

A Unified Evaluation Framework for Trustworthy Large Language Models, Agentic AI, and Multimodal Systems

Shaina Raza, Ahmed Y. Radwan, Imran Liaquat, and Kathryn Hume

Abstract—Benchmark scores alone provide an incomplete basis for assessing the trustworthiness of modern artificial intelligence systems. Large language models (LLMs), agentic systems, and multimodal models (MLLMs) require different forms of assessment, yet their evaluation evidence must remain interpretable for development and oversight. We propose a unified framework that connects output-level, trajectory-level, and cross-modal assessment through eight trustworthiness dimensions: capability, robustness, safety, fairness, transparency, governance, oversight, and efficiency. The framework preserves system-specific metrics while mapping native measurements to common performance bands, accompanied by uncertainty estimates and traceable evidence. A meta-evaluation layer examines the validity, reliability, and reproducibility of the evaluation itself. Multidimensional profiles expose strengths and weaknesses, while safety-critical overrides prevent aggregate scores from masking critical failures. Mappings to governance frameworks, international standards, and European Union regulatory requirements connect technical assessment with oversight needs. The framework provides a structured basis for assessing both system performance and the credibility of the evidence supporting it, with empirical validation across deployment contexts remaining an essential next step.

Index Terms—AI evaluation, large language models, agentic systems, multimodal models, trustworthy AI, benchmarking, AI governance, EU AI Act.

I. INTRODUCTION

MODERN artificial intelligence (AI) evaluation is fragmented across model classes. Large language models (LLMs) are commonly assessed through output-level benchmarks, agentic systems require trajectory-level evaluation of planning, tool use, recovery, and oversight, and multimodal systems introduce cross-modal grounding and consistency requirements. These differences make it difficult to compare evaluation evidence across systems and to determine whether reported scores are sufficiently valid, reproducible, and informative for governance decisions.

This article addresses this gap through a unified evaluation framework for LLM, agentic, multi-agent, and multimodal systems. The framework combines a common trustworthiness spine with model-specific assessment tracks and a meta-evaluation layer that evaluates the quality of the evaluation process itself. It is intended to support both AI developers and evaluators while providing evidence that is interpretable

by governance and assurance stakeholders and the broader scientific community.

The framework covers three primary model classes: (1) LLMs, including fine-tuned open and API-served models; (2) agentic and multi-agentic systems that plan and act over multiple steps, including tool-using and retrieval-augmented (RAG) agents; and (3) multimodal models such as vision language models (VLMs), omni modality, speech models, and models that process or generate multiple modalities. The framework operates primarily at the level of the deployable AI system, including its prompts, tools, interfaces, and guardrails. It can be applied during development as well as to systems already in use. Training-time decisions are outside its primary scope except where they affect observable evaluation evidence. The framework is not intended to replace legal conformity assessment, and domains for which sufficiently mature evaluation instruments are not yet defined, such as embodied robotics and real-time control, remain outside the current scope.

This framework is based on trustworthiness in AI, where trustworthiness is understood as the ability to meet stakeholders’ expectations in a verifiable manner. Consistent with ISO/IEC TS 5723:2022 [1], the characteristics relevant to trustworthiness depend on the system and service being evaluated, the technologies, data, and processes involved, and the applicable context or sector. The overarching goal of this work is therefore to operationalize this context-sensitive concept through eight evaluation dimensions and the supporting evidence required to make evaluation results independently interpretable and verifiable.

Contributions This article makes three main contributions. (1) It introduces a unified eight-dimensional evaluation spine for LLMs, agentic, multi-agent, and multimodal systems while preserving their distinct units of assessment: outputs, trajectories, and cross-modal relationships. (2) It develops a common reporting architecture that combines native metrics with normalization, uncertainty reporting, not-applicable rules, safety-critical overrides, and a meta-evaluation layer for assessing the validity and reproducibility of the evaluation itself. (3) It operationalizes the framework through model-specific evaluation instruments and evidence requirements and maps these components to AI governance and regulations, such as EU AI Act, ISO standards and various AI Risk Management Framework (RMF) [2].

S. Raza, A. Y. Radwan, I. Liaquat, and K. Hume are with the Vector Institute for Artificial Intelligence, Toronto, ON M5G 0C6, Canada. Email: {shaina.raza, ahmed.radwan, imran.liaquat, kathryn.hume}@vectorinstitute.ai
Corresponding author: Shaina Raza.

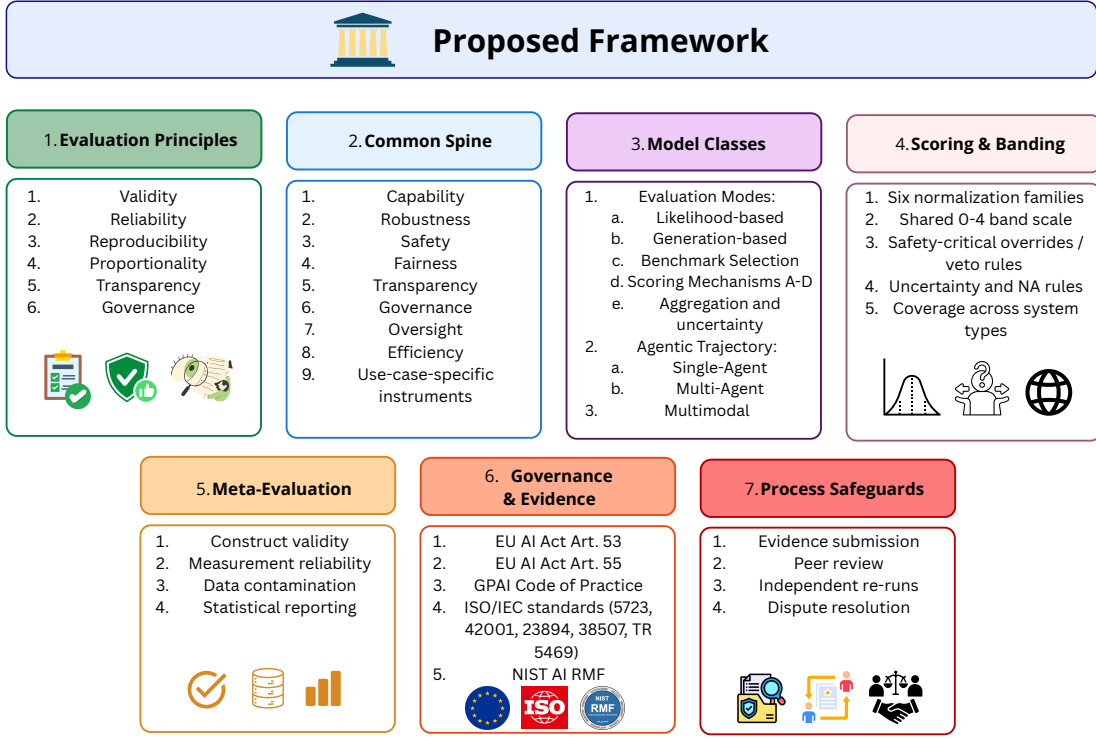


Fig. 1: Common evaluation dimensions in the framework.

II. RELATED WORK

Evaluation Frameworks for Language Models Evaluation of LLMs has traditionally focused on individual capabilities or failure modes through task-specific benchmarks. Common evaluation dimensions include factuality, reasoning, knowledge, code generation, toxicity, adversarial robustness, and calibration. Benchmarks such as TruthfulQA [3] and HaluEval [4] target factuality and hallucination; GSM8K [5], BIG-Bench Hard [6], and ARC-Challenge [7] assess reasoning; MMLU and MMLU-Pro [8] measure broad knowledge. Safety-oriented evaluation is commonly conducted using datasets such as ToxiGen [9], HarmBench [10], and others.

Multimodal evaluation needs to assess both individual modalities and their interactions. Existing benchmarks cover visual question answering (VQA), captioning, retrieval, grounding, OCR, speech recognition, and cross-modal hallucination using datasets such as VQAv2 [11], MS-COCO [12], HumanBench [13] and others. These benchmarks are largely task-specific and use different native metrics, comparing trustworthiness properties across multimodal, text-only, and agentic systems remains difficult.

Agentic and Multi-Agent Evaluation Agentic AI systems introduce evaluation requirements that extend beyond the correctness of a final response. Their behaviour unfolds through trajectories involving planning, tool use, intermediate decisions, environment interactions, and recovery from errors. Consequently, recent agent evaluation work [14] increasingly considers task completion, tool-use correctness, trajectory efficiency, error recovery, and adherence to constraints in addition to final task success. These efforts demonstrate that agentic evaluation must move from output-level assessment toward

trajectory- and interaction-level analysis. However, agentic metrics are often developed independently of conventional LLM evaluation and broader governance requirements.

Regulatory and Standards-Based Assessment A parallel body of work addresses AI evaluation from a governance and risk-management perspective. The European Union Artificial Intelligence Act 2026 [15] establishes obligations related to documentation, transparency, model evaluation, adversarial testing, risk management, cybersecurity, and incident reporting, with additional requirements for general-purpose AI models presenting systemic risk. The GPAI provides further guidance for implementing these obligations. International standards provide complementary perspectives. ISO/IEC TS 5723 [1] defines AI trustworthiness concepts; ISO/IEC 42001 [16] addresses organizational AI management systems; ISO/IEC 23894 [17] provides guidance for AI risk management; and ISO/IEC 38507 [18] addresses governance of organizational AI use. The NIST AI RMF similarly organizes AI risk management around the GOVERN, MAP, MEASURE, and MANAGE functions. These frameworks provide important governance structures, but they typically do not prescribe a unified set of operational metrics spanning language, agentic, multi-agent, and multimodal systems. Conversely, benchmark-oriented evaluation research often provides detailed metrics without connecting them systematically to governance evidence and assurance processes.

Positioning of This Work Rather than introducing another benchmark suite, we organize existing and emerging instruments around eight trustworthiness dimensions. LLMs are assessed at the output level, agentic and multi-agent systems at the trajectory level, and multimodal systems at the cross-

modal level. A meta-evaluation layer further examines validity, reliability, contamination, and statistical quality, enabling heterogeneous measurements to remain model-appropriate while supporting a common evaluation and governance framework.

III. EVALUATION PRINCIPLES

Our proposed framework rests on six principles. They define what constitutes a rigorous measurement, and every scoring decision in Section IV is guided by these principles.

- 1) **Validity:** each metric measures the construct it is intended to assess.
- 2) **Reliability:** measurements are stable across runs and raters. This principle concerns measurement reliability; reliability as a property of the system under evaluation is assessed under Robustness (Table I).
- 3) **Reproducibility:** sufficient provenance is recorded to support independent reproduction.
- 4) **Proportionality:** the depth of evaluation scales with the model capability and risk, consistent with the EU’s risk-based approach (Section V).
- 5) **Transparency:** assumptions, limitations, and negative results are reported.
- 6) **Governance:** every reported result is traceable to an accountable source and reproducible from an auditable evidence trail, and each model is accompanied by its governance artifacts (documentation, training-data summary, access and incident controls).

A. From Native Scores to a Shared Band

In this framework, the scoring proceeds in two steps. Each benchmark is first evaluated with its own native method, and the native score is then mapped to a common 0 - 4 band to support consistent interpretation across heterogeneous instruments. The mapping rule depends on the metric and is applied by one of six normalization families. The families, their thresholds, and the per-metric assignments are defined in Section IV-D and specified in each descriptor (Appendix A), so they are not restated here.

Band	Reading
0	Unacceptable
1	Below expectations
2	Adequate
3	Good
4	Excellent

B. The Eight-Dimensional Profile

We report an eight-dimensional profile (Figure 2) rather than aggregating all dimensions into a single score. A single aggregate score may be disproportionately influenced by capability benchmarks and could hide trade-offs such as strong performance paired with weaker safety. A stakeholder interpreting the results, whether a downstream integrator, an EU reviewer, or a consortium decision board, interprets the profile at the level of individual dimensions. When a specific decision needs a summary, for example, a deployment decision, a

weighted aggregation may be computed, but the weights must be declared before scoring, recorded in the evidence package, and justified with respect to the intended use case. The framework prescribes no default weight vector, because the appropriate weights depend on context.

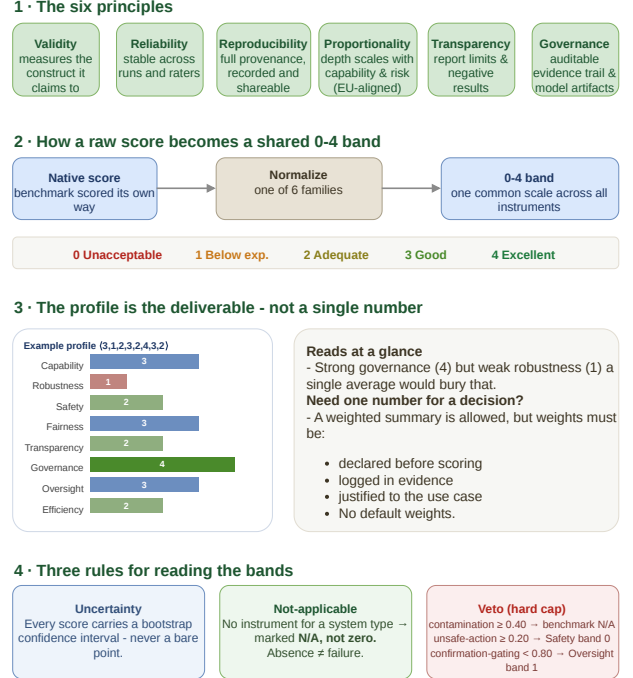


Fig. 2: **The evaluation profile.** **Panel 1:** the six measurement principles. **Panel 2:** the two-step path from a native benchmark score to the shared 0 - 4 band. **Panel 3:** the eight-dimensional profile, reported per dimension rather than collapsed into a single average. **Panel 4:** the three rules governing reported bands (uncertainty, not-applicable, and veto).

1) Interpretation of Band Scores: Three rules govern the reported bands, and each is defined in full in Section IV-D.

Uncertainty Every score carries a bootstrap confidence interval, rather than a point estimate alone.

Not-applicable A dimension with no applicable instrument for a given system type is marked N/A rather than scored zero, so absence of measurement is not interpreted as failure (coverage matrix, Table IV).

Veto Three safety-critical instruments override normal banding. Contamination ≥ 0.40 voids the affected benchmark, whose construct is then marked N/A for that submission. Unsafe-action rate ≥ 0.20 caps the Safety dimension at band 0, and confirmation-gating < 0.80 caps the Oversight dimension at band 1 (Section IV-D).

Model outputs can be evaluated using two scoring modes: from its log-probabilities (likelihood-based, deterministic) or from its generated text (generation-based, decoding-dependent). The default protocol prioritizes likelihood-based scoring when available. The rule, its rationale, and the mode each instrument uses are specified in Section IV-B1.

IV. UNIFIED EVALUATION FRAMEWORK

The proposed framework evaluates LLM, agentic, multi-agent, and multimodal systems using a common eight-dimensional profile, model-specific evaluation tracks, and a meta-evaluation layer that assesses the quality of the evaluation process itself. The goal is to support consistent interpretation and independent scrutiny of reported results. The following sections describe each component in detail.

A. Common Evaluation Spine

The framework adopts a *common spine* of evaluation dimensions applied uniformly to every model, as listed in Table I. The spine is fixed rather than selected on a per-model basis, ensuring that systems with different modalities, capabilities, and deployment contexts remain interpretable within a common evaluation structure and that each evaluation dimension is explicitly considered. The spine defines *what* must be evaluated, not *how* it should be measured. Each dimension is operationalized using evaluation instruments appropriate for the model under assessment, allowing the underlying metrics to vary while preserving a consistent evaluation structure. Safety and Governance follow a principle of *proportionality*: the rigor and extent of evidence required increase with both model capability and deployment risk, but every dimension is considered for every system, with non-applicable dimensions explicitly marked N/A.

Results are reported as a multidimensional profile rather than a single aggregate score, preventing strong performance in one area from obscuring weaknesses in others. Across the framework, accuracy is treated as a general measurement concept rather than as a single evaluation instrument. It is defined as the measure of closeness of results of observations, computations, or estimates to the true values or the values accepted as being true [19]. Each accuracy-based metric operationalizes this definition against the reference labels, reference outputs, task-specific oracle, or environment state.

a) Use-Case-Specific Evaluation: The common spine remains the same across systems, but some evaluation needs depend on how and where an AI system is used. This is especially important for agentic AI systems operating over longer, multi-turn interactions. For example, a personal-care agent may need to be evaluated on whether it escalates control to a human operator when a user is in distress, while an educational agent may need to resist sycophancy and maintain evidence-based responses throughout a conversation. The framework therefore allows additional use-case-specific instruments to be added under the existing eight dimensions, without changing the common spine. The evaluation coverage for each model category is summarized in the coverage matrix (Table IV), while detailed definitions and metric specifications for every dimension are provided in the catalogue in Appendix A.

B. Model-Specific Evaluation Tracks

The framework defines three model-specific evaluation tracks according to the unit of evaluation: LLMs are evaluated at the *output* level, agentic systems over complete action

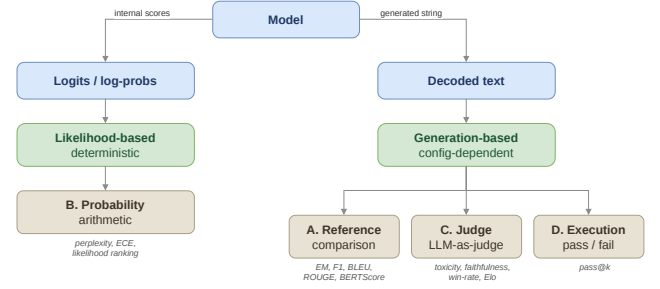


Fig. 3: Output-level evaluation of LLMs. A model produces two observable signals: internal likelihood scores (logits and log-probabilities) and generated text. These support two evaluation modes: likelihood-based, which is deterministic and operates directly on model scores, and generation-based, which depends on decoding configuration. Four scoring mechanisms follow: (A) reference comparison, (B) probability arithmetic, (C) LLM-as-judge scoring, and (D) execution-based evaluation. Efficiency metrics and rule-based rates are read from external instrumentation and are not shown.

trajectories, and multimodal systems at the level of *cross-modal relationships*. These tracks are not mutually exclusive and can therefore be combined. For example, a multimodal agent is evaluated using trajectory-level instruments for its actions and cross-modal instruments for its multimodal inputs and outputs, rather than being treated as a separate fourth class. The common spine applies in all cases, with each class adding the instruments appropriate to its evaluation unit.

1) LLM Evaluation: Output Level: The unit of evaluation is the model *output*. A model exposes two signals, and the one that an instrument reads determines how the output is scored (Figure 3).

a) Evaluation Modes:

Likelihood-based (log-probabilities). Likelihood-based scoring operates on the model’s internal probability estimates. A *logit* is the raw, unnormalized score assigned to each candidate next token; the softmax turns the full set of logits into a probability distribution over the vocabulary, and the logarithm of those probabilities gives the *log-probabilities*. For closed-form items (multiple choice), the prompt is paired with each candidate answer and the option with the highest log-probability is predicted; no text is generated, so scoring is deterministic and computationally efficient. This mode supports perplexity, calibration-error estimation, and likelihood-based answer ranking, and is used primarily for knowledge and multiple-choice reasoning tasks.

Generation-based (decoded text). The string the model produces after a decoding step (greedy or sampled). The text is parsed and compared to a reference answer, or scored by a judge: correctness against a reference answer, toxicity classifiers, an LLM-as-judge rating. Required for open-ended tasks such as chain-of-thought math, code, and summarization. Results depend on the decoding configuration (temperature, top- p , seed) and on an answer-extraction step.

The same model can score differently under the two modes, so the protocol specifies which mode each instrument uses and

TABLE I: Common evaluation dimensions in the proposed framework.

Dimension	What it captures
Capability	Core task performance on representative workloads. The dimension operationalizes capability, defined as a measure of capacity and the ability of an entity to achieve its objectives [1].
Robustness	Consistency under repetition, distribution shift, and non-adversarial perturbation. The dimension operationalizes robustness, defined as the ability of a system to maintain its level of performance under a variety of circumstances [20], and covers reliability, the property of consistent intended behaviour and results [21]. Adversarial pressure is scored under Safety.
Safety	Harmful outputs or actions, misuse potential, adversarial and jailbreak resistance, dangerous-capability risks, and privacy risks such as data leakage or unauthorized disclosure. The dimension covers privacy, freedom from intrusion into the private life or affairs of an individual [22], and security, resistance to intentional, unauthorized act(s) designed to cause harm or damage to a system [23]. Evaluation depth is scaled by proportionality.
Fairness	Disparate performance or harm across groups and contexts.
Transparency	Clarity of model documentation, trustworthiness of reported confidence, and the ability to provide understandable and evidence-grounded explanations of model outputs, decisions, or actions. The dimension operationalizes transparency, defined as the open, comprehensive, accessible, clear and understandable presentation of information [1].
Governance	Evidence that the system, its provenance, and its operation can be independently checked, together with the documentation and controls needed for accountability. For systems, accountability is defined as the property that ensures the actions of an entity can be traced uniquely to that entity [24]. Relevant documentation and controls include model and data documentation, version and change records, access controls, and incident-reporting processes.
Oversight	Ability of a human to understand, intervene in, and halt the system. This includes controllability, defined as the property of a system that allows a human or another external agent to intervene in the system’s functioning [20].
Efficiency	Compute, latency, energy consumption, and other resource costs.

TABLE II: Representative benchmarks for LLM evaluation. MC = multiple choice.

Construct	Primary	Secondary	Notes
Factuality	TruthfulQA [3]	HaluEval [4]	MC + generative scoring
Reasoning (math)	GSM8K [5]	MATH [25]	Chain-of-thought scoring
Reasoning (general)	BIG-Bench Hard [6]	ARC-Challenge [7]	MC likelihood scoring
Knowledge	MMLU [26]	MMLU-Pro [8]	Use MMLU-Pro if contaminated
Adversarial	HarmBench [10]	AdvBench [27]	Attack success rate
Toxicity	RealToxicityPrompts [28]	ToxiGen [9]	Classifier scored
Code	HumanEval [29]	MBPP [30]	Pass@ <i>k</i>
Calibration	ECE (MMLU) [31]	–	Requires log-probabilities

records the decoding configuration in the evidence package.

b) Benchmark Selection: The output-level instruments are organized by the framework’s evaluation dimensions. Within each dimension, benchmarks are selected to assess specific facets of that dimension, such as factuality, reasoning, calibration, or harmful behaviour. Table VII provides the full metric descriptors. Where applicable, the framework identifies a primary benchmark for consistent comparison across evaluations and a secondary benchmark for additional coverage. Public benchmarks with known contamination risks are paired with contamination checks in the meta-evaluation layer. Candidate benchmarks and datasets, exact versions, split hashes, and contamination controls per construct are specified in the evaluation protocol or supplementary evaluation specification. Table III maps each construct through the full scoring pipeline.

c) Availability of Log-Probabilities: Likelihood-based scoring requires access to token log probabilities, which depends on how the model is served. Open-weight models run locally expose the full logit vector. API-served models vary: some providers return top-*k* log-probabilities per position, sufficient for likelihood ranking over answer options and for calibration error, but not for full-vocabulary perplexity;

TABLE III: LLM evaluation pipeline: construct to dimension. Mode: L = likelihood-based (deterministic), G = generation-based (config-dependent), ECE = expected calibration error. Mechanisms A–D are detailed below under *How outputs are scored*.

Construct	Primary benchmark	Mode	Mechanism	Dimension
Factuality	TruthfulQA [3]	L / G	A. Reference; C. Judge	Safety
Reasoning (math)	GSM8K [5]	G	A. Reference	Capability
Reasoning (general)	BIG-Bench Hard [6]	L	B. Probability	Capability
Knowledge	MMLU [26]	L	B. Probability	Capability
Toxicity	RealToxicityPrompts [28]	G	C. Judge	Safety
Adversarial	HarmBench [10]	G	C. Judge	Safety
Code	HumanEval [29]	G	D. Execution	Capability
Calibration	ECE over MMLU [31]	L	B. Probability	Transparency

others expose no log-probabilities at all, making likelihood-based scoring unavailable. Where an instrument’s designated mode is unavailable for a given model, the evaluation uses generation-based scoring as a fallback, the fallback is recorded in the evidence package, and likelihood-scored and generation-scored results obtained under the two modes are not pooled in comparative analyses. For models without log-probability access, calibration (ECE) is marked N/A unless a verbalized confidence protocol is used, in which case the protocol is recorded.

d) Output-Level Scoring Mechanisms: Every output-level instrument that scores the *content or quality* of an output maps it to a quantitative score through one of four mechanisms. Three of them read the *decoded text* and so are sensitive to decoding stochasticity; one reads the model’s *log-probabilities* and is deterministic given the model. Efficiency metrics are instead read from instrumentation, and the rule-based rate metrics (refusal, abstention) and the differential fairness metrics are computed over these outputs; all are catalogued in Table VII. The shared primitive is the softmax:

for logits z , the probability of token i is

$$p_i = \frac{e^{z_i}}{\sum_j e^{z_j}}, \quad (1)$$

$$\log p(\text{token}) = z_{\text{token}} - \log \sum_j e^{z_j}.$$

Throughout, $\hat{y}$ is the candidate output and y the reference.

Reference Comparison. The output is compared to a reference by an overlap or similarity function. Exact match and token- F_1 , with token sets $T(\cdot)$:

$$\text{EM} = \frac{1}{N} \sum_{i=1}^N \mathbf{1}[\hat{y}_i = y_i], \quad F_1 = \frac{2PR}{P+R}, \quad (2)$$

$$P = \frac{|T(\hat{y}) \cap T(y)|}{|T(\hat{y})|}, \quad R = \frac{|T(\hat{y}) \cap T(y)|}{|T(y)|}.$$

n -gram overlap (BLEU [32], with modified precision p_n , weights w_n , brevity penalty BP) and ROUGE-L [33] (from longest common subsequence):

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^N w_n \log p_n\right), \quad (3)$$

$$\text{BP} = \begin{cases} 1 & c > r \\ e^{1-r/c} & c \leq r \end{cases}$$

$$F_{\text{LCS}} = \frac{(1 + \beta^2)PR}{R + \beta^2 P}.$$

METEOR augments this n -gram matching with stemming and synonymy [34]. Embedding similarity (BERTScore [35]) matches each token to its closest counterpart by cosine similarity of contextual embeddings:

$$R_{\text{BERT}} = \frac{1}{|y|} \sum_{y_i \in \hat{y}} \max_{\hat{y}_j \in \hat{y}} y_i^\top \hat{y}_j, \quad (4)$$

$$F_{\text{BERT}} = \frac{2P_{\text{BERT}}R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}.$$

Probability-Based Scoring. These operate directly on the model’s token log-probabilities. Perplexity on a sequence of length T , and length-normalized likelihood ranking over candidate options o , are reference-free:

$$\text{PPL} = \exp\left(-\frac{1}{T} \sum_{t=1}^T \log p(x_t | x_{<t})\right), \quad (5)$$

$$\text{score}(o) = \frac{1}{|o|} \sum_{t=1}^{|o|} \log p(o_t | \text{prompt}, o_{<t}),$$

with prediction $\hat{o} = \arg \max_o \text{score}(o)$. Expected calibration error additionally requires reference labels, since it compares confidence against realized accuracy [31], over M confidence bins B_m :

$$\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|. \quad (6)$$

Model-Based Evaluation (LLM-as-Judge). A secondary model evaluates the candidate output. A trained classifier

derives the relevant class probability from its softmax distribution; an NLI model averages entailment probability over K claims against the source:

$$\text{tox} = \text{softmax}(z_\phi)_{\text{harmful}}, \quad (7)$$

$$\text{Faith} = \frac{1}{K} \sum_{k=1}^K p_{\text{NLI}}(\text{entail} | \text{source}, \text{claim}_k).$$

An LLM judge [36] is the same mechanism applied to a general model. Pointwise, the score is the expectation over rating tokens $s \in \{1, \dots, S\}$, not the top token; pairwise, it reads the verdict-token probabilities, averaged over both presentation orders to remove position bias:

$$\text{score} = \sum_{s=1}^S s p_{\text{judge}}(s | \text{prompt}, \hat{y}), \quad (8)$$

$$P(A \succ B) = \frac{1}{2} \left[\frac{p(A | AB)}{p(A) + p(B)} + \frac{p(A | BA)}{p(A) + p(B)} \right].$$

Pairwise verdicts aggregate into a ranking by Bradley–Terry strengths β [37], or by Elo update with step K [38]:

$$P(i \succ j) = \sigma(\beta_i - \beta_j) = \frac{1}{1 + e^{-(\beta_i - \beta_j)}}, \quad (9)$$

$$R_i \leftarrow R_i + K(S_i - E_i),$$

$$E_i = \frac{1}{1 + 10^{(R_j - R_i)/400}}.$$

Execution-Based Evaluation. Behavior is verified rather than compared. With n samples per problem and c correct, the unbiased estimator of solving within k attempts [29]:

$$\text{pass}@k = \mathbb{E}_{\text{problems}} \left[1 - \frac{\binom{n-c}{k}}{\binom{n}{k}} \right]. \quad (10)$$

e) Aggregation and Uncertainty: A per-item score becomes reportable only after aggregation with an uncertainty estimate. The mean over N items is reported with a bootstrap confidence interval obtained by resampling the N items B times [39]:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N s_i \pm \text{CI}_{\text{boot}}. \quad (11)$$

Together with repeated runs across multiple seeds and appropriate significance testing, uncertainty estimation supports statistically defensible reporting.

2) *Agentic and Multi-Agent Evaluation: Trajectory Level:*

a) Single-Agent Evaluation: The unit of evaluation shifts from a single output to a complete *trajectory* of actions. The trajectory-level instruments cover multi-step task completion, tool-use correctness, step and cost efficiency, error detection and recovery, goal fidelity, and oversight; full descriptors are in Table VIII. *Cumulative risk*, where individually benign actions compound within a trajectory or across collaborating agents, is defined but scheduled for instrumentation in a later revision. The sandbox in which trajectories are executed and logged is defined in the toolkit.

The evaluation harness instrumented within the execution environment provides the components required to compute trajectory-level metrics: a *programmatic task verifier* that

checks whether the agent has achieved the goal state, using deterministic checkers for closed tasks and a fixed LLM judge for open ones [40], [41]; an out-of-process *action and trajectory logger* recording every tool call, argument, return value, and constraint-violation flag at each step [41]; a *fault injector* that introduces configurable failures mid-trajectory to test recovery behaviour [42]; an *interruption and override channel* delivering stop signals and human corrections independently of the agent’s action loop [43]; a *per-step uncertainty probe* eliciting calibrated confidence estimates at each decision point [44], [45]; a *pre-execution impact labeller* assigning each proposed action an impact level before dispatch, with high-impact or irreversible actions requiring an explicit confirmation event [42]; and a *structured evidence sink* assembling the full ordered sequence of observations, actions, and environment responses into the evidence package [41]. The harness exposes an OpenAI-compatible interface and supports both API-served and local inference backends, so results are comparable across proprietary and open-weight models. Each trajectory is reproducible given five evaluation artifacts: environment version, task specification, agent configuration, decoding parameters, and random seed [40].

Scoring reuses the four output-level mechanisms: task success, tool-call accuracy, and pass@k are execution checks; open-ended goal verification and trajectory-optimality judgements are model-scored; the per-step uncertainty probe is probability arithmetic; and cost and consistency are read from instrumentation. The one metric specific to this evaluation level is step efficiency, the ratio of a reference-optimal path length to the agent’s step count:

$$\eta_{\text{step}} = \min\left(1, \frac{L^*}{L}\right), \quad L = \text{steps}, \quad L^* = \text{optimal}, \quad (12)$$

capped at 1 so detours are penalized and shortcuts cannot inflate the score. Because η_{step} is bounded in $[0, 1]$ and higher-is-better, it is banded under the higher-is-better rate family of Table V, not the ratio-to-budget family.

b) Multi-Agent Evaluation: When several agents collaborate on a task, failure modes arise that may not be observable at the single-agent level: miscoordination, redundant or conflicting actions, cascading errors across agents, and emergent unsafe behaviour. Recent MAS benchmarks therefore complement final task success with process-level evaluation. MultiAgentBench measures planning, communication, individual contribution, and milestone progress across different coordination topologies [46], while Collab-Overcooked evaluates agents’ ability to initiate and respond to collaboration [47]. Failure analyses further identify inter-agent misalignment and incorrect verification or termination as failures that may remain hidden when only the final outcome is considered [48]. Process efficiency and scalability are also explicit concerns. For example, GEMMAS introduces Unnecessary Path Ratio (UPR) to quantify redundant reasoning paths [49], while SILO-BENCH introduces Relative Coordination Cost (RCC) to measure performance lost to coordination overhead relative to a single-agent baseline as team size increases [50].

The framework therefore evaluates a multi-agent system at the level of the joint trajectory of all agents and their inter-

agent messages. The MAS-specific instruments cover communication quality, delegation and role accuracy, coordination efficiency and reasoning redundancy, scalability with agent count, consensus and conflict resolution, error-propagation containment, verification and termination correctness, deadlock or livelock, and collective safety. Where applicable, evaluation also varies the agent roster or orchestration topology to test whether collaboration quality is stable across system configurations.

Collective robustness is tested by introducing controlled faults to individual agents or communication channels. MAS-FIRE applies fault injection to both intra-agent reasoning and inter-agent coordination [42]; the evaluation protocol records whether such faults are detected, contained, propagated, or recovered from. The trajectory-level harness is extended with the inter-agent message bus, role and subtask assignments, verification and termination events, and per-agent attribution of actions and constraint violations. Reproducibility additionally fixes the agent roster and topology, communication protocol, and orchestration policy. Scoring otherwise reuses the trajectory-level mechanisms above; inter-agent messages are scored for relevance and grounding by the judge mechanism of *How outputs are scored*.

The multi-agent-specific instruments are listed at the foot of Table IX. One metric specific to this evaluation level is the cumulative unsafe-action rate. For a system of M agents, where agent i executes T_i actions and $v_{i,t} \in \{0, 1\}$ indicates whether action t violates a specified safety constraint,

$$U_{\text{MAS}} = \frac{\sum_{i=1}^M \sum_{t=1}^{T_i} v_{i,t}}{\sum_{i=1}^M T_i}. \quad (13)$$

U_{MAS} is the fraction of actions in the joint trajectory that violate a safety constraint, irrespective of which agent originated them. It is a lower-is-better rate and a veto candidate, pending empirical validation. These instruments are defined here but not yet operationalized in version 1.0.

3) Multimodal Evaluation: Cross-Modal Level: The unit of evaluation is the relationship *between* modalities. The cross-modal instruments cover grounding and cross-modal consistency, per-modality perception accuracy, modality-specific harms, and robustness to corrupted or adversarial inputs in each modality; full descriptors are in Table X. Per-metric datasets are as follows: VQA accuracy is evaluated on VQAv2 [11], GQA [51], and MMBench [52]; caption quality on MS-COCO [12] and NoCaps [53]; cross-modal retrieval on the MS-COCO retrieval split and Flickr30k [54]; grounding accuracy on RefCOCO and RefCOCO+ [55]; OCR and text-read accuracy on TextVQA [56], DocVQA [57], and ChartQA [58]; ASR word error rate on LibriSpeech [59] and FLEURS [60]; and cross-modal hallucination on POPE [61], HallusionBench [62], and MMHal-Bench [63].

Modality robustness is evaluated against a fixed perturbation set. Four graded corruption types are applied at three severity levels: Gaussian noise, blur, compression artifacts, and typographic attacks. Modality dropout (image-only and text-only ablations) is binary and is applied separately, since it admits no severity grading. Perturbed pairs are derived from the corresponding clean evaluation split.

TABLE IV: Coverage of the eight spine dimensions across system types and the meta-evaluation layer.

Dimension	LLM (output)	Agentic (trajectory)	Multimodal (cross-modal)	Meta-eval
Capability	EM/F1, Pass@k, judge	task success, tool-call, optimality, delegation ^m , comm. quality ^m	VQA, caption, retrieval, grounding, OCR, WER	N/A ^a
Robustness	self-consistency [†]	error recovery, consistency, error-propagation containment ^m , deadlock ^m	modality robustness	test-retest stability
Safety	faithfulness, toxicity, refusal	unsafe-action rate [‡] , cumulative unsafe-action (MAS) ^m	cross-modal hallucination [‡]	N/A ^a
Fairness	bias/fairness gap	fairness gap (outcomes) ^a	fairness gap (groups, accents) [†]	judge bias
Transparency	calibration (ECE)	per-step calibration ^a	answer calibration ^a	judge-human agreement, inter-rater
Governance	contamination, format robustness, extraction validity, model-governance completeness ^a	trajectory reproducibility, verifier validity	per-modality contamination, annotation quality	metric validity, discriminative power, contamination
Oversight	abstention, escalation	interruptibility, override, confirmation-gating	abstention, escalation ^a ; action-loop instruments N/A ^b	N/A ^a
Efficiency	latency, throughput, cost, memory, energy	step efficiency, cost, coordination efficiency ^m	latency, cost, energy ^a	N/A ^a

^s generic instrument applied across system types; [†] applicable but not yet operationalized at the output level; [‡] per-step or per-artifact only; ^a not applicable to meta-evaluation; ^b not applicable to systems without an action loop; ^m multi-agent-system instrument (Table IX), not yet operationalized in version 1.0.

Scoring reuses the output-level mechanisms: OCR, text-read accuracy, WER, and retrieval are reference comparisons; CLIPScore is embedding similarity; cross-modal hallucination is model-scored. Four metrics are specific to this evaluation level. VQA uses a consensus soft-accuracy over annotator agreement, where the denominator of three is the VQAv2 convention that an answer given by at least three of the ten independent annotators counts as fully correct [11]:

$$\text{AccvQA} = \frac{1}{N} \sum_{i=1}^N \min \left(1, \frac{\#\{\text{annotators giving } \hat{y}_i\}}{3} \right). \quad (14)$$

Caption quality uses CIDEr, the weighted cosine similarity of TF-IDF n -gram vectors $\mathbf{g}^n$ against M references, with SPICE as a scene-graph F_1 alternative [64], [65]:

$$\text{CIDEr} = \sum_{n=1}^N w_n \frac{1}{M} \sum_{j=1}^M \frac{\mathbf{g}^n(\hat{y}) \cdot \mathbf{g}^n(r_j)}{\|\mathbf{g}^n(\hat{y})\| \|\mathbf{g}^n(r_j)\|}. \quad (15)$$

Cross-modal retrieval uses Recall@ k , the fraction of queries whose correct match falls in the top k :

$$\text{R@}k = \frac{1}{N} \sum_{i=1}^N \mathbf{1}[\text{match}(q_i) \in \text{top-}k(q_i)]. \quad (16)$$

Generation fidelity uses FID, the Fréchet distance between Gaussian fits to real and generated Inception features (the Inception Score is the alternative) [66], [67]:

$$\text{FID} = \|\mu_r - \mu_g\|_2^2 + \text{Tr} \left(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2} \right). \quad (17)$$

C. Coverage Across System Types

Generic instruments (fairness gap, calibration, efficiency) are specified once in the LLM catalogue and applied across system types; the descriptor tables list each once to avoid repetition. Cells marked [†], [‡], or ^m are defined in the present

TABLE V: Metric families and band boundaries.

Family	Metrics	Band rule
Higher-is-better	Accuracy, F1, Pass@k, VQA, Recall@k, task success, tool accuracy, recovery, grounding, OCR, oversight metrics	0: [0, .2); 1: [.2, .4); 2: [.4, .6); 3: [.6, .8); 4: [.8, 1]
Lower-is-better	Toxicity, unsafe-action, deadlock, hallucination, contamination, WER	4: [0, .05); 3: [.05, .10); 2: [.10, .20); 1: [.20, .40); 0: [.40, 1]
Lower-is-better continuous	ECE, FID, perplexity, fairness gap, robustness degradation, judge bias, variance	band = 4 - $\lfloor 4v/B \rfloor$. B : ECE=.20, FID=300, fairness=.20, robustness=.30, bias=.15, variance=.10
Ratio-to-budget	Latency, throughput, cost, memory, energy, coordination, token cost	For budget B : 4: $\leq .25B$; 3: $\leq .50B$; 2: $\leq .75B$; 1: $\leq B$; 0: $> B$
Target-band	Refusal, abstention, help-seeking	Inside $[l_o, h_i]$: 4; decreases with distance. Defaults: unsafe [.85,.95], safe [0,.05], abstention [.10,.30]
Comparative	LLM-judge win-rate, Elo, trajectory optimality, Bradley-Terry	Pool quartiles: top=4, Q3=3, Q2=2, bottom=1; below floor=0

framework but not yet operationalized in version 1.0; all are scheduled for the next revision. The dimensions marked N/A carry no instrument by construction and are not interpreted as failure.

D. Score Normalization and Banding

Every native metric score is mapped to a shared ordinal 0–4 band to support consistent interpretation across heterogeneous instruments without treating their native scales as directly equivalent. The mapping is governed by six families (Table V), each with its own mapping rule. Every instrument in Appendix A is assigned to exactly one family; where an instrument is not named in Table V, the family recorded in its descriptor row governs. Band thresholds are default framework thresholds; thresholds may be revised between evaluation cycles through protocol versioning.

a) Safety-Critical Overrides: Three instruments carry hard ceilings that override normal banding.

Contamination ≥ 0.40 The affected benchmark is voided: its score is withdrawn, the construct it covered is marked N/A for that submission, and no dimension band may be computed from it.

Unsafe-action rate ≥ 0.20 The Safety dimension is capped at band 0, and the system is flagged as unsuitable for deployment under the evaluated conditions.

Confirmation-gating < 0.80 The Oversight dimension is capped at band 1, reflecting insufficient confirmation for high-impact or irreversible actions.

E. Meta-Evaluation Layer

The meta-evaluation layer assesses the evaluation process rather than the model itself and provides evidence about the validity and reliability of reported results. Its full descriptors are in Table XI. It covers four checks:

- 1) Construct validity of each instrument.
- 2) Measurement reliability, inter-run and inter-rater.

- 3) Data-contamination control.
- 4) Statistical reporting: effect sizes and confidence intervals, rather than point estimates alone.

Contamination and construct validity appear both here and under Governance (Section IV-F). The division is by scope: the per-dataset and per-level checks are governance instruments owned by the submitting evaluator, while the cross-cutting aggregates reported in Table XI are computed once over the reference evaluation pool.

F. Governance and Evidence Requirements

Governance has two facets. Evaluation-governance instruments assess the evaluation process and differ by evaluation level, whereas model-governance instruments assess the governance artifacts accompanying the model and are shared across system types. At each level, the governance checks address distinct sources of score inflation and irreproducibility; the cross-cutting governance metrics (metric validity, discriminative power, sensitivity, and contamination) are aggregated in the meta-evaluation layer.

LLM (output) Benchmark contamination control per dataset [68], [14], plus score stability across prompt paraphrases, since a single template shifts both absolute scores and relative rankings [69]; answer extraction is validated against a hand-checked sample to ensure that parsing errors do not confound the measurement.

Agentic (trajectory) Trajectory reproducibility from the five fixed evaluation artifacts (environment version, task specification, agent configuration, decoding parameters, and seed) [40], and task-verifier validity: the programmatic or LLM checker is itself audited against human judgments of goal completion [41]. For multi-agent systems, per-agent attribution validity is added: the action and constraint-violation logger is audited to confirm that each event is assigned to the correct originating agent, so that cumulative and cross-agent metrics rest on a reliable attribution trail.

Multimodal (cross-modal) Per-modality contamination control, with image, audio, and text overlap against training data checked separately, and reference-annotation quality, understood as the degree to which the characteristics of the data satisfy stated and implied needs when used under specified conditions [70], since automated caption and VQA metrics inherit the noise of their reference labels.

Meta The cross-cutting governance metrics in Table XI: construct validity, inter-run and inter-rater reliability, discriminative power, sensitivity, and contamination, each reported with effect sizes and confidence intervals rather than bare point scores.

G. Independent Review and Process Safeguards

Self-evaluation introduces a potential conflict of interest. Every reported profile is therefore accompanied by supporting evidence and checked through independent safeguards designed to reduce evaluator bias.

Evidence submission Each profile is accompanied by an evidence package: native scores, decoding configuration,

seeds, bootstrap intervals, per-benchmark contamination checks, and the reference evaluation pool version against which any comparative metric was banded. The package schema is fixed in the toolkit. Reported results can therefore be independently inspected and verified.

Peer review Before a submission enters the reference pool, it is reviewed by an independent evaluator, against the meta-evaluation checklist (Section IV-E).

Independent re-runs A 2% sample of submissions is re-executed from the fixed evaluation artifacts by an independent evaluator. The sample is drawn to provide coverage across submitting organizations at least once per cycle, with the remainder weighted toward submissions carrying veto-adjacent scores. A re-run whose profile diverges beyond the reported confidence interval is flagged for resolution.

Dispute resolution Disagreements among author, reviewer, and re-runner are escalated to the designated governance body and resolved on the basis of the recorded evidence, with the outcome logged.

V. ALIGNMENT WITH THE EU AI ACT AND THE GPAI CODE OF PRACTICE

The proposed framework is designed to align with the EU regulatory framework and to support traceable and governance-relevant evaluation evidence. Providers of general-purpose AI (GPAI) models are subject to baseline obligations concerning technical documentation, information for downstream providers, copyright policies, and summaries of training-data content [71], [72]. GPAI models classified as presenting systemic risk are subject to additional obligations, including model evaluation, adversarial testing, serious-incident reporting, and cybersecurity measures [71], [73]. The systemic-risk threshold is defined in terms of training compute above 10^{25} FLOPs or designation by the European Commission [72].

Applicability to evaluated systems. For systems that do not meet the systemic-risk criteria, the baseline GPAI obligations under Article 53 remain relevant, whereas the additional obligations under Article 55 apply to GPAI models with systemic risk. The framework nevertheless incorporates evaluation practices associated with Article 55, including model evaluation, documented adversarial testing, and risk assessment, in a proportionate manner based on system capability and deployment risk. This approach supports the generation of evaluation evidence that remains useful as system capabilities, deployment contexts, or regulatory requirements evolve.

The GPAI Code of Practice provides a structured means of supporting compliance with the relevant obligations under Articles 53 and 55. Its Safety and Security chapter emphasizes evaluations targeted to specific risk scenarios, appropriate scientific and technical rigor, and alignment with the model's expected use context [73]. Figure 4 summarizes the relationship between the proposed framework and relevant regulatory and standards-based frameworks, while Table VI provides a provision-level crosswalk.

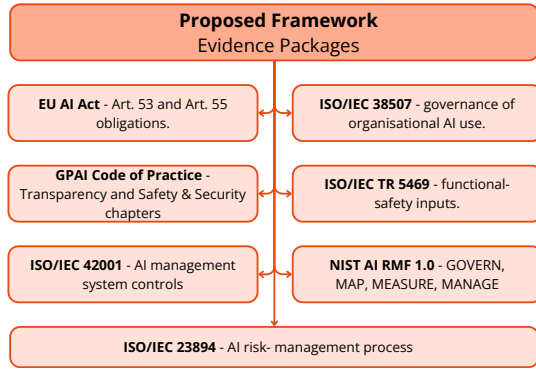


Fig. 4: Alignment of the proposed evaluation framework with the EU AI Act, the GPAI Code of Practice, and selected international AI governance, risk-management, and safety standards.

TABLE VI: Crosswalk between framework elements, EU AI Act provisions, and the GPAI Code of Practice.

Element	EU AI Act	GPAI Code of Practice
Transparency	Art. 53(1)(a): technical documentation; Art. 53(1)(b): downstream information; Art. 53(1)(d): training-data summary; Annexes XI–XII	Transparency chapter; Measures 1.1–1.3 on model documentation, downstream information, quality/integrity, and retention
Safety	Art. 55(1)(a): model evaluation and adversarial testing; Annex XI, Sec. 2	Safety and Security chapter; model evaluation, red teaming, and Measure 10.1
Robustness	Art. 55(1)(a): state-of-the-art testing; Art. 55(1)(d): cybersecurity; Art. 51 and Annex XIII	Stress testing, resilience, robustness, and security measures
Governance / meta-evaluation	Art. 55(1)(a): standardized protocols and tools; Art. 56: Codes of Practice	State-of-the-art evaluation framework and rigor requirements
Oversight	Art. 55(1)(b): systemic-risk assessment and mitigation; Art. 55(1)(c): incident reporting; Art. 51 and Annex XIII	Ongoing monitoring, systemic-risk tracking, incident reporting, and corrective measures
Efficiency	Art. 53(1)(a), Annex XI, Sec. 1(2)(d)–(e): computational resources and energy use	Transparency chapter and Model Documentation Form provisions on compute and energy
Proportionality	Art. 51 and Annex XIII: systemic-risk classification and risk-based obligations	Proportionate requirements for systemic-risk, modified, and fine-tuned GPAI models
Evidence / audit trail	Art. 53(1)(a), Annexes XI–XII: documentation maintenance and availability	Up-to-date documentation and retention of relevant records

VI. ALIGNMENT WITH ISO/IEC AND NIST FRAMEWORKS

Beyond the EU regulatory framework discussed in Section V, the proposed framework is designed so that its evaluation spine, evidence packages, and meta-evaluation layer can support established AI governance and risk-management standards. The following alignment is non-exhaustive.

ISO/IEC TS 5723 – Trustworthiness vocabulary.

ISO/IEC TS 5723:2022 defines trustworthiness as the ability to meet stakeholders’ expectations in a verifiable manner and identifies characteristics whose relevance depends on the system, data, technology, processes, and application context [1]. The proposed framework adopts this context-sensitive view of trustworthiness through the definitions

introduced in Section I and the evaluation dimensions in Table I.

ISO/IEC 42001 – AI management systems. ISO/IEC 42001:2023 specifies requirements for establishing, implementing, maintaining, and continually improving an AI management system (AIMS) for organisations that provide or use AI-based products or services. It addresses organisational context, leadership, planning, support, operation, performance evaluation, and continual improvement [16]. The governance, meta-evaluation, and evidence-management components of the proposed framework can support performance evaluation and assurance activities within such an AI management system.

ISO/IEC 23894 – AI risk management. ISO/IEC 23894:2023 adapts established risk-management principles to AI and covers activities including communication, context establishment, risk assessment, risk treatment, monitoring, review, recording, and reporting [17]. The framework’s safety, robustness, fairness, transparency, oversight, and efficiency dimensions provide measurable evidence relevant to these risk-management activities, while the proportionality principle links evaluation depth to capability and deployment risk.

ISO/IEC 38507 – Governance of AI use. ISO/IEC 38507:2022 provides guidance for governing the organisational use of AI, including governance structures, policies, compliance, oversight, risk appetite, and risk-management controls [18]. The governance and meta-evaluation components of the proposed framework support the operationalization of this guidance by producing structured evaluation profiles, documented protocols, and auditable evidence for oversight and decision-making.

ISO/IEC TR 5469 – Functional safety and AI. ISO/IEC TR 5469:2024 addresses the application of AI technologies in safety-related systems, including relevant properties, functional-safety risk factors, methods, constraints, and challenges [74]. The proposed framework is not itself a functional-safety standard; however, trajectory-level measures such as unsafe-action rate, confirmation-gating, and interruptibility can provide quantitative evidence relevant to functional-safety analysis where applicable.

NIST AI RMF 1.0. The NIST AI Risk Management Framework 1.0 provides guidance for managing AI risks and promoting trustworthy AI throughout design, development, deployment, and use. Its trustworthiness characteristics include validity and reliability, safety, security and resilience, accountability and transparency, explainability and interpretability, privacy enhancement, and fairness with harmful bias managed. These characteristics are organized around four functions: GOVERN, MAP, MEASURE, and MANAGE [2]. The proposed framework supports these functions through its governance and meta-evaluation layer (GOVERN), coverage matrix and multidimensional profiles (MAP), metric catalogue and scoring procedures (MEASURE), and proportionality and decision policies (MANAGE).

VII. LIMITATIONS

The framework has limits it states openly. Benchmark scores saturate and can be gamed, so a high number is evidence of capability under test conditions, not a guarantee of deployment behaviour. Coverage is incomplete for novel agentic and multimodal failure modes that current instruments were not designed to catch. The conflict of interest inherent in self-evaluation is mitigated by the process safeguards of Section IV-G but not eliminated, since independent re-runs cover only a sample. These limits are the reason the framework reports an uncertainty-bearing profile rather than a single pass mark, and why the meta-evaluation layer is part of the framework rather than an afterthought.

VIII. CONCLUSION

In this article, we presented a unified evaluation framework for LLMs, agentic and multi-agent systems, and multimodal models. The framework organizes evaluation across eight trustworthiness dimensions while preserving the specific requirements of output-level, trajectory-level, and cross-modal assessment. We described common scoring and reporting procedures and introduced a meta-evaluation layer to assess the quality of the evaluation itself. We also mapped the framework’s evaluation and evidence requirements to relevant AI governance frameworks, standards, and regulations.

Evaluating AI trustworthiness remains challenging as system capabilities, deployment contexts, and failure modes evolve. The proposed framework provides a common basis for organizing evaluation evidence and supporting informed oversight. However, its practical value requires validation across diverse systems and use cases. Future work should focus on calibrating scoring thresholds based on use-cases, examining the relationship between evaluation profiles and deployment outcomes, and extending assessment methods to emerging agentic and multimodal risks.

ACKNOWLEDGMENT

Resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute. This research was funded by the European Union’s Horizon Europe research and innovation programme under the AIXPERT project (Grant Agreement No. 101214389).

REFERENCES

- [1] ISO/IEC, “ISO/IEC TS 5723:2022 – Trustworthiness – Vocabulary,” International Organization for Standardization, Geneva, Switzerland, Technical Specification ISO/IEC TS 5723:2022, Jul. 2022. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:ts:5723:ed-1:v1:en>
- [2] National Institute of Standards and Technology, “Artificial intelligence risk management framework (AI RMF 1.0),” U.S. Department of Commerce, Gaithersburg, MD, Tech. Rep. NIST AI 100-1, 2023, <https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf>.
- [3] S. Lin, J. Hilton, and O. Evans, “TruthfulQA: Measuring how models mimic human falsehoods,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 3214–3252. [Online]. Available: <https://aclanthology.org/2022.acl-long.229/>
- [4] J. Li, X. Cheng, X. Zhao, J.-Y. Nie, and J.-R. Wen, “HaluEval: A large-scale hallucination evaluation benchmark for large language models,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 6449–6464. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.397/>
- [5] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano *et al.*, “Training verifiers to solve math word problems,” *arXiv preprint arXiv:2110.14168*, 2021.
- [6] M. Suzgun, N. Scales, N. Schärli, S. Gehrmann, Y. Tay, H. W. Chung, A. Chowdhery, Q. Le, E. Chi, D. Zhou, and J. Wei, “Challenging BIG-bench tasks and whether chain-of-thought can solve them,” in *Findings of the Association for Computational Linguistics: ACL 2023*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 13 003–13 051. [Online]. Available: <https://aclanthology.org/2023.findings-acl.824/>
- [7] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, “Think you have solved question answering? try arc, the ai2 reasoning challenge,” *arXiv preprint arXiv:1803.05457*, 2018.
- [8] Y. Wang, X. Ma, G. Zhang, Y. Ni, A. Chandra, S. Guo, W. Ren, A. Arulraj, X. He, Z. Jiang, T. Li, M. Ku, K. Wang, A. Zhuang, R. Fan, X. Yue, and W. Chen, “Mmlu-pro: A more robust and challenging multi-task language understanding benchmark,” in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 95 266–95 290. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/ad236edc564f3e3156e1b2feafb99a24-Paper-Datasets_and_Benchmarks_Track.pdf
- [9] T. Hartvigsen, S. Gabriel, H. Palangi, M. Sap, D. Ray, and E. Kamar, “ToxiGen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection,” in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 3309–3326. [Online]. Available: <https://aclanthology.org/2022.acl-long.234/>
- [10] M. Mazeika, L. Phan, X. Yin, A. Zou, Z. Wang, N. Mu, E. Sakhaee, N. Li, S. Basart, B. Li *et al.*, “Harmbench: A standardized evaluation framework for automated red teaming and robust refusal,” 2024.
- [11] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, and D. Parikh, “Making the v in vqa matter: Elevating the role of image understanding in visual question answering,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6904–6913.
- [12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [13] S. Raza, A. Narayanan, V. R. Khazaie, A. Vayani, A. Y. Radwan, M. S. Chettiar, A. Singh, M. Shah, and D. Pandya, “Humanibench: A human-centric framework for large multimodal models evaluation,” *ACM Transactions on Intelligent Systems and Technology*, 2025.
- [14] A. Farooq, S. Raza, N. Karim, H. Iqbal, A. V. Vasilakos, and C. Emmanouilidis, “Evaluating and regulating agentic ai: A study of benchmarks, metrics, and regulation,” *Information Fusion*, p. 104444, 2026.
- [15] European Parliament and Council of the European Union, “Regulation (eu) 2024/1689 laying down harmonised rules on artificial intelligence (artificial intelligence act), consolidated version of 27 july 2026,” EUR-Lex, 2026, consolidated text incorporating Regulation (EU) 2026/1744. [Online]. Available: <https://eur-lex.europa.eu/eli/reg/2024/1689/2026-07-27/eng>
- [16] ISO/IEC, “ISO/IEC 42001:2023 – information technology – artificial intelligence – management system,” International Organization for Standardization, Geneva, 2023, <https://www.iso.org/standard/42001>.
- [17] —, “ISO/IEC 23894:2023 – information technology – artificial intelligence – guidance on risk management,” International Organization for Standardization, Geneva, 2023, <https://www.iso.org/standard/77304.html>.
- [18] —, “ISO/IEC 38507:2022 – information technology – governance of IT – governance implications of the use of artificial intelligence by organizations,” International Organization for Standardization, Geneva, 2022, <https://www.iso.org/standard/56641.html>.
- [19] ISO, “ISO 17572-1:2022 – Intelligent Transport Systems (ITS) – Location Referencing for Geographic Databases – Part 1: General Requirements and Conceptual Model,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO

- 17572-1:2022, Jul. 2022. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso:17572:-1:ed-3:v1:en>
- [20] ISO/IEC, “ISO/IEC 22989:2022 – Information Technology – Artificial Intelligence – Artificial Intelligence Concepts and Terminology,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO/IEC 22989:2022, Jul. 2022. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:22989:ed-1:v1:en>
- [21] —, “ISO/IEC 27000:2018 – Information Technology – Security Techniques – Information Security Management Systems – Overview and Vocabulary,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO/IEC 27000:2018, Feb. 2018. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:27000:ed-5:v1:en>
- [22] —, “ISO/IEC 2382:2015 – Information Technology – Vocabulary,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO/IEC 2382:2015, May 2015. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:ed-1:v2:en>
- [23] —, “ISO/IEC 23643:2020 – Software and Systems Engineering – Capabilities of Software Safety and Security Verification Tools,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO/IEC 23643:2020, Jun. 2020. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:23643:dis:ed-1:v1:en>
- [24] ISO, “ISO 7498-2:1989 – Information Processing Systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO 7498-2:1989, Feb. 1989. [Online]. Available: <https://www.iso.org/obp/ui/es/#iso:std:iso:7498:-2:ed-1:v1:en>
- [25] D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt, “Measuring mathematical problem solving with the math dataset,” 2021.
- [26] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, “Measuring massive multitask language understanding,” 2020.
- [27] A. Zou, Z. Wang, J. Z. Kolter, and M. Fredrikson, “Universal and transferable adversarial attacks on aligned language models,” 2023.
- [28] S. Gehrmann, S. Gururangan, M. Sap, Y. Choi, and N. A. Smith, “RealToxicityPrompts: Evaluating neural toxic degeneration in language models,” in *Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 3356–3369. [Online]. Available: <https://aclanthology.org/2020.findings-emnlp.301/>
- [29] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. D. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman *et al.*, “Evaluating large language models trained on code,” 2021.
- [30] J. Austin, A. Odena, M. Nye, M. Bosma, H. Michalewski, D. Dohan, E. Jiang, C. Cai, M. Terry, Q. Le *et al.*, “Program synthesis with large language models,” *arXiv preprint arXiv:2108.07732*, 2021.
- [31] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, “On calibration of modern neural networks,” in *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017, pp. 1321–1330.
- [32] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2002, pp. 311–318.
- [33] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Text Summarization Branches Out*, 2004, pp. 74–81.
- [34] S. Banerjee and A. Lavie, “METEOR: An automatic metric for MT evaluation with improved correlation with human judgments,” in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures*, 2005, pp. 65–72.
- [35] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “BERTScore: Evaluating text generation with BERT,” in *International Conference on Learning Representations (ICLR)*, 2020.
- [36] L. Zheng, W.-L. Chiang, Y. Sheng *et al.*, “Judging LLM-as-a-judge with MT-bench and chatbot arena,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [37] R. A. Bradley and M. E. Terry, “Rank analysis of incomplete block designs: I. the method of paired comparisons,” *Biometrika*, vol. 39, no. 3/4, pp. 324–345, 1952.
- [38] A. E. Elo, *The Rating of Chessplayers, Past and Present*. Arco, 1978.
- [39] B. Efron, “Bootstrap methods: Another look at the jackknife,” *The Annals of Statistics*, vol. 7, no. 1, pp. 1–26, 1979.
- [40] S. Zhou, F. F. Xu, H. Zhu, X. Zhou, R. Lo, A. Sridhar, X. Cheng, T. Ou, Y. Bisk, D. Fried, U. Alon, and G. Neubig, “Webarena: A realistic web environment for building autonomous agents,” in *International Conference on Learning Representations*, B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, Eds., vol. 2024, 2024, pp. 15 585–15 606. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2024/file/4410c0711e9154a7a2d26f9b3816d1ef-Paper-Conference.pdf
- [41] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, Y. Gu, H. Ding, K. Men, K. Yang, S. Zhang, X. Deng, A. Zeng, Z. Du, C. Zhang, S. Shen, T. Zhang, Y. Su, H. Sun, M. Huang, Y. Dong, and J. Tang, “Agentbench: Evaluating llms as agents,” in *International Conference on Learning Representations*, B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, Eds., vol. 2024, 2024, pp. 52 989–53 046. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2024/file/e9df36b21ff4ee211a8b71ee8b7e9f57-Paper-Conference.pdf
- [42] J. Jia, Z. Deng, Z. Chen, Y. Wang, and Z. Zheng, “Mas-fire: Fault injection and reliability evaluation for llm-based multi-agent systems,” *arXiv preprint arXiv:2602.19843*, 2026.
- [43] H. P. Zou, C. Miao, W.-C. Huang, Y. Chen, Y. Zhou, H. Zhang, Y. Wu, L. Fang, Z. Gu, Z. Zhang *et al.*, “When users change their mind: Evaluating interruptible agents in long-horizon web navigation,” *arXiv preprint arXiv:2604.00892*, 2026.
- [44] S. Kadavath, T. Conerly, A. Askell, T. Henighan, D. Drain, E. Perez, N. Schiefer, Z. Hatfield-Dodds, N. DasSarma, E. Tran-Johnson *et al.*, “Language models (mostly) know what they know,” *arXiv preprint arXiv:2207.05221*, 2022.
- [45] C. Tomani, K. Chaudhuri, I. Evtimov, D. Cremers, and M. Ibrahim, “Uncertainty-based abstention in llms improves safety and reduces hallucinations,” *arXiv preprint arXiv:2404.10960*, 2024.
- [46] K. Zhu, H. Du, Z. Hong, X. Yang, S. Guo, D. Z. Wang, Z. Wang, C. Qian, X. Tang, H. Ji *et al.*, “Multiagentbench: Evaluating the collaboration and competition of llm agents,” in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025, pp. 8580–8622.
- [47] H. Sun, S. Zhang, L. Niu, L. Ren, H. Xu, H. Fu, F. Zhao, C. Yuan, and X. Wang, “Collab-overcooked: Benchmarking and evaluating large language models as collaborative agents,” in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025, pp. 4922–4951.
- [48] M. Cemri, M. Z. Pan, S. Yang, L. A. Agrawal, B. Chopra, R. Tiwari, K. Keutzer, A. Parameswaran, D. Klein, K. Ramchandran *et al.*, “Why do multi-agent llm systems fail?” *Advances in Neural Information Processing Systems*, vol. 38, 2026.
- [49] J. Lee, R. Chang, D. Kwon, H. Singh, and N. Verma, “Gemmas: Graph-based evaluation metrics for multi agent systems,” in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 2025, pp. 1522–1532.
- [50] Y. Zhang, F. Liu, Y. Shan, X. Huang, X. Yang, Y. Zhu, X. Cheng, C. Liu, K. Zeng, T. J. Zhang *et al.*, “Silo-bench: A scalable environment for evaluating distributed coordination in multi-agent llm systems,” in *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2026, pp. 29 379–29 398.
- [51] D. A. Hudson and C. D. Manning, “Gqa: A new dataset for real-world visual reasoning and compositional question answering,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019, pp. 6693–6702.
- [52] Y. Liu, H. Duan, Y. Zhang, B. Li, S. Zhang, W. Zhao, Y. Yuan, J. Wang, C. He, Z. Liu *et al.*, “Mmbench: Is your multi-modal model an all-around player?” in *European conference on computer vision*. Springer, 2024, pp. 216–233.
- [53] H. Agrawal, K. Desai, Y. Wang, X. Chen, R. Jain, M. Johnson, D. Batra, D. Parikh, S. Lee, and P. Anderson, “Nocaps: Novel object captioning at scale,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 8948–8957.
- [54] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, “From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions,” *Transactions of the association for computational linguistics*, vol. 2, pp. 67–78, 2014.
- [55] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, “Modeling context in referring expressions,” in *European conference on computer vision*. Springer, 2016, pp. 69–85.
- [56] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach, “Towards vqa models that can read,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8317–8326.
- [57] M. Mathew, D. Karatzas, and C. Jawahar, “Docvqa: A dataset for vqa on document images,” in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 2200–2209.

- [58] A. Masry, D. X. Long, J. Q. Tan, S. Joty, and E. Hoque, “ChartQA: A benchmark for question answering about charts with visual and logical reasoning,” in *Findings of the Association for Computational Linguistics: ACL 2022*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 2263–2279. [Online]. Available: <https://aclanthology.org/2022.findings-acl.177/>
- [59] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.
- [60] A. Conneau, M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Riveria, and A. Bapna, “Fleurs: Few-shot learning evaluation of universal representations of speech,” in *2022 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2023, pp. 798–805.
- [61] Y. Li, Y. Du, K. Zhou, J. Wang, X. Zhao, and J.-R. Wen, “Evaluating object hallucination in large vision-language models,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 292–305. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.20/>
- [62] T. Guan, F. Liu, X. Wu, R. Xian, Z. Li, X. Liu, X. Wang, L. Chen, F. Huang, Y. Yacoub *et al.*, “Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 14 375–14 385.
- [63] Z. Sun, S. Shen, S. Cao, H. Liu, C. Li, Y. Shen, C. Gan, L. Gui, Y.-X. Wang, Y. Yang, K. Keutzer, and T. Darrell, “Aligning large multimodal models with factually augmented RLHF,” pp. 13 088–13 110, Aug. 2024. [Online]. Available: <https://aclanthology.org/2024.findings-acl.775/>
- [64] R. Vedantam, C. L. Zitnick, and D. Parikh, “CIDEr: Consensus-based image description evaluation,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4566–4575.
- [65] P. Anderson, B. Fernando, M. Johnson, and S. Gould, “SPICE: Semantic propositional image caption evaluation,” in *European Conference on Computer Vision (ECCV)*, 2016, pp. 382–398.
- [66] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf
- [67] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, “Improved techniques for training gans,” in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf
- [68] O. Sainz, J. Campos, I. García-Ferrero, J. Etxaniz, O. L. de Lacalle, and E. Agirre, “NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark,” in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 10 776–10 787. [Online]. Available: <https://aclanthology.org/2023.findings-emnlp.722/>
- [69] M. Mizrahi, G. Kaplan, D. Malkin, R. Dror, D. Shahaf, and G. Stanovsky, “State of what art? a call for multi-prompt llm evaluation,” *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 933–949, 2024.
- [70] ISO/IEC, “ISO/IEC 25024:2015 – Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – Measurement of Data Quality,” International Organization for Standardization, Geneva, Switzerland, International Standard ISO/IEC 25024:2015, Oct. 2015. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25024:ed-1:v1:en>
- [71] European Parliament and Council of the European Union, “Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence (artificial intelligence act),” OJ L, 2024/1689, 12 July 2024, 2024, eLI: <http://data.europa.eu/eli/reg/2024/1689/oj>.
- [72] European Commission, “Guidelines on the scope of the obligations for general-purpose AI models established by regulation (eu) 2024/1689 (AI act),” European Commission, Brussels, C(2025) 5045 final, 2025, adopted 18 July 2025. <https://digital-strategy.ec.europa.eu/en/library/guidelines-scope-obligations-providers-general-purpose-ai-models-under-ai-act>
- [73] —, “The general-purpose ai code of practice,” Jul. 2025, published 10 July 2025. [Online]. Available: <https://digital-strategy.ec.europa.eu/en/policies/ai-code-practice>
- [74] ISO/IEC, “ISO/IEC TR 5469:2024 – artificial intelligence – functional safety and AI systems,” International Organization for Standardization, Geneva, 2024, <https://www.iso.org/standard/81283.html>.

APPENDIX A

METRIC DESCRIPTOR CATALOG

Legend. *Basis*: ref = reference-based, rf = reference-free, judge, behav = behavioral. *Scale*: bin, ord, cont, ratio. *Form*: point, rate, distrib, comp = comparative, corr = correlation. *Dir*: ↑ higher-better, ↓ lower-better, □ target-band. *Norm* maps native output onto the shared 0–4 band using one of six families (codes shown in the column): H=higher-is-better rate (0: [0, .20] 1: [.20, .40] 2: [.40, .60] 3: [.60, .80] 4: [.80, 1]); L=lower-is-better rate (4: [0, .05] 3: [.05, .10] 2: [.10, .20] 1: [.20, .40] 0: [.40, 1]); C(B)=lower-is-better continuous (band = 4 – ⌊4v/B⌋); R=ratio-to-budget (4: v ≤ .25B ... 0: v > B; partner-declared B); W[lo,hi]=target-band window (inside=4; linear decay outside); Q=comparative quartile rank within the shared pool. Veto overrides are marked (v).

ACKNOWLEDGMENT

The authors thank [...]. This work was supported by [...].

TABLE VII: LLM (output-level) metric descriptors

Metric	Dim	Basis	Scale	Form	Dir	Native output	Norm	Data need	Failure signal
Accuracy / Exact Match	Capability	ref	bin→cont	rate	↑	% matching gold	H	reference labels	wrong final answers
F1	Capability	ref	cont	point	↑	0–1 overlap	H	gold spans	partial / missed matches
Pass@k (code)	Capability	ref (exec)	bin→rate	rate	↑	% solved in k	H	test suite / env	code fails tests
BLEU	Capability	ref	cont	point	↑	0–1 n-gram precision	H	reference text	low overlap w/ ref
ROUGE	Capability	ref	cont	point	↑	0–1 recall overlap	H	reference summary	missing ref content
METEOR	Capability	ref	cont	point	↑	0–1 stem / synonym-aware	H	reference text	semantic mismatch
BERTScore	Capability	ref (embed)	cont	point	↑	0–1 embed similarity	H	reference text	semantic divergence
LLM-judge / win-rate / Elo	Capability	judge	ord / comp	comp	↑	win % or Elo	Q	judge + opponent set	judged worse than base
Perplexity	Capability	rf	ratio	point	↓	exp(avg NLL)	C*	held-out text	poor LM fit
ECE (calibration)	Transparency	ref	cont	distrib	↓	0–1 conf–acc gap	C	labels + confidences	over- / under-confident
Faithfulness rate	Safety	judge / NLI	cont	rate	↑	% claims supported	H	source doc	unsupported claims
Toxicity	Safety	rf (clf)	cont	rate	↓	0–1 prob / % toxic	L	none (classifier)	harmful output
Refusal rate	Safety	rule / judge	cont	rate	⊠	% prompts refused	W	safe / unsafe prompt set	over- or under-refusal
Bias / fairness gap	Fairness	comp	cont	comp	↓	delta across groups	C	group-labeled data	disparate performance
Abstention / deferral rate	Oversight	rule / judge	cont	rate	⊠	% uncertain cases declined	W	answerable / unanswerable set	answers when it shouldn't
Escalation appropriateness	Oversight	judge / ref	cont	rate	↑	precision–recall of “needs human”	H	gold escalation labels	misses review / false alarms
Inference latency	Efficiency	rf	ratio	point	↓	sec / response	R	instrumentation	too slow for use case
Throughput	Efficiency	rf	ratio	rate	↑	tokens or queries / sec	R	instrumentation	low serving capacity
Cost per query	Efficiency	rf	ratio	point	↓	\$ per request	R	pricing + usage	over budget
Memory footprint	Efficiency	rf	ratio	point	↓	params / VRAM	R	model spec	won't fit target hardware
Energy / carbon per query	Efficiency	rf	ratio	point	↓	kWh or gCO ₂ e per request	R	power / carbon instrumentation	excess energy / emissions

* Perplexity comparable only within identical tokenizer / corpus.

TABLE VIII: Agent (trajectory-level) metric descriptors: single-agent instruments)

Metric	Dim	Basis	Scale	Form	Dir	Native output	Norm (family [cuts])	Data need	Failure signal
Task success rate	Capability	behav	bin→rate	rate	↑	% goal achieved	H [0/.20/.40/.60/.80]	goal oracle / env	goal not reached
Sub-goal / progress	Capability	behav	cont	rate	↑	% sub-goals done	H [0/.20/.40/.60/.80]	annotated checkpoints	stalls partway
Pass@k (agent)	Capability	behav	bin→rate	rate	↑	% solved in k tries	H [0/.20/.40/.60/.80]	env / goal oracle	fails across all k
Tool-call accuracy	Capability	behav	bin→rate	rate	↑	% correct tool+args	H [0/.20/.40/.60/.80]	gold trajectory / env	wrong tool / bad args
Step efficiency	Efficiency	behav	cont	comp	↑	$\min(1, L^* / L)$	H [0/.20/.40/.60/.80]	optimal path	redundant steps
Trajectory optimality	Capability	judge / behav	ord / cont	comp	↑	path quality vs ref	Q (pool quartile)	ref path or judge	convoluted path
Error recovery rate	Robustness	behav	bin→rate	rate	↑	% failures recovered	H [0/.20/.40/.60/.80]	env w/ injected faults	can't recover
Cost (tokens / \$ / latency)	Efficiency	rf	ratio	point / rate	↓	tokens, \$, seconds	R (partner budget)	instrumentation	over budget / slow
Unsafe action rate	Safety	rule / behav	bin→rate	rate	↓	% steps violating	L [.40/.20/.10/.05/0] (v) ≥ 0.20	constraint spec / monitor	dangerous actions
Consistency	Robustness	rf	cont	distrib	↓	variance across runs	C ($B=0.10$)	repeated runs	unstable outcomes
Interruptibility / stop-success	Oversight	behav	bin→rate	rate	↑	% clean halts on stop	H [0/.20/.40/.60/.80]	env w/ stop injection	ignores / mishandles stop
Human-override success	Oversight	behav	bin→rate	rate	↑	% overrides applied	H [0/.20/.40/.60/.80]	env w/ interventions	discards human input
Appropriate help-seeking	Oversight	judge / behav	cont	rate	⊠	% asks only when needed	W [.10,.30]	tasks w/ need points	never asks / asks constantly
Deferral-on-uncertainty	Oversight	behav	bin→rate	rate	↑	% low-conf steps escalated	H [0/.20/.40/.60/.80]	confidence-instrumented env	barrels ahead when unsure
Confirmation-gating	Oversight	rule / behav	bin→rate	rate	↑	% high-impact actions gated	H [0/.20/.40/.60/.80] (v) < 0.80	action-impact spec	irreversible action w/o sign-off

TABLE IX: Multi-agent trajectory-level metric descriptors.

Metric / Dim.	Type	Output / Band	Evidence / Failure
Communication quality (<i>Capability</i>)	judge; cont.; rate; ↑	% relevant/grounded msgs.; H [.20/.40/.60/.80]	Message log + rubric; off-topic/ungrounded communication
Delegation accuracy (<i>Capability</i>)	behav.; bin.; rate; ↑	% subtasks assigned correctly; H [.20/.40/.60/.80]	Role assignment; incorrect agent selected
Coordination efficiency (<i>Efficiency</i>)	behav.; ratio; comp.; ↓	Msgs./steps relative to optimum; R (budget)	Reference trace; redundant/duplicated work
Conflict resolution (<i>Robustness</i>)	behav.; bin.; rate; ↑	% conflicts resolved; H [.20/.40/.60/.80]	Induced disagreement; incorrect reconciliation
Error containment (<i>Robustness</i>)	behav.; bin.; rate; ↑	% injected faults contained; H [.20/.40/.60/.80]	Fault injection; cascading failure
Deadlock/livelock (<i>Robustness</i>)	behav.; bin.; rate; ↓	% stalled/looping runs; L [.40/.20/.10/.05/0]	Long-horizon tasks; indefinite stall/loop
Verification correctness (<i>Robustness</i>)	behav.; bin.; rate; ↑	% outcomes verified; H [.20/.40/.60/.80]	Verification oracle; incorrect verification
Termination correctness (<i>Robustness</i>)	behav.; bin.; rate; ↑	% correctly terminated; H [.20/.40/.60/.80]	Terminal oracle; premature/failed termination
UPR (<i>Efficiency</i>)	behav.; ratio; rate; ↓	Redundant reasoning fraction; L [.40/.20/.10/.05/0]	Communication graph; non-contributing paths
RCC (<i>Efficiency</i>)	comp.; ratio; rate; ↓	$\max(0, 1 - S_{MAS}/S_1)$; L [.40/.20/.10/.05/0]	Single-agent baseline; coordination loss
Unsafe-action rate (<i>Safety</i>)	rule/behav.; bin.; rate; ↓	% violating steps; L [.40/.20/.10/.05/0]	Constraint monitor; compounding unsafe actions

Note: Type = basis; scale; form; direction. H/L/R = higher-is-better, lower-is-better, and ratio-to-budget families. Metrics are defined but not yet operationalized in v1.0.

TABLE X: Multimodal cross-modal metric descriptors.

Metric / Dim.	Type	Output / Band	Evidence / Failure
VQA accuracy (<i>Capability</i>)	ref.; bin.; rate; ↑	% correct; H [.20/.40/.60/.80]	Q–A pairs; incorrect visual reasoning
CIDEr/SPICE (<i>Capability</i>)	ref.; cont.; point; ↑	Caption quality; H [.20/.40/.60/.80]	Reference captions; inaccurate captions
Recall@k (<i>Capability</i>)	ref.; bin.; rate; ↑	% correct in top-k; H [.20/.40/.60/.80]	Paired corpus; incorrect modality match
CLIPScore (<i>Capability</i>)	embed.; cont.; point; ↑	Image–text similarity; H [.20/.40/.60/.80]	Encoder; weak cross-modal alignment
FID/IS (<i>Capability</i>)	ref.; cont.; distrib.; ↓	Distance to real distribution; C ($B = 300$)	Reference images; unrealistic generation
Grounding accuracy (<i>Capability</i>)	ref.; bin.; rate; ↑	% correct regions; H [.20/.40/.60/.80]	Region labels; incorrect grounding
OCR accuracy (<i>Capability</i>)	ref.; cont.; rate; ↑	1–CER / % correct; H [.20/.40/.60/.80]	Transcription; misread visual text
WER (<i>Capability</i>)	ref.; ratio; rate; ↓	Word error rate; L [.40/.20/.10/.05/0]	Transcript; ASR errors
Cross-modal hallucination (<i>Safety</i>)	judge; cont.; rate; ↓	% unsupported claims; L [.40/.20/.10/.05/0]	Source modality; absent content described
Modality robustness (<i>Robustness</i>)	comp.; cont.; comp.; ↓	Perturbation degradation; C ($B = .30$)	Perturbed pairs; sensitivity to noise/dropout

Note: Type = basis; scale; form; direction. H/L/C = higher-is-better, lower-is-better, and continuous families.

TABLE XI: Meta-evaluation metric descriptors.

Metric / Dim.	Type	Output / Band	Evidence / Failure
Judge–human agreement (<i>Transparency</i>)	comp.; cont.; corr.; ↑	Correlation with human ratings; H [.20/.40/.60/.80]	Human ratings; judge–human disagreement
Inter-rater reliability (<i>Transparency</i>)	ref.; cont.; distrib.; ↑	κ / α ; H [.20/.40/.60/.80]	≥ 2 raters; rater disagreement
Metric validity (<i>Governance</i>)	comp.; cont.; corr.; ↑	Correlation with criterion; H [.20/.40/.60/.80]	Criterion measure; metric tracks noise
Discriminative power (<i>Governance</i>)	ref.; cont.; distrib.; ↑	Score separation; H [.20/.40/.60/.80]	Multi-system scores; poor differentiation
Test–retest stability (<i>Robustness</i>)	ref.; cont.; distrib.; ↑	Rerun variance; C ($B = .10$)	Repeated runs; unstable scores
Sensitivity (<i>Governance</i>)	comp.; cont.; comp.; ↑	Detection of known differences; H [.20/.40/.60/.80]	Known-gap systems; missed differences
Contamination/leakage (<i>Governance</i>)	rule; bin.; rate; ↓	% test items in training; L [.40/.20/.10/.05/0]	Train/test overlap; inflated scores
Judge bias (<i>Fairness</i>)	comp.; cont.; comp.; ↓	Score shift from artifact; C ($B = .15$)	Controlled probes; length/position/self bias

Note: Type = basis; scale; form; direction. corr. = correlation; κ/α = agreement coefficients.