and human evaluation. Table 2 maps evidence quality and traceability requirements for AI assurance cases, including evidence relevance, sufficiency, validity, provenance, documentation, auditability, and evidence uncertainty.

Table 2. Evidence Quality and Traceability Requirements for AI Assurance Cases: Evidence Relevance, Sufficiency, Validity, Credibility, Completeness, Uncertainty, Provenance, Data Provenance, Model Provenance, Decision Provenance, Documentation, Evaluation Evidence, Test Evidence, Monitoring Evidence, Auditability, Evidence Trails, and Evidence-Sufficiency Boundaries

Evidence requirement	Core evidence question	Required evidence or process	Evidence function	Potential output	Failure or weakness risk	Traceability or audit need
<i>Evidence relevance</i>	Does the evidence directly address the scoped claim?	Claim–evidence fit assessment	Prevents irrelevant support	Relevance statement	Metric or evidence mismatch	Link to exact subclaim
<i>Evidence sufficiency</i>	Is available support adequate for the intended deliberation?	Coverage and triangulation review	Informs provisional confidence	Sufficiency judgment with limitations	Counting evidence instead of evaluating it	Decision-context record
<i>Evidence validity</i>	Does the method measure what the claim requires?	Construct, internal, external and ecological validity review	Limits invalid inference	Validity profile	Invalid proxy or poor transferability	Validation rationale

<i>Evidence credibility</i>	Is the source competent, independent and reproducible where relevant?	Source, conflict and reproducibility review	Calibrates evidential confidence	Credibility statement	Undisclosed conflicts or unverifiable evidence	Provenance record
<i>Evidence completeness</i>	Are planned evidence categories represented?	Requirement–evidence coverage analysis	Exposes omissions	Evidence-gap register	False completeness	Versioned requirement links
<i>Evidence uncertainty</i>	What uncertainty surrounds data, methods and conclusions?	Quantitative and qualitative characterization	Prevents certainty inflation	Uncertainty annotation	False precision	Link to residual-risk record
<i>Evidence provenance</i>	Where did the evidence originate and how was it transformed?	Source, custody, processing, ownership and version records	Establishes evidential history	Provenance record	Unknown origin or transformation	Versioned history
<i>Data provenance</i>	How were training, validation, test and monitoring data produced?	Collection, labeling, cleaning and sampling records	Supports data-fitness reasoning	Dataset lineage	Leakage, bias or undocumented processing	Dataset-to-model links
<i>Model provenance</i>	Which model, code, dependencies and settings produced the result?	Version and configuration records	Prevents artefact mismatch	Model lineage	Evidence linked to the wrong model	Model change log
<i>Decision provenance</i>	Who made the decision using which case and evidence?	Decision rationale, date, participants and conditions	Supports accountability	Decision record	Unowned or irreproducible decision	Link to case snapshot
<i>Documentation</i>	Are relevant activities and limitations recorded consistently?	Documentation protocol and ownership	Supports communication and inspection	Documentation set	Documentation theatre	Quality and currency review
<i>Model documentation</i>	Are purpose, architecture, training, limits and versions recorded?	Structured model record	Supports scope and replication	Model document	Generic or obsolete description	Model-version link

<i>Data documentation</i>	Are motivation, composition, collection, use and maintenance recorded?	Structured dataset record	Supports provenance and fitness analysis	Data document	Documentation mistaken for validation	Dataset-version link
<i>System documentation</i>	Are interfaces, dependencies, humans and operational controls described?	Architecture and workflow documentation	Supports system-boundary reasoning	System description	Model-only assessment	Configuration audit
<i>Decision documentation</i>	Are criteria, alternatives, dissent and conditions recorded?	Decision protocol	Supports accountable review	Decision memorandum	Unexplained approval	Decision audit trail
<i>Evaluation evidence</i>	Does evaluated behavior support the specified claim?	Context-appropriate evaluation	Supports behavioral subclaims	Evaluation report	Inadequate population or comparator	Protocol-to-result trace
<i>Test evidence</i>	Did specified tests execute under relevant conditions?	Test plan, environment, cases and results	Supports requirements assessment	Test report	Narrow coverage or leakage	Requirement-to-test trace
<i>Simulation evidence</i>	Does a simulation represent credible conditions?	Assumption and simulator validation	Examines difficult scenarios	Simulation report	Unrealistic scenarios or models	Assumption provenance
<i>Benchmark evidence</i>	Is the benchmark relevant to the intended use?	Dataset, metric and protocol appraisal	Supports bounded comparison	Benchmark analysis	Overfitting or weak ecological validity	Benchmark-version trace
<i>Monitoring evidence</i>	What does operation show over time?	Performance, drift, event and control monitoring	Supports continuing assurance	Monitoring review	Incomplete telemetry or delayed detection	Time-stamped lineage
<i>Incident evidence</i>	What do failures and near misses reveal?	Reporting, investigation and corrective-action process	Challenges previous claims	Incident record	Underreporting or defensive review	Link to affected claims

<i>Human oversight evidence</i>	Can designated people understand, intervene and escalate?	Authority, competence, workload and override evaluation	Supports oversight claims	Oversight assessment	Nominal human involvement	Role and intervention records
<i>Human review evidence</i>	Have qualified reviewers challenged evidence and arguments?	Independent or multidisciplinary review	Detects omissions and weak reasoning	Review report	Conflict or superficial review	Reviewer and disposition log
<i>Explainability evidence</i>	Are explanations faithful, stable and useful for the intended audience?	Technical and human evaluation	Supports bounded explanation claims	Explanation assessment	Plausible but misleading explanations	Method and audience trace
<i>Fairness evidence</i>	Which groups, definitions, outcomes and trade-offs were examined?	Disaggregated and context-sensitive analysis	Supports bounded fairness claims	Fairness assessment	Metric cherry-picking	Data, metric and decision links
<i>Robustness evidence</i>	What credible variation or perturbation was tested?	Stress, adversarial and boundary testing	Supports conditional robustness claims	Robustness analysis	Unrepresentative testing	Scenario-to-test trace
<i>Reliability evidence</i>	How consistently does the system perform across relevant conditions?	Repeatability, failure and degradation evidence	Supports reliability claims	Reliability analysis	Short observation period	Operational history
<i>Accountability evidence</i>	Are explanation, correction, escalation and remedy mechanisms established?	Role, complaint and decision records	Supports accountability claims	Accountability map	Responsibility diffusion	Actor-to-obligation trace
<i>Auditability</i>	Can an authorized reviewer reconstruct relevant activities and judgments?	Accessible records, identifiers and procedures	Enables challenge	Audit-ready package	Records without interpretive context	Audit protocol

<i>Evidence trail</i>	Is the path from source evidence to claim visible?	Bidirectional artefact links	Supports inspection and change analysis	Evidence map	Orphan evidence or broken links	Link-integrity review
<i>Audit trail</i>	Is the sequence of changes, reviews and decisions reconstructable?	Time-stamped versioned logs	Supports accountability	Audit log	Missing or mutable history	Access and integrity controls
Evidence-sufficiency boundary	Can organized evidence prove that all necessary evidence exists?	Explicit limitations and gap analysis	Prevents overclaiming	Boundary statement	Traceability mistaken for sufficiency	Independent challenge

Structured Arguments and Residual Risk

A structured assurance argument makes the inferential architecture of a trustworthiness case inspectable. A top-level claim is decomposed into subclaims, each connected to evidence through warrants, assumptions, contextual conditions, and traceable inference links. Model-based assurance research shows that explicit relationships among claims, arguments, evidence, and contextual artefacts can improve consistency and change analysis [17]. Defeaters and counterevidence should enter the same structure rather than remain outside it, while evidence gaps, argument gaps, uncertainty, and residual risks should remain visible wherever they qualify the conclusion. Figure 1 illustrates the structured AI assurance argument logic, showing how trustworthiness claims are decomposed into subclaims, connected to evidence, challenged by defeaters, and bounded by uncertainty and residual risk.

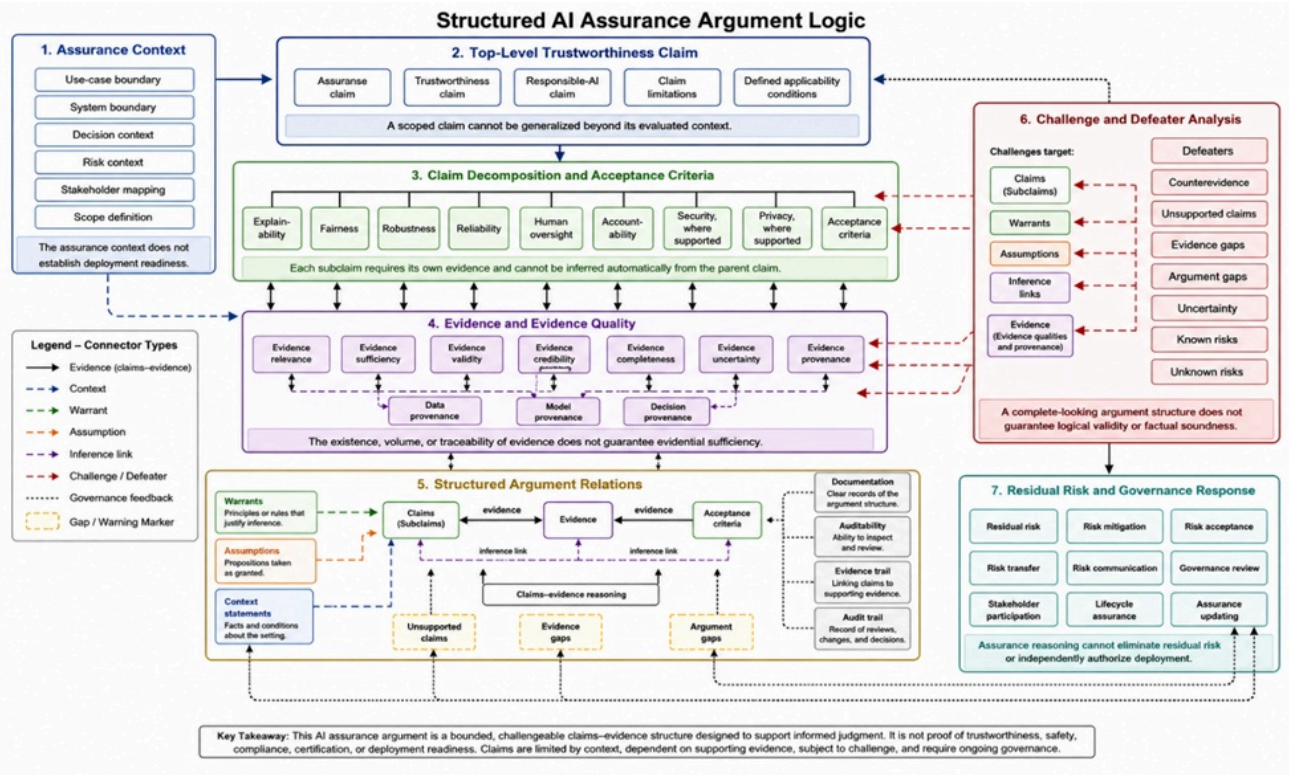


Figure 1. Structured AI Assurance Argument Logic

Alt-text description:

A conceptual structured argument diagram showing trustworthiness claims, subclaims, evidence, warrants, assumptions, context, defeaters, counterevidence, uncertainty, evidence gaps, argument gaps, and residual risk.

Confidence in an assurance argument should be distinguished from the factual truth of its top-level claim. Confidence may depend on evidence credibility, the plausibility of warrants, the stability of assumptions, and the extent to which uncertainty has been identified. Research on confidence cases shows that these sources of confidence can be represented explicitly rather than concealed within a binary conclusion [18]. Such representation may support calibrated review, but it cannot quantify every epistemic limitation or transform uncertainty into certainty.

Structured notation can clarify decomposition, inference, and evidence dependencies, particularly in complex autonomous systems [19]. Nevertheless, graphical completeness is not logical validity. A well-formed case may contain circular reasoning, unsupported warrants, context changes, weak evidence, or unacknowledged trade-offs. The proposed methodology therefore treats notation as a communication and review aid rather than as a preferred universal standard or a mechanism that proves the claim.

Defeater analysis asks what would undermine a claim, invalidate an assumption, weaken an inference, or reveal that evidence is incomplete. Assurance weakeners include unsupported claims, fallacious reasoning, inconsistent artefacts, ambiguous terminology, missing evidence, and failure to consider credible counterarguments [20]. Material counterevidence should be linked to the affected claim and resolved, mitigated, or retained as an explicit limitation. A case should not be closed merely because its supporting branch is well populated.

Independent assessment is necessary because case authors may normalize assumptions or overlook weaknesses that are apparent to reviewers with different expertise [21]. Review should address argument structure, domain validity, evidence quality, stakeholder concerns, and the treatment of known and unknown risk. Residual risk remains after available evidence and controls have been considered; documenting it does not mean that it has been accepted. Risk acceptance, transfer, or continued exposure requires a separate accountable decision.

Assurance-Based Deployment Decisions

An assurance case can inform a deployment decision by clarifying what is supported, what remains uncertain, and which conditions limit use. Continuing review is necessary because postdeployment evidence may reveal drift, emerging harms, or control failures that weaken earlier conclusions [22]. A decision gate should therefore examine the current case version, unresolved defeaters, residual risks, required mitigations, monitoring arrangements, and the consequences of proceeding, deferring, restricting, or discontinuing use.

Operational tools and governance methods can help translate responsible-AI principles into practical activities, but their functions and evidential strength vary substantially [23]. A checklist, assessment worksheet, documentation template, or audit process may contribute evidence without determining the decision. The methodology therefore avoids treating completion of a tool or satisfaction of isolated criteria as an automatic go/no-go mechanism.

Deployment reasoning must remain embedded in organizational governance. Governance practices determine who owns risks, who can challenge evidence, who authorizes exceptions, how conflicts are handled, and how changes are escalated [24]. Conditional deployment may be considered where supported by domain-specific governance, but only with explicit boundaries, verified controls, monitoring requirements, review dates, and rollback or suspension arrangements. The methodology does not recommend conditional deployment as a generally acceptable option.

Accountability requires a documented relationship between the assurance case and the people or institutions making the decision. Algorithmic accountability concerns obligations to explain, justify, review, correct, and accept responsibility for consequences rather than attributing responsibility to the algorithm [25]. Decision provenance should identify the case version reviewed, participants, dissent, alternatives, residual risks, imposed conditions, and the authority under which the decision was made.

Assurance boundaries should also recognize indirect, distributed, or development-stage harms that may be overlooked by narrowly technical evaluations [26]. Stakeholder participation may reveal affected interests, organizational incentives, or downstream dependencies that are not visible in model testing. **Table 3** connects structured arguments, defeaters, counterevidence, residual risk, decision gates, deployment boundaries, monitoring requirements, and lifecycle assurance in assurance-based deployment reasoning.

Table 3. Structured Arguments, Residual Risk, and Assurance-Based Deployment Decision Requirements: Claims-Evidence Reasoning, Warrants, Assumptions, Context, Inference Links, Defeaters, Counterevidence, Argument Gaps, Evidence Gaps, Residual Risk, Risk Mitigation, Risk Acceptance, Decision Gates, Deployment Boundaries, Monitoring Requirements, and Assurance Boundaries

Argument, risk, or decision requirement	Core assurance or decision question	Required evidence or assumption	Assurance or decision function	Potential output	Failure or residual risk	Review or monitoring need
<i>Claims-evidence reasoning</i>	Why does the evidence support the claim?	Evidence and explicit warrant	Connects observations to propositions	Justified inference	Evidence accumulation without relevance	Claim-level challenge
Warrant	What principle licenses the inference?	Scientific, engineering, normative, or contextual rationale	Explains support relation	Warrant statement	Hidden or invalid inference	Domain and logical review
Assumption	Which condition is provisionally accepted?	Source, owner, status, and expiry	Exposes dependency	Assumption register	Assumption treated as fact	Revalidation
Context	Where does the argument apply?	System, population, use, environment, and time	Bounds interpretation	Context statement	Unsupported generalization	Context review
Inference link	Which evidence supports which claim?	Traceable relation	Enables inspection and change analysis	Link map	Broken or ambiguous relation	Link-integrity review
Defeater	What could undermine the claim or inference?	Counter-scenario or conflicting consideration	Tests argument resilience	Defeater register	Confirmation bias	Independent challenge

Counterevidence	What evidence contradicts or qualifies the claim?	Negative or conflicting result	Prevents selective evidence use	Counterevidence record	Suppression or dismissal	Evidence review
Argument gap	Where is reasoning absent or invalid?	Structural and logical assessment	Identifies unsupported inference	Gap register	Gap concealed by formal structure	Reviewer disposition
Evidence gap	Which required support is missing?	Requirement–evidence comparison	Directs further work	Evidence plan	Weak proxy substituted for evidence	Materiality review
Unsupported claim	Which proposition lacks credible support?	Claim and evidence inventory	Prevents unjustified assertion	Flagged claim	Unsupported conclusion presented as established	Escalation or removal
Uncertainty	What remains uncertain in evidence, assumptions, and inference?	Quantitative or qualitative analysis	Calibrates interpretation	Uncertainty profile	False precision	Sensitivity review
Residual risk	What risk remains after evidence and controls?	Risk analysis, defeaters, uncertainty, and controls	Informs deliberation	Residual-risk record	Documented risk treated as acceptable	Authorized review
Risk mitigation	Which control reduces exposure?	Control rationale and verification evidence	Modifies risk	Mitigation plan	Ineffective control or new hazard	Control monitoring
Risk acceptance	Who may accept the remaining risk?	Authority, alternatives, rationale, and conditions	Records accountable judgment	Acceptance or rejection record	Analyst implicitly accepting risk	Reauthorization
Risk transfer	Is responsibility or consequence shifted?	Contractual and operational basis	Records allocation	Transfer arrangement	Risk displaced rather than reduced	Governance review
Risk communication	Are remaining risks understandable?	Audience-appropriate explanation	Supports informed judgment	Risk summary	Uncertainty minimized or obscured	Comprehension review
Decision gate	Has sufficient information been assembled for deliberation?	Current case, gaps, risks, and alternatives	Creates a formal review point	Proceed, restrict, defer, or stop	Mechanical checklist approval	Named accountable authority

Deployment boundary	Under which conditions may use occur?	Scope, controls, versions, users, and environment	Restricts exposure	Bounded-use conditions	Boundary expansion	Configuration and access monitoring
Conditional deployment	Where supported, can restricted use proceed?	Mitigations, monitoring, expiry, and rollback	Enables constrained use	Conditional decision	Conditions ignored or ineffective	Intensive monitoring
Go/no-go decision	Should use begin or continue?	Integrated case and organizational criteria	Requires explicit judgment	Go, no-go, or defer	Binary framing conceals alternatives	Human governance review
Monitoring requirement	Which indicators, triggers, owners, and responses are required?	Risk-to-monitor mapping	Tests continuing applicability	Monitoring plan	Incomplete telemetry or alert fatigue	Operational review
Assurance maintenance	Who maintains claims, evidence, and links?	Ownership and version-control process	Preserves case integrity	Maintained case	Obsolete assurance record	Case custodian
Assurance updating	What change requires reassessment?	Trigger conditions and impact analysis	Revises conclusions	Updated case version	Cosmetic update without re-evaluation	Change-control review
Governance review	Who challenges the case and decision?	Mandate, competence, independence, and access	Provides scrutiny	Review findings	Conflict or rubber-stamping	Independent oversight
Stakeholder participation	Whose knowledge and interests should be considered?	Stakeholder map and participation record	Broadens risk analysis	Challenge record	Tokenistic participation	Transparent response process
Human oversight requirement	Which interventions remain under human authority?	Competence, workload, authority, and override evidence	Defines control responsibility	Oversight conditions	Nominal human involvement	Workflow monitoring
Postdeployment monitoring	Does operation remain within the assurance boundary?	Outcomes, drift, incidents, and control status	Tests continuing support	Monitoring review	Delayed detection	Escalation and suspension process

Incident review	What must change after failure or near miss?	Incident evidence and affected-claim links	Challenges prior reasoning	Corrective action and case revision	Defensive or incomplete investigation	Independent review where appropriate
No argument-validity guarantee boundary	Does structured form guarantee soundness?	Explicit limitation	Prevents formalism overclaim	Boundary statement	Structure treated as proof	Logical and empirical challenge
No residual-risk elimination boundary	Can assurance remove all risk?	Known- and unknown-risk statement	Prevents zero-risk claims	Boundary statement	"Assured" interpreted as risk-free	Continuing review
No deployment-decision automation boundary	Can case status automatically authorize deployment?	Explicit human authority	Prevents automated approval	Boundary statement	Governance displaced by scoring	Human sign-off
No compliance claim boundary	Does the case establish compliance?	Separate qualified legal analysis	Prevents category error	Boundary statement	Case presented as legal proof	External legal review
No deployment-readiness boundary	Does favorable reasoning certify readiness?	Separate operational and institutional review	Prevents certification implication	Boundary statement	Support interpreted as approval	Authorized governance decision

Proposed AI Assurance-Case Methodology

The proposed methodology treats an AI assurance case as a lifecycle artefact that links explicit claims to development and operational evidence and must be revised when engineering artefacts, operating conditions, or system versions change [27, 28]. Its first stage defines the use case, assurance context, system boundary, decision boundary, risk context, relevant stakeholders, and accountable roles. These definitions determine which claims may be made and prevent evidence from being transferred silently across models, populations, environments, or decisions.

The second stage formulates trustworthiness claims, decomposes them into reviewable subclaims, and defines acceptance criteria. The third specifies evidence requirements before examining available results. Quality-assurance scholarship supports distinguishing data, model, evaluation, testing, monitoring, and process evidence rather than treating all artefacts as equivalent [29]. Each item is then assessed for relevance, validity, credibility, completeness, uncertainty, provenance, and traceability.

The fourth stage constructs the assurance argument by linking claims, evidence, warrants, assumptions, and context. The fifth challenges that argument through counterevidence, defeaters, evidence gaps, argument gaps, and unsupported claims. Safety-critical AI research reinforces the need to situate model evidence within the behavior of the larger system, including interfaces, human activity, and operational controls [30]. Material gaps should lead to new evidence, revised controls, narrowed claims, or a decision to defer judgment.

The sixth stage characterizes residual risks and distinguishes mitigation from acceptance. It then defines how the case may inform a decision gate, which deployment boundaries would be necessary, and what monitoring or reassessment

conditions would apply. Automation may assist with artefact linking or pattern-based case construction, but generated structures can contain malformed relations or context-sensitive errors and therefore require expert review [31]. The methodology prohibits automatic deployment decisions based on case completion, scores, or generated argument structures.

The final stage maintains the case through monitoring, incident review, system change, stakeholder challenge, and governance reassessment. Interacting intelligent systems may require connected cases because one system’s boundary, evidence, or failure can affect another system operating in a shared environment [32]. **Table 4** presents the proposed AI assurance-case methodology linking assurance context, claims, acceptance criteria, evidence quality, traceability, structured argumentation, residual risk, deployment boundaries, monitoring, and assurance updating.

Table 4. Proposed AI Assurance-Case Methodology for Trustworthy Intelligent Systems: Assurance Context, System Boundaries, Stakeholder Mapping, Trustworthiness Claims, Claim Decomposition, Acceptance Criteria, Evidence Requirements, Evidence Quality, Evidence Traceability, Structured Arguments, Assumptions, Defeaters, Evidence Gaps, Residual Risk, Deployment Boundaries, Monitoring, Governance Review, Assurance Updating, and Deployment-Readiness Boundaries

Methodology stage	Core purpose	Required input	Assurance function	Potential output	Validation or governance need	Failure risk	Fe me
<i>Use-case and assurance-context definition</i>	Establish intended purpose and operating conditions	Use description, population, environment, workflow	Bounds all later reasoning	Context statement	Domain and stakeholder review	Idealized or incomplete context	Re cc (
<i>System and decision-boundary definition</i>	Identify included components and decision purpose	Architecture, interfaces, human roles, decision mandate	Prevents model-only assurance	System and decision map	Systems and governance review	Hidden dependencies	C imp
<i>Stakeholder and responsibility mapping</i>	Identify affected interests and accountable actors	Stakeholder and role analysis	Assigns ownership and challenge rights	Responsibility map	Participation and conflict review	Responsibility diffusion	Up rol
<i>Trustworthiness claim formulation</i>	Convert general principles into propositions	Risk context and stakeholder requirements	Creates assessable claims	Scoped claim set	Claim-owner review	Vague global claim	Refi after
Claim decomposition	Break complex claims into evaluable subclaims	Top-level claim and reasoning strategy	Localizes evidence and uncertainty	Claim hierarchy	Dependency review	Omission or circularity	Re s
Acceptance criteria definition	State what counts for or	Claim, context, risk	Makes evaluation	Criterion register	Domain validation	Arbitrary or circular	Re ev

	against each claim	and evidence feasibility	explicit			criteria	
Evidence requirement specification	Identify evidence classes in advance	Claims, criteria and lifecycle phase	Reduces post hoc selection	Evidence plan	Multidisciplinary review	Cherry-picking	Gap
Evidence quality assessment	Evaluate relevance, validity, credibility and uncertainty	Evidence artefacts and appraisal process	Calibrates evidential weight	Quality profile	Qualified review	False precision	Re
Evidence provenance and traceability review	Establish origin, version, transformation and links	Data, model, test, monitoring and decision records	Supports inspection and change analysis	Provenance and traceability map	Audit review	Broken lineage or wrong version	Lin
Structured argument construction	Connect claims, evidence, warrants and assumptions	Claim hierarchy and assessed evidence	Makes inference reviewable	Structured argument	Logical and domain review	Formal appearance masking weakness	I
Assumption and context documentation	Expose conditional dependencies	Assumption source, owner, status and expiry	Prevents hidden premises	Assumption register	Verification review	Assumptions treated as facts	Re
Defeater and counterevidence review	Seek reasons the case may fail	Negative evidence, incidents and alternative explanations	Counteracts confirmation bias	Defeater register	Selective evidence use	Independent challenge	F
Evidence- and argument-gap identification	Locate missing support and invalid inference	Evidence plan, traceability map and argument	Prevents false completeness	Gap register	Gap minimized or hidden	Materiality review	Ne
Residual-risk assessment	Characterize remaining known and unknown risks	Uncertainty, defeaters and control evidence	Informs decision deliberation	Residual-risk profile	Risk record treated as acceptance	Risk-owner review	Mon
Risk-mitigation and acceptance review	Separate technical controls from accountable judgment	Residual-risk profile and control evidence	Records reduced and remaining exposure	Mitigation and decision inputs	Analyst implicitly accepts risk	Authorized decision authority	Ver

Deployment-decision boundary definition	State how the case may inform a decision	Governance mandate and decision context	Prevents certification overreach	Decision-use statement	Governance approval	Case treated as authorization	Re n c
Monitoring requirement definition	Link risks and assumptions to observable indicators	Risk register and change scenarios	Supports continuing assurance	Monitoring and escalation plan	Unobservable or ownerless indicators	Operational review	M feec
Assurance maintenance and updating	Keep the case synchronized with change	Versioning, monitoring, incidents and triggers	Prevents obsolescence	Updated case version	Change-control review	Cosmetic updating	Sc ar reas
Governance review and stakeholder challenge	Subject reasoning to accountable scrutiny	Case, limitations, dissent and access arrangements	Tests adequacy and legitimacy	Review findings	Conflict or tokenistic participation	Independence safeguards	Re alte e
No evidence-sufficiency guarantee boundary	Preserve limits of available evidence	Quality and gap assessment	Prevents sufficiency overclaim	Boundary statement	Evidence volume treated as proof	Independent review	Rea
No argument-validity guarantee boundary	Preserve logical and factual uncertainty	Structured argument and challenge record	Limits formalism	Boundary statement	Well-formed but unsound case	External challenge	D
No residual-risk elimination boundary	Preserve known and unknown risk	Residual-risk statement	Prevents zero-risk language	Boundary statement	Assurance interpreted as risk-free	Lifecycle review	I fe
No assurance-completeness boundary	Acknowledge incomplete system and harm knowledge	Scope, exclusions and unknowns	Prevents comprehensive coverage claims	Boundary statement	Unrecognized harm	Stakeholder challenge	Scop
No compliance claim boundary	Separate assurance from legal determination	Qualified analysis outside the method	Prevents legal overreach	Boundary statement	Case represented as compliance proof	External legal review	Up lega
No deployment-readiness boundary	Separate justification from authorization	Operational, legal, ethical and institutional review	Prevents certification implication	Boundary statement	Case treated as readiness certificate	Authorized human decision	Rea at

Figure 2 presents the proposed AI assurance-case methodology, integrating assurance context, trustworthiness claims, acceptance criteria, evidence quality, evidence traceability, structured arguments, residual-risk reasoning, deployment boundaries, monitoring, and assurance updating.

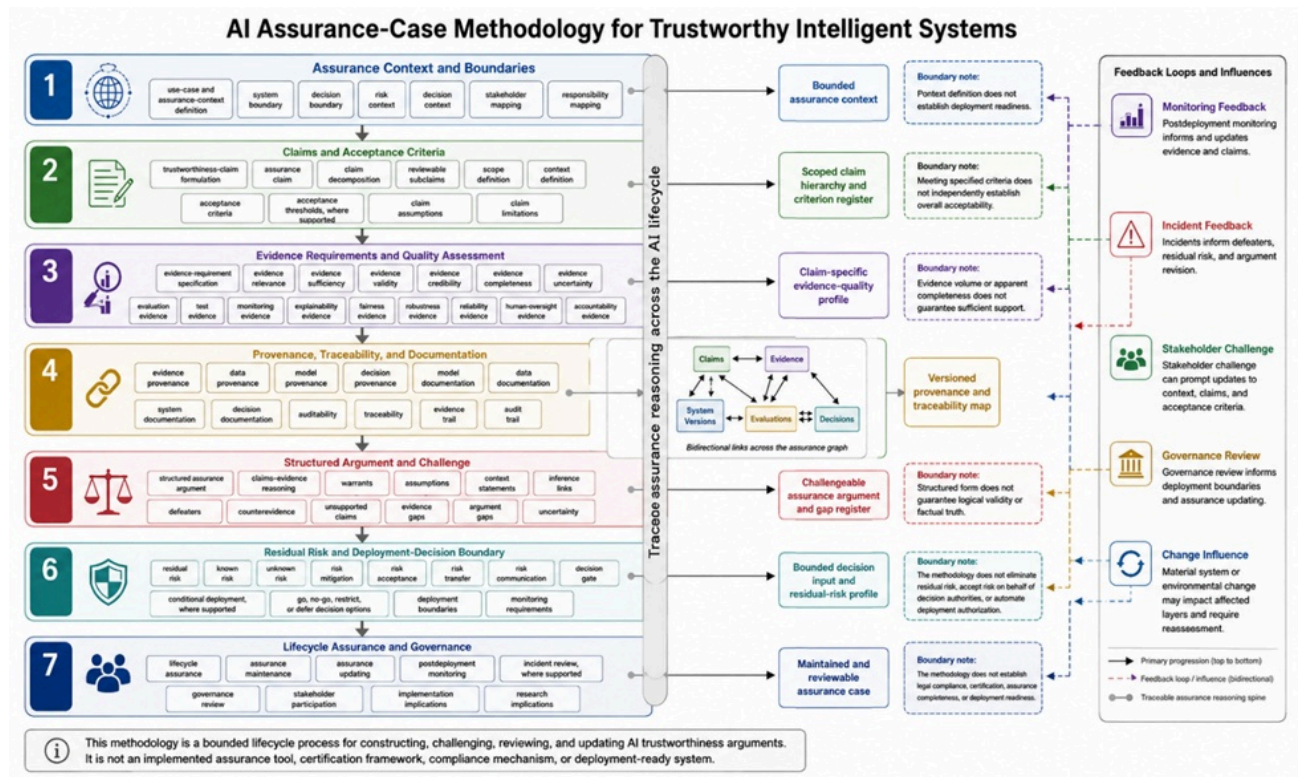


Figure 2. AI Assurance-Case Methodology for Trustworthy Intelligent Systems

Alt-text description:

A conceptual AI assurance-case methodology showing assurance context, system boundaries, stakeholders, claims, acceptance criteria, evidence, traceability, structured arguments, defeaters, residual risks, deployment boundaries, monitoring, governance review, and assurance updating.

Implementation and Research Implications

Implementation research should first examine whether assurance claims and acceptance criteria are understandable, appropriately scoped, and sensitive to context. Lifecycle assurance research indicates that evidence requirements vary across development and operation [10]. Future work should therefore evaluate claim templates, decomposition strategies, threshold rationales, and methods for identifying omitted dimensions without converting the methodology into a fixed checklist.

Evidence research should compare approaches for appraising relevance, validity, credibility, completeness, and uncertainty. Dataset documentation provides a basis for provenance but does not establish fitness [13]. Further studies should test whether traceability links are semantically correct, whether change-impact mechanisms identify all affected claims, and whether evidence-quality judgments remain consistent across reviewers and domains.

Argument review requires methods for eliciting defeaters, evaluating warrants, communicating uncertainty, and identifying weaknesses without creating misleading assurance scores. Mapped assurance weakeners provide a starting point for

systematic challenge [20]. Human-oversight, explainability, fairness, robustness, reliability, and accountability evidence also require distinct evaluation methods because technical indicators cannot alone demonstrate meaningful control, understandable explanation, equitable outcomes, or institutional responsibility.

Lifecycle research should investigate how drift, incidents, stakeholder challenges, and organizational change trigger assurance updates. Assurance-centred engineering demonstrates the value of connecting evolving cases to changing artefacts [28]. These implications remain conceptual research directions rather than legal advice, compliance guidance, audit instruction, certification instruction, procurement guidance, model- or platform-selection advice, safety certification, or deployment-decision instruction. Domain-specific institutions must determine how, whether, and by whom the methodology may be evaluated or adapted.

Conclusion

This article has proposed an AI assurance-case methodology for constructing bounded trustworthiness cases through explicit claims, acceptance criteria, evidence-quality assessment, provenance, traceability, structured arguments, assumptions, defeaters, uncertainty, residual-risk reasoning, deployment boundaries, monitoring, and assurance updating. The methodology replaces unsupported labels and isolated metrics with transparent relationships among what is claimed, why it may be supported, where the support is limited, what could defeat the reasoning, and which risks remain. It may improve the clarity, reviewability, and accountability of trustworthiness deliberation, but it does not prove safety, fairness, legality, ethical sufficiency, compliance, operational readiness, or deployment acceptability. Its contribution is a conceptual architecture for methodology development that requires domain-specific evidence, stakeholder challenge, governance review, and continuing lifecycle evaluation.

Acknowledgements

None

Conflict of interest

None

Financial support

None

Ethics statement

None

Received: 15 Sep 2025

Revised: 14 Oct 2025

Accepted: 26 Nov 2025

Published online: 15 March 2026

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and

distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

- Jobin A, Ienca M, Vayena E. The global landscape of AI ethics guidelines. *Nat Mach Intell*. 2019;1(9):389-99.
<https://doi.org/10.1038/s42256-019-0088-2>.
- Floridi L, Cowls J, Beltrametti M, Chatila R, Chazerand P, Dignum V, et al. AI4People—an ethical framework for a good AI society: opportunities, risks, principles, and recommendations. *Minds Mach*. 2018;28(4):689-707.
<https://doi.org/10.1007/s11023-018-9482-5>.
- Thiebes S, Lins S, Sunyaev A. Trustworthy artificial intelligence. *Electron Mark*. 2021;31(2):447-64.
<https://doi.org/10.1007/s12525-020-00441-4>.
- Shneiderman B. Human-centered artificial intelligence: reliable, safe & trustworthy. *Int J Hum Comput Interact*. 2020;36(6):495-504.
<https://doi.org/10.1080/10447318.2020.1741118>.
- Lu J, Liu A, Dong F, Gu F, Gama J, Zhang G. Learning under concept drift: a review. *IEEE Trans Knowl Data Eng*. 2019;31(12):2346-63.
<https://doi.org/10.1109/TKDE.2018.2876857>.
- Kelly CJ, Karthikesalingam A, Suleyman M, Corrado G, King D. Key challenges for delivering clinical impact with artificial intelligence. *BMC Med*. 2019;17(1):195.
<https://doi.org/10.1186/s12916-019-1426-2>.
- Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, Benetot A, Tabik S, Barbado A, et al. Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf Fusion*. 2020;58:82-115.
<https://doi.org/10.1016/j.inffus.2019.12.012>.
- Mehrabi N, Morstatter F, Saxena NA, Lerman K, Galstyan AG. A survey on bias and fairness in machine learning. *ACM Comput Surv*. 2021;54(6):1-35.
<https://doi.org/10.1145/3457607>.
- Burton S, Habli I, Lawton T, McDermid JA, Morgan PDJ, Porter ZLM. Mind the gaps: assuring the safety of autonomous systems from an engineering, ethical, and legal perspective. *Artif Intell*. 2020;279:103201.
<https://doi.org/10.1016/j.artint.2019.103201>.
- Ashmore R, Calinescu R, Paterson C. Assuring the machine learning lifecycle: desiderata, methods, and challenges. *ACM Comput Surv*. 2021;54(5):1-39.
<https://doi.org/10.1145/3453444>.
- Batarseh FA, Freeman L, Huang CH. A survey on artificial intelligence assurance. *J Big Data*. 2021;8(1):60.
<https://doi.org/10.1186/s40537-021-00445-7>.
- Habli I, Lawton T, Porter Z. Artificial intelligence in health care: accountability and safety. *Bull World Health Organ*. 2020;98(4):251-6.
<https://doi.org/10.2471/BLT.19.237487>.
- Gebru T, Morgenstern J, Vecchione B, Vaughan JW, Wallach H, Daumé H 3rd, et al. Datasheets for datasets. *Commun ACM*. 2021;64(12):86-92.
<https://doi.org/10.1145/3458723>.

Mökander J, Morley J, Taddeo M, Floridi L. Ethics-based auditing of automated decision-making systems: nature, scope, and limitations. *Sci Eng Ethics*. 2021;27(4):44.
<https://doi.org/10.1007/s11948-021-00319-4>.

de Laat PB. Algorithmic decision-making based on machine learning from big data: can transparency restore accountability? *Philos Technol*. 2018;31(4):525-41.
<https://doi.org/10.1007/s13347-017-0293-z>.

Ananny M, Crawford K. Seeing without knowing: limitations of the transparency ideal and its application to algorithmic accountability. *New Media Soc*. 2018;20(3):973-89.
<https://doi.org/10.1177/1461444816676645>.

Wei R, Kelly TP, Dai X, Zhao S, Hawkins R. Model based system assurance using the Structured Assurance Case Metamodel. *J Syst Softw*. 2019;154:211-33.
<https://doi.org/10.1016/j.jss.2019.05.013>.

Wang R, Guiochet J, Motet G, Schön W. Modelling confidence in railway safety case. *Saf Sci*. 2018;110:286-99.
<https://doi.org/10.1016/j.ssci.2017.11.012>.

Chelouati M, Boussif A, Beugin J, El Koursi EM. Graphical safety assurance case using Goal Structuring Notation (GSN)—challenges, opportunities and a framework for autonomous trains. *Reliab Eng Syst Saf*. 2023;230:108933.
<https://doi.org/10.1016/j.ress.2022.108933>.

Shahandashti KK, Belle AB, Lethbridge TC, Odu O, Sivakumar M. A PRISMA-driven systematic mapping study on system assurance weakeners. *Inf Softw Technol*. 2024;175:107526.
<https://doi.org/10.1016/j.infsof.2024.107526>.

Luo Y, van den Brand MJG, Li Z, Khabbaz Saberi A. A systematic approach and tool support for GSN-based safety case assessment. *J Syst Archit*. 2017;76:1-16.
<https://doi.org/10.1016/j.sysarc.2017.04.001>.

Mökander J, Axente M, Casolari F, Floridi L. Conformity assessments and post-market monitoring: a guide to the role of auditing in the proposed European AI regulation. *Minds Mach*. 2022;32(2):241-68.
<https://doi.org/10.1007/s11023-021-09577-4>.

Morley J, Floridi L, Kinsey L, Elhalal A. From what to how: an initial review of publicly available AI ethics tools, methods and research to translate principles into practices. *Sci Eng Ethics*. 2020;26(4):2141-68.
<https://doi.org/10.1007/s11948-019-00165-5>.

Papagiannidis E, Enholm IM, Dremel C, Mikalef P, Krogstie J. Toward AI governance: identifying best practices and potential barriers and outcomes. *Inf Syst Front*. 2023;25(1):123-41.
<https://doi.org/10.1007/s10796-022-10251-y>.

Horneber D, Laumer S. Algorithmic accountability. *Bus Inf Syst Eng*. 2023;65(6):723-30.
<https://doi.org/10.1007/s12599-023-00817-8>.

Thomas C, Roberts H, Mökander J, Tsamados A, Taddeo M, Floridi L. The case for a broader approach to AI assurance: addressing “hidden” harms in the development of artificial intelligence. *AI Soc*. 2025;40(3):1469-84.
<https://doi.org/10.1007/s00146-024-01950-y>.

Paterson C, Hawkins RD, Picardi C, Jia Y, Calinescu R, Habli I. Safety assurance of machine learning for autonomous systems. *Reliab Eng Syst Saf*. 2025;264(Pt A):111311.
<https://doi.org/10.1016/j.ress.2025.111311>.

Wei R, Foster SD, Mei H, Yan F, Yang R, Habli I, et al. ACCESS: assurance case centric engineering of safety-critical systems. *J Syst Softw*. 2024;213:112034.

<https://doi.org/10.1016/j.jss.2024.112034>.

Ogrizović M, Drašković D, Bojić D. Quality assurance strategies for machine learning applications in big data analytics: an overview. *J Big Data*. 2024;11(1):156.

<https://doi.org/10.1186/s40537-024-01028-y>.

Perez-Cerrolaza J, Abella J, Borg M, Donzella C, Cerquides J, Cazorla FJ, et al. Artificial intelligence for safety-critical systems in industrial and transportation domains: a survey. *ACM Comput Surv*. 2024;56(7):1-40.

<https://doi.org/10.1145/3626314>.

Odu O, Belle AB, Wang S, Kpodjedo S, Lethbridge TC, Hemmati H. Automatic instantiation of assurance cases from patterns using large language models. *J Syst Softw*. 2025;222:112353.

<https://doi.org/10.1016/j.jss.2025.112353>.

Vierhauser M, Bayley S, Wyngaard J, Xiong W, Cheng J, Huseman J, et al. Interlocking safety cases for unmanned autonomous systems in shared airspaces. *IEEE Trans Softw Eng*. 2021;47(5):899-918.

<https://doi.org/10.1109/TSE.2019.2907595>.