Lecture 7: TEVV for LLM applications

Testing, Evaluation, Verification and Validation for LLM Applications

Patrick Hall

Department of Decision Sciences

November 2, 2026

© 2026–2027 Patrick Hall (jphall@gwu.edu). Some rights reserved. Text and copy are licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0).

Agenda

Measurement foundations

Model testing

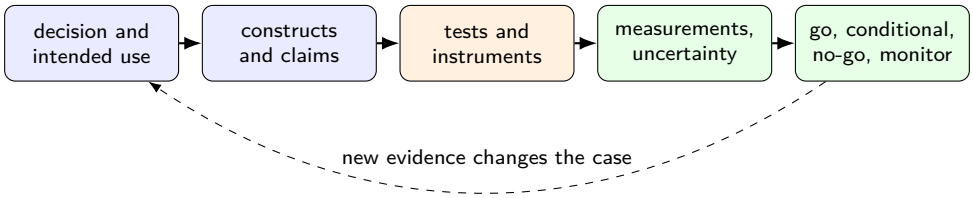
Red teaming

Field testing

Validation and synthesis

References

TEVV builds evidence for a decision



A decision-centered evidence chain connects intended use to constructs, instruments, measurements, and action, with monitoring feeding back into reassessment.

TEVV

Test behavior; **evaluate** results against criteria; **verify** that requirements were implemented; **validate** fitness for the intended use.

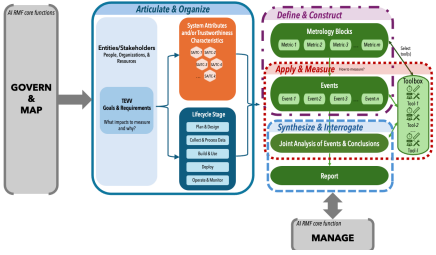
Central discipline

Do not begin with a convenient score. Begin with the claim and decision the score must support.

Source: (Phillips et al. 2026), decision-centered, contextual TEVV.

Measurement connects evidence to governance

- **Context:** start with organizational goals, stakeholders, and lifecycle context.
- **Constructs:** define trustworthy characteristics as measurable constructs.
- **Instrument validation:** assess reliability, validity, sensitivity, and intended use.
- **Measurement:** apply tools through planned events.
- **Synthesis:** form evidence and qualified conclusions.
- **Governance:** use the evidence to support Govern, Map, and Manage decisions.



The TEVV-Athlon framework connects structured measurement evidence to the AI RMF Govern, Map, and Manage functions.

Source: (Wallach et al. 2024); (Phillips et al. 2026), Figure 3(b).

Measurement science makes TEVV claims defensible

Evaluate the system and the instrument

For LLMs, instruments include prompts, benchmark items, rubrics, raters, surveys, interview protocols, and scoring code. Their quality constrains the claims a score can support.

Measurement quality

- Define the objective, construct, and intended interpretation before selecting a metric.
- Validate that the instrument captures the intended property, covers important facets, generalizes beyond test conditions, and relates to real-world outcomes.
- Calibrate against baselines or ground truth; report uncertainty, variability, assumptions, and limitations.

Experimental and process design

- Select data that reflect the intended context and prevent test-data contamination.
- Specify variables, conditions, procedures, and comparison points; repeat and sample to expose variation.
- Preserve enough detail for verification, invite independent challenge, and revalidate as systems, users, and environments change.

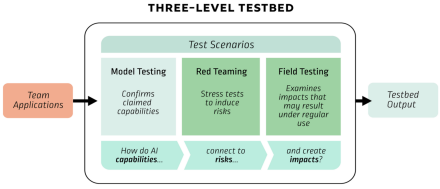
Source: (Phillips et al. 2026), Section 4.4–4.5 and Appendices D–E.

Model, adversarial, and field tests answer different questions

Model testing	Can scripted inputs confirm claimed capabilities and guardrails?
Red teaming	Can motivated testers discover adverse outcomes or defeat safeguards?
Field testing	How do representative people use, interpret, rely on, and act on the system in context?

No level dominates the others. Together they move from repeatable component evidence toward sociotechnical effects.

Source: (Amironesei et al. 2025), ARIA 0.1 pilot evaluation.



ARIA connects model testing, red teaming, and field testing within one layered evaluation design.

A benchmark is a measurement instrument

Benchmark specification

Population and sampling frame → task cases → prompt and decoding protocol → scorer → aggregation → claim.

- Define the construct before selecting the dataset.
- Freeze versions, prompts, tools, and settings.
- Include baselines, negative controls, slices, and failures.
- Report sample size and uncertainty, not only a rank.



A model-testing workflow combines benchmark cases and embedding-based diagnostics rather than relying on one aggregate score.

Source: (Hendrycks et al. 2021); (Chen et al. 2021); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute).

Score the task, then inspect uncertainty and slices

For binary correctness on n cases:

$$\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}(\hat{y}_i = y_i)$$

$$\mathrm{SE}(\widehat{p}) \approx \sqrt{\frac{\widehat{p}(1-\widehat{p})}{n}}.$$

- Prefer confidence intervals or bootstrap intervals to naked point estimates.
- Compare paired outputs when systems see the same cases.
- Break results down by task, language, demographic group, difficulty, and failure mode.



Error Metrics for PAY_0

PAY_0	Prevalence	Accuracy	True Positive Rate	Precision	Specificity	Negative Predicted Value	False Positive Rate	False Discovery Rate	False Negative Rate	False Omissions Rate
-2	0.124	0.864	0.099	0.333	0.972	0.884	0.028	0.667	0.901	0.116
-1	0.168	0.816	0.206	0.406	0.939	0.854	0.061	0.594	0.794	0.146
0	0.121	0.867	0.107	0.341	0.972	0.888	0.028	0.659	0.893	0.112
1	0.325	0.491	0.903	0.381	0.292	0.862	0.708	0.619	0.097	0.138
2	0.709	0.709	1	0.709	0	0.5	1	0.291	0	0.5
3	0.748	0.748	1	0.748	0	0.5	1	0.252	0	0.5
4	0.571	0.571	1	0.571	0	0.5	1	0.429	0	0.5
5	0.444	0.444	1	0.444	0	0.5	1	0.556	0	0.5
6	0.25	0.25	1	0.25	0	0.5	1	0.75	0	0.5
7	0.5	0.5	1	0.5	0	0.5	1	0.5	0	0.5
8	0.75	0.75	1	0.75	0	0.5	1	0.25	0	0.5

Error Metrics for SEX

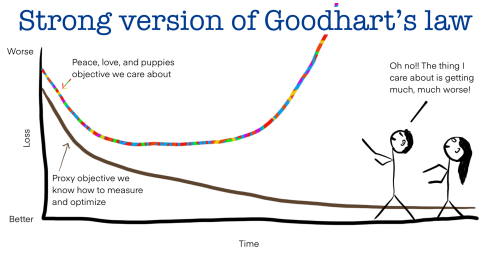
SEX										
Male	0.235	0.782	0.626	0.531	0.83	0.879	0.17	0.469	0.374	0.121
Female	0.209	0.797	0.552	0.514	0.862	0.879	0.138	0.486	0.448	0.121

Segmented error analysis reveals performance differences hidden by an overall average.

Source: Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): segmented error analysis; (Phillips et al. 2026).

Benchmark scores can support claims they do not warrant

- **Construct underrepresentation**: the task covers only an easy proxy.
- **Contamination**: test items or near-duplicates enter development data.
- **Repeated reuse**: leaderboard feedback turns the test into training information.
- **Aggregation**: averages hide slices, variance, abstentions, and severe failures.
- **Staleness**: users, threats, models, and environments change.
- **Goodhart pressure**: optimizing the score weakens its relationship to the goal.



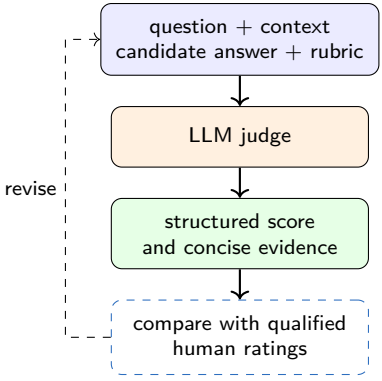
A Goodhart's Law teaching graphic warns that a target metric may cease to represent the intended goal.

Source: (Thomas and Uminsky 2022); (Balloccu et al. 2024); Hall, 2026, *Guest Lecture for Prof. Pless, GW Computer Science*.

LLM-as-a-judge converts a rubric into model-mediated labels

1. Fix the question, reference/context, candidate answer, and rubric.
2. Request a constrained score and concise evidence.
3. Parse and validate the schema; keep raw outputs.
4. Randomize answer order for pairwise judging.
5. Repeat or use multiple judges to estimate instability.
6. Calibrate against qualified human ratings.

A judge is not ground truth: position, verbosity, self-preference, prompt wording, and shared model failures can bias judgments.



An evaluation package passes through an LLM judge to produce structured scores and evidence, which are checked against qualified human ratings and used to revise the instrument.

Source: (Zheng et al. 2023); (Liu et al. 2023); Hall, 2026, *Guest Lecture for Prof. Pless, GW Computer Science*.

Validate the judge as an instrument

Signal	Test	Measurement interpretation
Rubric design	Does it define and cover the target construct?	Construct/content validity
Process integrity	Are data, prompts, settings, parsing, and scoring correct?	Quality assurance
Small-change stability	Do repeated runs and irrelevant perturbations agree?	Reliability/robustness
Large-change sensitivity	Do construct-relevant changes move scores as expected?	Construct validity/calibration
Null detection	Can the judge abstain on unratable inputs?	Calibration
Similarity/difference	Are close cases close and meaningfully different cases separated?	Convergent/discriminant validity
Human agreement	Does the judge agree with qualified raters on held-out cases?	Criterion validity

Report the instrument: judge model/version, rubric, prompt, order policy, settings, parser, repeats, agreement, calibration sample, and known failure modes.

Source: Hall, unpublished synthesis; validity and V&V foundations (Messick 1987; Cronbach and Meehl 1955; Campbell and Fiske 1959; National Aeronautics and Space Administration 2016; Institute of Electrical and Electronics Engineers 2025); operational checks (Alvarez-Melis and Jaakkola 2018; Yeh et al. 2019; Adebayo et al. 2018; Geifman and El-Yaniv 2017; Artstein and Poesio 2008; Zheng et al. 2023).

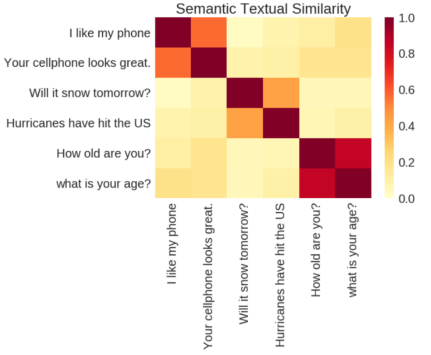
Embeddings and classifiers test specific, bounded properties

For normalized embeddings $e(x), e(y) \in \mathbb{R}^d$,

$$s(x, y) = e(x)^\top e(y)$$

can support retrieval relevance, semantic similarity, clustering, duplicate detection, and drift analysis.

- A trained classifier can test toxicity, topic, policy class, or failure type at scale.
- Validate labels, thresholds, calibration, subgroup behavior, and domain shift.
- Embedding geometry is model-dependent; a two-dimensional projection is exploratory evidence.

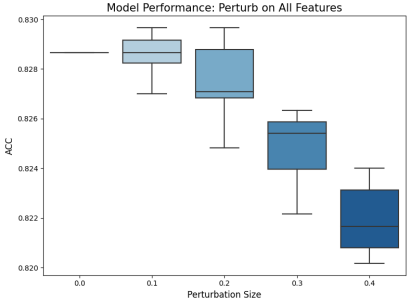


Sentence embeddings map text into vectors whose geometry supports similarity, clustering, and classification tasks.

Source: (Reimers and Gurevych 2019); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute).

Use a battery of model tests, not one metric

Method	Question
Exact/schema checks	Is the output structurally and deterministically correct?
Reference metrics	Does it match a known answer or executable verifier?
Perturbation tests	Is behavior invariant to irrelevant changes?
Metamorphic tests	Do related inputs satisfy expected relations?
Slice/error analysis	Where and how does it fail?
Embedding tests	Does it express a bounded semantic property?
Repeated trials	How variable is stochastic behavior?



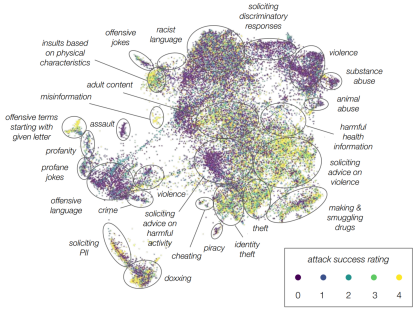
A drift diagram shows data moving away from the training distribution, motivating repeated and segmented testing.

Source: (Phillips et al. 2026); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): robustness to drift.

Red teaming searches for consequential failures

AI red teaming is a structured, authorized effort to find flaws and vulnerabilities under an adversarial mindset.

- Start with assets, actors, access, harms, and system boundaries.
- State rules of engagement, containment, escalation, and stop conditions.
- Test the application, tools, retrieval layer, interface, and people—not only the base model.
- Record attack preconditions, transcripts, impact, reproducibility, and mitigations.



Anthropic clusters red-team attacks by harm type and colors them by the human tester's attack-success rating.

Source: (Perez et al. 2022); (Ganguli et al. 2022), Anthropic research memo, Figure 1.1; Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): what is red teaming?

Selected red-teaming methods

Security and privacy

Test availability failures such as resource exhaustion; confidentiality failures such as prompt leakage, extraction, inversion, and membership inference; and integrity failures such as injection, poisoning, obfuscation, and retrieval manipulation.

Fairness harms

Hold the task constant while varying demographic groups, languages, or dialects. Counterfactual prompting tests whether comparable cases produce disparate outputs or outcomes.

Safeguards and harmful capabilities

Probe jailbreak susceptibility through multi-turn attacks, role-play, benign framing, or task blending; test malicious-code elicitation and unsafe use of connected tools or excessive agency.

Factuality and knowledge robustness

Probe expert and niche knowledge, temporal knowledge, numerical reasoning, and false premises to test whether outputs remain accurate under challenging or misleading conditions.

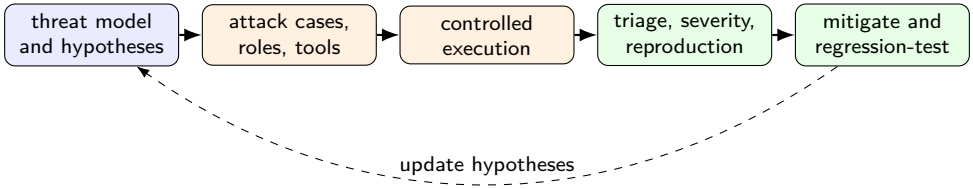
Source: (Phillips et al. 2026), Appendix B; Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): red-teaming methods.

Selected red-teaming methods

Method family	Outcome under test	Example probes
Availability attacks	Can service be degraded, delayed, or denied?	Sponge examples, repeated-token attacks, input overflows
Confidentiality attacks	Can confidential information or internal functionality be exposed?	Model extraction or inversion, membership inference, system-prompt leakage
Counterfactual fairness prompting	Do comparable cases receive disparate outputs or outcomes?	Hold the task fixed; vary demographic groups, languages, or dialects
Factuality and knowledge robustness	Do outputs remain accurate under challenging or misleading conditions?	Expert, niche, temporal, and numerical probes; false premises
Integrity attacks	Can inputs or data manipulate intended outputs or outcomes?	Direct or indirect injection, poisoning, obfuscation, retrieval weaknesses
Jailbreaking	Can prompting induce violations of safeguards?	Multi-turn or poetic attacks, benign framing, role-play, task blending
Malicious-code elicitation	Can the system produce exploitative or covertly harmful code?	Malware, hidden payloads, unsafe scripts, backdoored implementations
Agent or tool abuse	Can connected tools or external actions be misused?	Unsafe tool selection, excessive agency, unauthorized or harmful execution

Source: Adapted from (Phillips et al. 2026), Appendix B, Table 4.

Treat a red-team campaign as an experiment



A red-team campaign cycles from threat hypotheses through controlled attacks, triage, mitigation, and regression testing.

Denominators matter

Report success per attempt, per unique attack family, per tester, and per severity tier; include duplicate handling and tester exposure.

Absence is bounded evidence

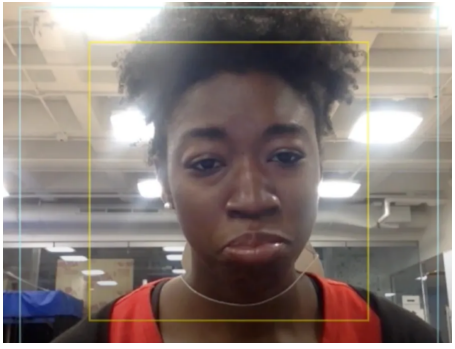
"No failure found" means no failure under the tested access, time, skills, tools, and cases—not that the system is safe.

Source: (Perez et al. 2022); (National Institute of Standards and Technology 2023).

Field testing closes the context gap

Field tests examine representative use in realistic settings, including the human–AI configuration and downstream actions.

- Who uses the system, for which tasks, under what incentives and time pressure?
- What do people understand, trust, ignore, verify, or act upon?
- How do interfaces, policies, handoffs, and organizational controls shape outcomes?
- Which benefits and harms appear across groups and over time?



Joy Buolamwini's demonstration showed a facial-analysis system failing to detect her face until she used a white mask, illustrating why field testing must include affected users and realistic conditions.

Source: Joy Buolamwini, 2016, *How I'm fighting bias in algorithms* (video still); (Amironesei et al. 2025); (Weidinger et al. 2023); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): what is field testing?

Selected field testing methods

Method family	Outcome under test	Example evidence
Impact assessment	What benefits and harms could the system create for people, organizations, and resources?	Impacts, affected groups, severity, likelihood, and mitigations
Field pilots	How does the system perform in a limited, realistic operational deployment?	Task outcomes, failures, workarounds, incidents, and operational constraints
Focus groups and interviews	How do participants understand and experience the system?	Themes, unmet needs, concerns, expectations, and contextual detail
Structured human-subject experiments	How does the system change human behavior, decisions, or outcomes under controlled conditions?	Comparisons across conditions, behavioral measures, and subgroup effects
Post-deployment feedback	Which benefits, failures, and harms emerge after release?	Ratings, complaints, support records, incident reports, and usage patterns
Usability and UX research	Can intended users interact with the system effectively, efficiently, and satisfactorily?	Task completion, errors, time on task, comprehension, and satisfaction
User surveys	What standardized perceptions and experiences do users or stakeholders report?	Scaled responses, distributions, open-text feedback, and subgroup comparisons

Source: Adapted from (Phillips et al. 2026), Appendix C, Table 5.

Example field-testing rubric

Dimension	1 point	2 points	3 points
Accuracy	Incorrect or misleading	Mostly correct; minor issues	Correct and well-qualified
Relevance	Off-topic	Partly responsive	Directly responsive
Completeness	Major omissions	Covers basics	Covers all key parts
Clarity	Confusing	Understandable	Clear and well organized
Instructions	Misses key instructions	Mostly follows	Fully follows
Usefulness	Not helpful	Somewhat helpful	Practical and helpful
Safety / fit	Unsafe or inappropriate	Generally safe; minor concerns	Safe and appropriate

Checklist: realistic scenarios; 30–50 ratings per use case for analysis; at least three raters for agreement; repeated key questions for response variance; per-dimension review; avoid redundant dimensions and floor/ceiling effects.

Interpretation

18–21: strong
13–17: weak
7–12: unacceptable

Source: Hall, 2026, *Field Testing Rubric*, CC BY-NC-SA 4.0.

Design field evidence with people, not around them

- Recruit participants that reflect intended users and affected groups.
- Randomize or counterbalance order; define comparison conditions.
- Capture task success, time, reliance, overrides, confidence, comprehension, and downstream actions.
- Analyze participant, rater, scenario, and run-level variation.
- Plan consent, privacy, compensation, accessibility, deception, debriefing, and escalation.
- Use institutional human-subjects review when the activity is research involving people.

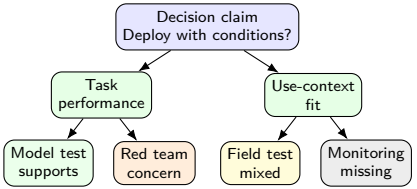


In 2012, Facebook altered users' News Feeds to test emotional contagion without obtaining informed consent, prompting a major research-ethics controversy.

Source: Chris Chambers, 2014, *Facebook fiasco: was Cornell's study of "emotional contagion" an ethics breach?*, *The Guardian*; (Amironesei et al. 2025); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute): human-subjects research.

Synthesize evidence without hiding disagreement

1. Map each claim to model, red-team, field, process, and monitoring evidence.
2. Preserve separate results before aggregation.
3. Weight evidence by relevance, quality, independence, and uncertainty.
4. Record contradictions and missing branches explicitly.
5. Tie acceptance criteria to a decision: deploy, deploy with conditions, defer, or reject.



A measurement tree keeps supporting, concerning, mixed, and missing evidence on separate branches below decision-relevant constructs.

Measurement tree

Leaves are observed indicators; intermediate nodes are constructs; the root is the decision-level claim. Aggregation rules must be transparent and contestable.

See NIST AI 700-2 for applied measurement trees.

Source: (Amironesei et al. 2025); (Phillips et al. 2026); Hall, 2026, *AI Governance Accelerator* (GW TRAILS Institute).

TEVV is a maintained argument, not a one-time score

Before deployment

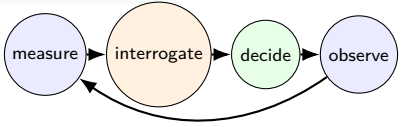
Define the intended use and unacceptable outcomes; validate instruments; establish baselines, slices, adversarial cases, field evidence, and acceptance criteria.

After deployment

Monitor drift, incidents, overrides, feedback, and subgroup outcomes; rerun regression suites after model, prompt, retrieval, tool, policy, or interface changes.

Five questions to carry forward

1. What decision is this evidence for?
2. What construct does the instrument measure?
3. What important context is missing?
4. How reliable is the measurement in its context of use?
5. What result would cause you to question the measurement?



Measurement, interrogation, decision, and observation form a lifecycle.
Source: (National Institute of Standards and Technology 2023); (Phillips et al. 2026).

References

Adebayo, Julius, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim (2018). "Sanity Checks for Saliency Maps". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett. Vol. 31. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/294a8ed24b1ad22ec2e7efea049b8737-Paper.pdf.

Alvarez-Melis, David and Tommi S. Jaakkola (2018). "On the Robustness of Interpretability Methods". In: *arXiv preprint arXiv:1806.08049*. URL: https://arxiv.org/abs/1806.08049.

Amironesei, Razvan, Afzal Godil, Craig Greenberg, Kristen Greene, Patrick Hall, Theodore Jensen, Jonathan Fiscus, and Noah Schulman (2025). *Assessing risks and impacts of AI (ARIA): ARIA 0.1 pilot evaluation report*. Tech. rep. AI 700-2. National Institute of Standards and Technology. URL: https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.700-2.pdf.

Artstein, Ron and Massimo Poesio (2008). "Survey Article: Inter-Coder Agreement for Computational Linguistics". In: *Computational Linguistics* 34.4, pp. 555–596. URL: https://aclanthology.org/J08-4004/.

Balloccu, Simone, Patrícia Schmidtová, Mateusz Lango, and Ondřej Dušek (2024). *Leak, cheat, repeat: data contamination and evaluation malpractices in closed-source LLMs*. arXiv: 2402.03927. URL: https://doi.org/10.18653/v1/2024.eacl-long.5.

Campbell, D. T. and D. W. Fiske (1959). "Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix". In: *Psychological Bulletin* 56.2, pp. 81–105. URL: https://doi.org/10.1037/h0046016.

References

Chen, Mark, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. (2021). *Evaluating large language models trained on code*. arXiv: 2107.03374. URL: https://arxiv.org/abs/2107.03374.

Cronbach, L. J. and P. E. Meehl (1955). "Construct Validity in Psychological Tests". In: *Psychological Bulletin* 52.4, pp. 281–302. URL: https://doi.org/10.1037/h0040957.

Ganguli, Deep, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. (2022). *Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned*. Anthropic. URL: https://arxiv.org/abs/2209.07858.

Geifman, Yonatan and Ran El-Yaniv (2017). "Selective Classification for Deep Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/4a8423d5e91fda00bb7e46540e2b0cf1-Paper.pdf.

Hendrycks, Dan, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt (2021). "Measuring massive multitask language understanding". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=d7KBjmI3GmQ.

Institute of Electrical and Electronics Engineers (2025). *IEEE Standard for System, Software, and Hardware Verification and Validation*. URL: https://ieeexplore.ieee.org/document/11134780.

References

Liu, Yang, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu (2023). *G-Eval: NLG Evaluation Using GPT-4 with Better Human Alignment*. URL: https://arxiv.org/abs/2303.16634.

Messick, Samuel (1987). "Validity". In: *ETS Research Report Series* 1987.2, pp. i–213. URL: https://doi.org/10.1002/j.2330-8516.1987.tb00244.x.

National Aeronautics and Space Administration (2016). *NASA Systems Engineering Handbook*. Tech. rep. NASA/SP-2016-6105 Rev2. National Aeronautics and Space Administration. URL: https://www.nasa.gov/wp-content/uploads/2018/09/nasa_systems_engineering_handbook_0.pdf.

National Institute of Standards and Technology (2023). *Artificial intelligence risk management framework (AI RMF 1.0)*. URL: https://doi.org/10.6028/NIST.AI.100-1.

Perez, Ethan, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving (2022). *Red teaming language models with language models*. arXiv: 2202.03286. URL: https://arxiv.org/abs/2202.03286.

Phillips, P. Jonathon, Theodore Jensen, Patrick Hall, Razvan Amironesei, Yee-Yin Choong, Craig Greenberg, and Kristen K. Greene (2026). *The TEVV-Athlon Framework for Evaluating AI Systems*. Tech. rep. NIST AI 200-2 ipd. National Institute of Standards and Technology. URL: https://doi.org/10.6028/NIST.AI.200-2.ipd.

Reimers, Nils and Iryna Gurevych (2019). "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 3982–3992. URL: https://aclanthology.org/D19-1410/.

References

Thomas, Rachel L. and David Uminsky (2022). "Reliance on metrics is a fundamental challenge for AI". In: *Patterns* 3.5. URL: https://doi.org/10.1016/j.patter.2022.100476.

Wallach, Hanna M. et al. (2024). *Evaluating generative AI systems is a social science measurement challenge*. arXiv: 2411.10939v1. URL: https://arxiv.org/abs/2411.10939v1.

Weidinger, Laura et al. (2023). *Sociotechnical safety evaluation of generative AI systems*. arXiv: 2310.11986. URL: https://arxiv.org/abs/2310.11986.

Yeh, Chih-Kuan, Cheng-Yu Hsieh, Arun Suggala, David I. Inouye, and Pradeep K. Ravikumar (2019). "On the (In)fidelity and Sensitivity of Explanations". In: *Advances in Neural Information Processing Systems*. Vol. 32. URL: https://proceedings.neurips.cc/paper_files/paper/2019/hash/a7471fdc77b3435276507cc8f2dc2569-Abstract.html.

Zheng, Lianmin et al. (2023). "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena". In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine. Vol. 36. Curran Associates, Inc., pp. 46595–46623. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf.