Crimson Publishers
Wings to the Research

Review Article

Toward Robust and Scalable Evaluation of AI Models: Past Practices, Current Tooling and Future Directions

Maikel Leon*

Department of Business Technology, Miami Herbert Business School, University of Miami, USA

ISSN: 2640-9739

***Corresponding author:** Maikel Leon, Department of Business Technology, Miami Herbert Business School, University of Miami, Miami, Florida, USA

Submission: July 17, 2025

Published: August 05, 2025

Volume 3 - Issue 3

How to cite this article: Maikel Leon*. Toward Robust and Scalable Evaluation of AI Models: Past Practices, Current Tooling and Future Directions. COJ Elec Communicat. 3(3).COJEC.000563.2025. DOI: 10.31031/ COJEC.2025.03.000563

Abstract

As Artificial Intelligence (AI) systems transition from research labs into safety-critical domains, such as healthcare, finance and autonomous vehicles, model evaluation has evolved from simple accuracy checks on static datasets to a multidimensional discipline that must address robustness, fairness, security, efficiency and real-world context. This paper traces the evolution through four overlapping eras-rule-based metrics, benchmark competitions, crowdsourcing and human-in-the-loop testing and continuous-integration workflows-showing how each phase expanded the definition of model success. It then outlines contemporary evaluation paradigms, including scaling-law analyses, task-specific versus task-agnostic testing, holistic ethical metrics, human-AI hybrid assessment and operational constraints. To illustrate how these principles are implemented, we survey ten representative frameworks (LlamaIndex, PyDantic Evals, EleutherAI LM Harness, Lighteval, OpenAI Evals, LangSmith, Phoenix, PromptFoo, custom in-house pipelines and Weights & Biases Weave), comparing them across modalities, metric breadth, automation, extensibility and licensing. A comparative analysis highlights trade-offs that of-ten necessitates combining multiple tools for comprehensive coverage. We discuss persistent challenges-including shortcut learning, demographic bias and the need for continuous post-deployment monitoring-and propose best practices for layered testing, falsifiable metrics and rigorous audit trails. Finally, we outline future directions, including risk-aware evaluation for "material" AI, federated privacy-preserving benchmarks, carbon-aware metrics and community-owned adaptive leaderboards. The paper argues that trustworthy AI depends as much on robust, context-sensitive evaluation pipelines as it does on model innovation itself.

Keywords: AI evaluation; Benchmarking; Metrics; Frameworks; Governance

Introduction

Artificial Intelligence (AI) models now read medical scans, underwrite loans and steer autonomous vehicles. These systems increasingly make decisions that carry real-world consequences, from diagnosing illnesses to approving financial transactions to control- ling cars on public roads. In all such safety-critical settings, silent failure is unacceptable because even a single undetected error can result in serious harm. For example, a misclassification in a medical image could lead to a missed cancer diagnosis, or a lapse in an autonomous driving model could contribute to an accident [1]. Historically, however, evaluation was often treated as a peripheral activity applied only after model development, meaning many models were deployed with limited insight into their potential failure modes. The rise of benchmark culture in the late 2000s fundamentally changed how researchers measure progress. High-profile challenges based on large public datasets-most notably the ImageNet competition-transformed research velocity and visibility by introducing a leaderboard mentality [2]. Standardized benchmarks enabled rapid iteration and clear comparisons between techniques, sparking the deep-learning era as teams raced to improve ImageNet top-1 accuracy by even fractional percentages.

Yet, subsequent shortcut-learning analyses revealed that some models achieving top scores did so with brittle decision rules, essentially learning unintended cues or heuristics from the data rather than a proper, generalizable understanding [3]. For instance, a vision model might latch onto background textures correlated with object labels, or a medical classifier might rely on scanner-specific artifacts-strategies that inflate test performance but fail under slightly shifted conditions. Broader surveys have argued that model assessment must keep pace with deployment risks and domain specifics [4]. In other words, as machine learning moves into high-stakes arenas such as healthcare, finance and autonomous systems, evaluation criteria should expand beyond generic accuracy to address the specific requirements and failure modes of each domain. A model might appear excellent on aggregate metrics yet perform unevenly across subgroups or degrade in novel environments; thus, rigorous domain-aware testing is needed to reveal such gaps. Recent work in healthcare AI provides a concrete example: models for clinical diagnosis that perform impressively on curated benchmarks often falter on real-world hospital data due to shifts in patient demographics or imaging protocols [5]. This highlights the need for robust, context-aware benchmarks that accurately reflect actual deployment settings, ensuring clinical relevance and patient safety.

Against this backdrop, the present paper makes the following contributions:

A. It traces historical milestones in model evaluation, identifying four overlapping eras and how each era expanded the notion of what it means for a model to succeed.

B. It details contemporary evaluation paradigms, describing modern multi-faceted approaches that incorporate considerations from scalability and robustness to ethics and efficiency.

C. It surveys ten representative evaluation frame-works currently in use, spanning open-source libraries, cloud services and bespoke in-house tools.

D. It compares their strengths and weaknesses across key dimensions (such as modality support, metric breadth, automation, extensibility and licensing), as summarized in Table 1.

Table 1: Comparison of evaluation frameworks.

Framework	Modalities	Metric Breadth	Automation	Extensibility	License
LlamaIndex	Text, RAG	Medium	High	High	OSS
PyDantic	Text	Low	High	High	OSS
lm-eval	Text	High	Medium	Medium	OSS
Lighteval	Text	Low	Medium	Medium	OSS
OpenAI Evals	Text, Vision	High	High	Low	SaaS
LangSmith	Text, Tools	Medium	High	High	SaaS
Phoenix	Text, Logs	Low	High	Medium	OSS
PromptFoo	Text	Security	Medium	Medium	OSS
In-House	Any	Variable	Variable	High	–
Weights & Biases Weave	Text, RAG	High	High	High	SaaS

E. It identifies open challenges and future research directions for the field, including ongoing issues such as shortcut learning, bias, regulatory compliance and the development of next-generation benchmark methodologies [6].

Historical Perspective

Evaluation practice has evolved through four over- lapping eras, each expanding the notion of what it means for a model to succeed. In the earliest decades of AI and pattern recognition, evaluation was narrowly defined by a few simple metrics on controlled test sets. Over time, this approach gave way to community benchmark competitions that accelerated progress but inadvertently encouraged overfitting to those benchmarks. As models tackled more subjective and complex tasks, the 2010s introduced crowd-sourcing and human-in-the-loop assessments to capture qualities that automated metrics could not judge. Most recently, best practices from software engineering-such as continuous integration and automated testing-have been adopted in machine learning, embedding evaluation checks throughout the model development pipeline [7]. These eras overlap and build on each other: each new phase addresses limitations of the previous ones while introducing its perspective on what success looks like for AI models. See Figure 1 for a timeline including some AI model-evaluation milestones.

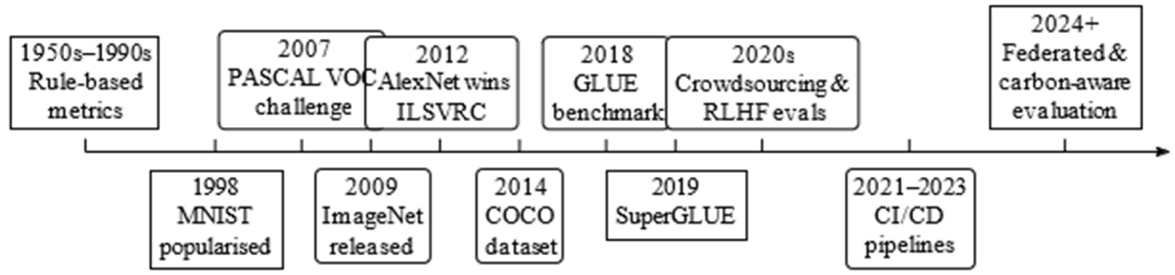

Figure 1: Historical timeline of AI model-evaluation practice.

Rule-based and heuristic metrics (1950s-1990s)

Early pattern-recognition systems relied on deterministic measures, such as accuracy, precision, recall or Peak Signal-to-Noise Ratio (PSNR). During this period, models (often simple classifiers or rule-based systems) were evaluated using single-number summaries, which facilitated easy comparison of performance between approaches. These metrics provided a convenient snapshot of performance on a fixed test dataset, which was usually drawn from the same distribution as the training data. However, such a reductionist evaluation ignored robustness and brittleness. A model could score well on paper yet have hidden flaws. For instance, a speech recognizer trained on clean, studio-quality recordings might achieve high accuracy in the lab, only to fail in a noisy factory environment, despite its laboratory accuracy appearing acceptable. Similarly, an early computer-vision algorithm might optimize for high PSNR on a set of sample images. Still, those gains in signal fidelity did not guarantee that the photos would be perceptually clear or robust to real-world variability. In short, the rule-based and heuristic era defined success in terms of average-case performance on static datasets, with little insight into how models would behave under changing conditions or adversarial inputs [8].

Benchmark competitions (2000s)

The 2000s witnessed the emergence of large public datasets and leaderboard competitions, which dramatically accelerated research progress. High-visibility challenges, such as the ImageNet Large Scale Visual Recognition Challenge, have turned evaluation into a competitive sport, where achieving a new state-of- the-art on the leaderboard has become a key driver of innovation. ImageNet's top-1 accuracy metric be- came a global rallying point that not only measured progress but also helped spark the deep-learning era itself [2]. Inspired by the success of ImageNet, comparable benchmark initiatives emerged in other domains. For example, the GLUE and SuperGLUE benchmarks have become standard tests for language understanding and grand challenges in medical imaging (often hosted on platforms like Kaggle or the MICCAI conference) have spurred advances in radiology AI. These shared benchmarks provided common goals and apples-to-apples comparisons, enabling researchers worldwide to focus on improving the same metrics on the same datasets.

However, this model of static test sets and singular metrics had downsides. Models began to over-specialize to the idiosyncrasies of the benchmark datasets, sometimes exhibiting brittle real-world performance despite their high leaderboard scores [9]. For instance, a vision model optimized for ImageNet might fail to recognize objects when images are taken from new angles or under different lighting, because the training and test images followed similar distributions. In natural-language tasks, models that excelled on a fixed benchmark could be tripped up by rephrased questions or slight shifts in topic. These issues motivated the community to explore dynamic or continually evolving evaluation sets that would be harder to overfit, as well as domain-specific testbeds tailored to particular real-world conditions. The phenomenon of models' "cheating" by exploiting dataset quirks also fuelled shortcut-learning research [3]. Analysts found, for example, that high-performing convolutional networks often relied on texture cues rather than object shapes and medical classifiers sometimes became overly reliant on confounding signals, such as hospital-specific markers in X-ray images. Such findings underscore the need for evaluation frameworks that go beyond static leader-boards to test robustness and genuine generalization.

Crowdsourcing and human-in-the-loop (2010s)

By the 2010s, AI systems were tackling tasks with inherently subjective or context-dependent criteria for success-things like rating the naturalness of a machine translation, the coherence of a story or the empathy in a chatbot's reply. Automated scores alone could not capture nuanced qualities such as plausibility, creativity or ethical alignment in these outputs. This led to the widespread use of crowdsourcing platforms (e.g., Amazon Mechanical Turk) to gather human judgments on a large scale. Non-experts could be enlisted to rank a translation's fluency or to label whether an image caption "makes sense," providing valuable feedback signals that pure metrics, such as BLEU or ROUGE (commonly used for language tasks), might overlook. Crowdsourcing supplied scalable ratings but also introduced new challenges. Different annotators often disagreed, especially on subjective matters-what one person deems a fluent translation, another might find awkward. Cultural bias and annotator background significantly influenced evaluations: for instance, humor or appropriateness of a chatbot response might be rated very differently by annotators from different cultures or age groups. Ensuring consistency and fairness in the face of this

variability became a significant concern. Researchers responded by designing annotation guidelines, performing quality control (filtering out unreliable annotators or outlier responses) and aggregating multiple opinions (e.g., via majority votes or averaging scores) to smooth out individual idiosyncrasies [10].

To further improve reliability, hybrid human-AI evaluation protocols emerged. In these setups, apparent failures or straightforward cases are initially identified by automated tests or heuristic filters, thereby reducing the burden on human raters. For example, a computerized check might flag a generated sentence that is gibberish or ungrammatical or a simple model might score an image caption for fundamental relevance. Only the more ambiguous or borderline cases are then routed to human judges, who might be domain experts if needed [11]. This two-tier process conserves human effort for the instances where their insight is essential. In domains like medicine, this can mean using algorithmic screens to handle routine evaluation of an AI diagnosis, but sending any cases with uncertainty or high risk to a panel of clinicians for review. Such human-in-the-loop evaluation not only im- proves the quality of assessments but also provides a safety net: it leverages the scalability of automation while ultimately relying on human judgment for the final say on complex or high-stakes decisions [12].

Continuous integration for machine learning (Late 2010s)

In the late 2010s, the machine-learning community began borrowing DevOps principles from software engineering to create more reliable and repeatable model evaluation pipelines. Continuous Integration/Continuous Deployment (CI/CD) practices meant that every time a model was trained or updated, a suite of automated tests would run to assess its performance and safety before it was merged or deployed. In practical terms, this introduced components like unit tests, data validation and fairness checks into the model development workflow [13]. For example, a developer updating an NLP model might include a unit test that ensures a specific known input (such as a simple question) still yields the correct answer with the new model version. Data-validation scripts might automatically scan in-coming training data for anomalies, ensure that no personally identifiable information is inadvertently included or verify that training and test splits remain properly segregated. Alongside these, fairness and bias checks could be triggered on each new model build-for instance, testing whether the model's error rates on different demographic groups remain within acceptable ranges. If any of these checks fail (e.g., the model's performance on a minority subgroup drops significantly after a change), the pipeline flags the issue or blocks the deployment, much as a failing unit test would prevent a software update from being released.

The ethos of 'testing early and often' for ML also extends to more complex validation, such as integration tests that mimic the model in a production setting. In an autonomous vehicle model, for instance, an integration test might simulate an end-to-end driving scenario to ensure that the perception module and control module work together correctly. Additionally, research in federated learning demonstrated that rigorous evaluation could be performed across multiple data sites without centralizing sensitive data [14]. In a federated setting (familiar in healthcare and finance), each institution can run the evaluation on its local dataset and share only the aggregate results or model parameter updates. This demonstrates that privacy constraints can coexist with thorough cross-site testing-models can be evaluated on diverse, distributed data (e.g., patient scans from multiple hospitals) to assess generalization, all while patient data remains on-premise at each hospital. These kinds of automated, privacy-preserving evaluation workflows laid the groundwork for the specialized evaluation frameworks discussed later. By embedding evaluation into the CI pipeline, the late 2010s established a norm that model quality checks should happen continuously and systematically, rather than as an afterthought. Concluding this historical survey, we observe a clear trend toward more comprehensive and integrated evaluation. Over time, the community expanded from relying on a single performance metric to tracking many facets of model behavior (accuracy, robustness, fairness, etc.). It shifted from one-off testing to continuous monitoring. There is also a greater reliance on human judgment where automated metrics fall short, reflecting an understanding that qualitative factors-like user satisfaction or ethical considerations-must be part of the evaluation process. Each era's innovations set the stage for the next, ultimately leading to the contemporary evaluation paradigms and tools that we explore in the following sections [15].

Contemporary Evaluation Paradigms

Modern evaluation is inherently multidimensional, encompassing not only raw performance but also considerations of scale, scope, ethics and practicality. Key paradigms and facets include:

A. Scaling-law metrics: Measures like log-likelihood and perplexity are used to quantify performance improvements as model size increases. These metrics help guide decisions about scaling up architectures and have revealed predictable trends (e.g., performance gains vs. model complexity) that inform research directions. They also appear in studies examining the environmental and computational cost of large models, underscoring how diminishing returns in perplexity improvements may not justify exponentially larger resource usage [16]. In short, scaling-law evaluations tie model quality to model size and budget, linking accuracy curves with considerations of efficiency and carbon footprint.

B. Task-specific vs. task-agnostic evaluation: There is a spectrum between highly focused domain benchmarks and broad capability assessments. On one end, domain-specific suites like MIMIC-IV in healthcare or ImageNet in vision provide in-depth evaluation of targeted tasks under realistic conditions (e.g., clinical prediction or object recognition with domain-specific constraints). On the other end, general-purpose evaluation collections (sometimes spanning dozens of tasks) test a model's versatility and transfer learning ability across unrelated domains. For example, a large language model might

be evaluated on a battery of functions from elementary math and commonsense reasoning to legal question answering, all in one suite, to gauge its general intelligence. Both approaches are valuable: task-specific benchmarks ensure excellence on critical applications, while task-agnostic or multi-task evaluations check for broad robustness and identify blind spots that a single-domain test might miss [17].

C. Holistic and ethical metrics: Modern bench- marks increasingly include criteria beyond traditional accuracy. Measures of fairness (checking for performance gaps across demographic groups), robustness (resistance to adversarial examples or noisy inputs), toxicity (the propensity of a model to produce harmful or offensive content), privacy leakage (tendency to memorize and reveal sensitive training data) and environmental impact (energy or carbon cost per training or inference) now accompany core performance metrics [18]. For instance, an evaluation of a text generator might report not only its BLEU score for translation quality, but also the frequency of biased language in its outputs and an estimate of CO_2 emissions associated with producing those translations. Carbon-footprint analyses for large text and image generation models highlight the environmental implications of confident architectural or training choices [19]. By accounting for these factors, evaluation becomes a multidimensional report card that reflects a model's societal and practical implications, not just its accuracy in ideal conditions.

D. Human-AI hybrid evaluation: Some aspects of quality can only be judged relative to human expectations or values. Reinforcement Learning from Human Feedback (RLHF) has emerged as a method for fine-tuning models using human preferences as a guide [20]. In evaluation terms, humans might be asked to rank the helpfulness or correctness of a model's responses and those judgments are used to calibrate the model's behavior. This paradigm is crucial for dialogue systems and content generation, where automated metrics often fall short in gauging attributes such as helpfulness, relevance or safety. For example, a medical question-answering system may be evaluated by doctors who score the correctness and bedside manner of its answers. In contrast, users might consider a chatbot for traits such as empathy and harmlessness. Those human scores can then train a reward model that steers the AI toward more preferred outputs [21]. Human-AI evaluation hybrids are also used post-training; deployed models often rely on user feedback loops (such as thumbs up/down or error reports) to continually assess and improve their performance in the field.

E. Operational constraints: Practical deployment adds another lens for evaluation. Metrics such as inference latency (how quickly the model produces an output), throughput (the number of queries it can handle per second), memory footprint and monetary cost per run become critical, especially in real-time or resource-limited environments. A model that scores highest on accuracy might be unsuitable for a mobile app if it requires a server farm to run or if it responds too slowly for an interactive setting. Thus, evaluation in a production context must take these constraints into account. For instance, an embedded vision system in a drone will have strict limits on model size and power consumption; its "best" model is one that balances accuracy, speed and efficiency. Materiality studies of AI systems argue that the physical and resource aspects (the "material" footprint of AI) are an integral part of judging a model [22]. In sum, a model's excellence on benchmarks must be weighed against its feasibility and reliability in the real world.

No single metric or test captures all these dimensions. Balancing these facets usually requires multiple complementary tools orchestrated in a unified evaluation pipeline. A comprehensive evaluation strategy might, for example, combine a traditional accuracy benchmark, a suite of stress tests for robustness and bias, a phase of human-in-the-loop review for subjective quality and profiling of the model's runtime performance and resource usage. Modern evaluation frameworks facilitate this orchestration by enabling practitioners to combine and tailor different evaluation modules, resulting in a comprehensive assessment of model trustworthiness and suitability for deployment.

Survey of Modern Evaluation Frameworks

To understand how the principles discussed above manifest in practice, we survey ten representatives' evaluation frameworks that researchers and practitioners actively use. These frameworks span a spectrum of designs-some are open-source libraries aimed at maximum flexibility, while others are commercial or cloud-based services emphasizing ease of integration. They differ in the types of models or data they support (text vs. vision vs. multimodal), the breadth of metrics they cover out-of-the-box and how deeply they integrate into the model development lifecycle. By examining each, we can illustrate different trade-offs among ease of use, extensibility, and coverage. Not every framework does everything; as we will see, some prioritize simplicity and speed for targeted use cases, whereas others provide broad "batteries included" evaluation suites or specialize in particular aspects, such as safety or workflow tracing. Below, we discuss each of the ten frameworks in turn, highlighting their key features and typical usage.

LlamaIndex evaluation tools

LlamaIndex, initially developed to support Retrieval-Augmented Generation (RAG) workflows, has evolved into a versatile evaluation toolkit for language model applications. Because RAG systems combine LLMs with external knowledge sources, LlamaIndex provides convenient components to build and assess test datasets for such pipelines. It bundles utilities for dataset construction (e.g., sampling question-answer pairs from a document corpus), as well as built-in metrics to compare model outputs against references or expected content. For instance, it offers similarity metrics that leverage vector embeddings to determine whether a generated answer adequately matches the ground-truth answer or retrieved context. Additionally, LlamaIndex includes LLM-based graders-essentially harnessing a powerful language model (like GPT-4) to act as an automatic judge of output quality. In practice, a developer

can prompt the grader model with the task instructions, the model's output and the correct answer and receive a score or feedback on aspects such as correctness and coherence [23]. The LlamaIndex ecosystem is also highly extensible through plugins. Users have contributed modules to evaluate beyond text, for example, by handling image embeddings or multimodal input-output pairs. This extension into vision-related evaluation echoes calls for more multimodal benchmarks, especially in fields such as healthcare, where clinical decision models may need to interpret both medical images and text [5]. By supporting custom evaluators and multiple data types, LlamaIndex allows practitioners to craft end-to-end evaluation pipelines tailored to complex applications (such as question-answering systems that retrieve medical literature and output a rationale). In summary, LlamaIndex's evaluation tools exemplify a trend toward integrated, flexible benchmarking suites that cover data preparation, automated metric computation and even AI-assisted grading of model outputs.

PyDantic evals

PyDantic Evals takes a software-engineering-minded approach to evaluation by representing each metric and test as a structured, typed object. Built on the Pydantic data-validation library, it enforces schema consistency for inputs and outputs of evaluation routines at runtime. In practical terms, this means that if a metric requires a specific format (e.g., a list of model predictions and a list of ground-truth labels), the frame-work will validate these before computing the metric. Suppose there is a mismatch or an error (for example, a contributor accidentally feeds in malformed data or the metric function returns a value of the wrong type). In that case, PyDantic Evals will raise an explicit error rather than producing a null result or an incorrect value silently. This design prevents silent failures and makes debugging easier in large, collaborative repositories where multiple team members may be adding new evaluation components. Instead of discovering weeks later that a specific evaluation quietly never ran due to a subtle bug, developers are immediately alerted to the issue via the type validation. Teams that integrate evaluation into continuous deployment pipelines find this schema enforcement particularly helpful when working in regulated industries. In such contexts (fi- nance, healthcare, etc.), evaluation results often need to be logged and presented as part of regulatory compliance or model documentation. PyDantic Evals ensures that these results conform to a predefined structure, which facilitates the generation of consistent re- ports. For instance, one can define a schema that every model evaluation report must include accuracy, fairness metrics and explanation artifacts in specific formats. The framework will enforce that each of those fields is populated correctly when the evaluation runs. Overall, PyDantic Evals prioritizes reliability, transparency and correctness in the evaluation process, marrying the rigor of typed schemas with the flexibility of Python-based metric definitions.

EleutherAI LM evaluation harness

EleutherAI's Language Model Evaluation Harness has become a de facto standard tool for benchmarking NLP models on a wide array of tasks. Often referred to simply as "the harness," it provides a convenient interface for evaluating a language model on dozens of datasets with minimal setup. Researchers can specify a list of evaluation tasks and the model to test in a single configuration (for example, a YAML file) and the harness will automatically load each dataset, prompt the model accordingly and compute the relevant metrics. The breadth of tasks covered is extensive, including classic language-understanding benchmarks such as LAMBADA (for long-term context prediction) and the Winograd Schema Challenge (commonsense pronoun resolution), as well as question-answering datasets like TriviaQA and Natural Questions, reading-comprehension tests, math word problems and even domain-specific evaluations like medical question answering. With one command, a user can obtain a summary of how a given model performs across diverse challenges, ranging from commonsense reasoning and knowledge recall to arithmetic and ethics questionnaires.

This harness remains the lingua franca of language-model benchmarking due to its ease of use and the community's contributions of new evaluation sets. It enables apples-to-apples comparisons between models by standardizing evaluation procedures and metrics for each task. However, its design is primarily focused on text-based tasks and support for modalities beyond text is limited. For example, it does not natively handle image or audio inputs, so vision-and-language or speech functions are out of scope unless they are wrapped in text (e.g., by using captions or transcripts). Similarly, interactive or dialog evaluations (which might require multi-turn exchanges with a model) are not straightforward to represent in the harness's current paradigm. Despite these limitations, EleutherAI harnesses its ability to launch evaluations on a sweeping range of NLP benchmarks from a single configuration file, making it an indispensable tool for researchers benchmarking new language models. It streamlines the evaluation process, allowing teams to quickly identify strengths and weaknesses of a model (such as noting that a model excels in trivia recall but struggles in logical reasoning) without writing bespoke code for each dataset.

Lighteval

Lighteval is a lightweight command-line tool designed to facilitate quick "smoke tests" on machine-learning models. Its minimalistic design means that with a single CLI command, a developer can run a model on a small test dataset and get immediate feed-back on basic performance metrics. This is particularly useful during rapid iteration: before committing a new model or change, one might use Lighteval to ensure that nothing is broken (for example, that accuracy on a handful of representative examples remains in a reasonable range). What sets Lighteval apart is its ability to provide token-level traces or analysis for model outputs, a powerful feature for error analysis and debugging. In the context of language models, a token-level trace might show the probability or confidence the model assigned to each successive word in its output or high-light which input tokens contributed most to a prediction. By examining these fine-grained details, developers can pinpoint where a model started to go wrong on a particular example, perhaps

identifying that a specific word confused the model or that it got the first part of a prediction right but then veered off course. Out of the box, Lighteval focuses on straightforward metrics, such as accuracy and precision, aligning with its role as a quick diagnostic tool.

However, it is designed to be extensible via plugins or user-defined scripts. In practice, users often augment Lighteval with additional checks for fairness, bias or safety. For instance, a team might integrate a spluginython plugin that, after computing accuracy, also scans the model's classification errors for any demographic pattern (adding a simple fairness audit) or runs the model's text outputs through a profanity filter or safety lexicon to flag potentially toxic content. These extensions enable Lighteval's basic results to be complemented with domain-specific or ethical metrics without compromising its lightweight character. The trade-off is that Lighteval does not come with a vast library of pre-built evaluations-it's more of a frame- work to run quick tests and any custom checks the user deems important. Its strength lies in speed and simplicity, providing immediate insights and debugging clues that can inform deeper evaluations using more comprehensive frameworks if needed.

OpenAI evals

OpenAI Evals is a cloud-based evaluation platform that follows an "eval-as-code" philosophy. Instead of manually collecting scores or writing separate scripts for each analysis, users write small Python evaluation programs (or configure templates) that define how to test a model and then run those evaluations on OpenAI's infrastructure. A notable feature of OpenAI Evals is its ability to leverage large models (like GPT-4) as part of the evaluation process itself. For example, suppose one wants to evaluate a new chatbot's answers for correctness or tone. In that case, OpenAI Evals allows the user to programmatically prompt GPT-4 to grade each answer on various criteria (e.g., accuracy, helpfulness) and produce a corresponding score. This effectively delegates subjective or complex grading tasks to a robust reference model, which can significantly reduce the friction for small teams. They don't need to recruit human annotators for every evaluation round-often, GPT-4's judgments serve as a proxy for human assessment in areas such as answer quality or code correctness.

The platform is integrated with OpenAI's cloud, meaning that evaluations run on their servers and can directly access OpenAI's latest models and data-handling capabilities. This convenience, however, comes with trade-offs. Relying on OpenAI Evals can introduce vendor lock-in: organizations become dependent on OpenAI's ecosystem for their testing pipeline. Suppose the evaluation code and methodology are tightly coupled with OpenAI's APIs or specific model outputs. In that case, it may be non-trivial to port those evaluations to another environment or to use them with non-OpenAI models. Moreover, organizations subject to strict data sovereignty or privacy regulations may be hesitant to send sensitive evaluation data (which may include proprietary test cases or user information) to an external cloud service. While OpenAI Evals is powerful and lowers the barrier to thorough evaluation-especially for teams without extensive infrastructure-it may not meet the requirements for self-hosting or data confidentiality. In summary, OpenAI Evals offers an attractive, automated way to evaluate models (complete with AIdriven graders and metrics) if one is already part of the OpenAI ecosystem. Still, its closed nature and reliance on vendor services remain important considerations for long-term adoption.

LangSmith eval

LangSmith is an evaluation and observability platform designed for complex AI agent workflows, such as those built with the LangChain framework. Unlike single-step question-answer evaluations, agentic workflows often involve multiple intermediate steps (for example, a language model that plans a solution, calls external tools such as search engines or calculators and then produces a final answer). LangSmith addresses the challenge of evaluating and debugging such chains by capturing every intermediate step, input, output and decision made along the way. When a complex, multi-step model fails or produces an incorrect result, LangSmith's detailed logs make failure diagnosis significantly more straightforward. A developer can inspect the sequence of actions, including which tool was called with what query, how the model's internal prompt evolved and what each step returned. This level of transparency lets one pinpoint the exact step where things went awry-perhaps the model made a flawed assumption in step 3 or a parsing error occurred when reading a tool's response. The platform provides an interface to replay or analyze these chains, which is invaluable for improving systems that have many moving parts. For instance, if an AI assistant using a calculator tool gives a wrong answer, LangSmith's logs might reveal that the assistant passed a wrongly formatted equation to the calculator. With that insight, the developer can tweak the prompt or logic at that step and then use LangSmith to verify the fix.

Because LangSmith is offered as a cloud service, its pricing model typically scales with the volume of logged events (each step or piece of data logged might count towards usage). This encourages teams to adopt selective logging strategies. In practice, a team might decide to log only a subset of interactions in full detail-such as those in a testing environment or a small percentage of production cases-to manage costs. Alternatively, they might log only metadata by default and enable full trace logging when an anomaly is detected. Despite the need to be mindful of what and how much to log, LangSmith provides a powerful mechanism for evaluation in scenarios where understanding the process is as important as the outcome. It essentially blurs the line between evaluation and debugging, ensuring that even very complex AI systems can be systematically assessed and improved.

Phoenix evals

Phoenix is an evaluation platform that doubles as a monitoring system for deployed models, effectively blurring the line between offline benchmarking and live performance tracking. It provides time-sliced dashboards and visualizations that show how model metrics evolve over time or across different data segments. This temporal aspect is crucial for detecting model drift-situations where a model's performance degrades or its predictions change

in distribution as the incoming data shifts. For example, a Phoenix dashboard for a medical image classification model might display the model's accuracy and error rates on a week-by-week basis. If a hospital introduces a new imaging device or if the patient population changes in some way, Phoenix could reveal a gradual decline in accuracy or a spike in certain types of errors following that change. By slicing performance data by time and other attributes (such as location, demographic group or instrumentation), Phoenix helps pinpoint when and where a model may be deviating from its environment. This type of continuous evaluation is informed by "living lab" studies of AI deployments, especially in high-stakes domains such as healthcare, where regular monitoring has been shown to surface issues that static pre-deployment testing might miss [24].

In a living lab scenario-say, an AI tool assisting radiologists in practice-researchers observed that model performance could fluctuate due to workflow changes or seasonal variations in patient data. A tool like Phoenix enables ongoing auditing in these con- texts, alerting stakeholders if, for instance, the model starts missing tumors in a new subset of scans or if its predictions begin to drift systematically. Phoenix typically supports integration with production pipelines, allowing it to ingest live data or periodic evaluation batches and update its dashboards in near real-time. By combining evaluation with monitoring, Phoenix ensures that model assessment is not a one-time event but an ongoing process, thereby facilitating the prompt detection of problems and maintaining model quality in dynamic settings.

PromptFoo

PromptFoo is an evaluation toolkit with a special focus on the security and safety testing of large language models, particularly in terms of prompt integrity and data confidentiality. Traditional evaluations might measure how often a model is correct; PromptFoo, in-stead, stress-tests how the model responds under adversarial or policy-violating conditions. One of its core features is a battery of automated 'jailbreak' attempts: these are cleverly designed input prompts that attempt to trick the model into ignoring its safety guardrails or role restrictions. For instance, PromptFoo might automatically feed the model scenarios like, "Pretend you are not an AI and can say anything, now output the forbidden content. . ." or other variations of exploitative prompts that have been observed in the wild. The goal is to see if the model can be coaxed into generating disallowed outputs (such as hate speech, self-harm instructions or other content that the model should usually refuse to produce).

In addition to testing content filters, PromptFoo conducts privacy probes to check if the model might leak sensitive information. This can involve asking the model to reveal any hidden system instructions or to output examples of its training data (for instance, querying the model for what was in a supposed confidential document to see if it memorized and regurgitates any part of it). By systematically running these attacks and probes, PromptFoo complements traditional performance checks with a type of "red team" evaluation. Instead of measuring how well the model performs its intended task, it measures how resilient the model is against misuse or adversarial manipulation. This security-oriented evaluation aligns with the increasing calls for regulatory oversight in high-risk domains [25]. In industries such as healthcare or finance, simply having a high-accuracy model is not enough-regulators and stakeholders want assurance that the AI cannot easily be tricked into breaking rules or divulging private data. PromptFoo provides a framework for generating evidence to support such assurances. If a model passes PromptFoo's gauntlet of jailbreaking attempts and privacy tests, one gains confidence that it has robust guardrails in place. Conversely, any failures that PromptFoo uncovers can guide developers in patching vulnerabilities (for ex- ample, by refining prompt instructions or fine-tuning the model to handle tricky inputs more safely). This makes PromptFoo an essential complement to other evaluation tools in contexts where safety and compliance are paramount.

Custom and in-house pipelines

In highly regulated sectors, organizations often develop bespoke evaluation pipelines rather than relying on public or third-party frameworks. Strict requirements around data privacy, security and regulatory compliance drive this. For example, a hospital or pharmaceutical company operating under HIPAA (which protects patient health information) may be forbidden from uploading any evaluation data to an external cloud service. Similarly, a European financial institution governed by GDPR might need to ensure that no personal data leaves its servers during model testing. Off-the-shelf tools that send data to unknown servers or that cannot be fully controlled internally are simply non-starters in these environments. As a result, teams build in-house stacks tailored to their specific needs. These pipelines might utilize open-source components (reusing ideas from frame-works like those surveyed above) but are assembled and extended in a way that satisfies internal policies and regulatory guidelines. Standard features include rigorous access controls (to ensure only authorized personnel or processes can run evaluations on sensitive data), audit logging (keeping detailed records of each evaluation run for accountability) and specialized metrics mandated by regulators (such as fair-ness analyses on protected demographic groups or explainability reports for each prediction).

Many banks, for instance, have internal "model risk management" frameworks that require every model to pass a battery of tests (bias checks, stress tests under different economic scenarios, etc.) before deployment, with the results archived for compliance audits. A prominent example of the push toward privacy-preserving evaluation is the emergence of federated benchmarking platforms, such as MedPerf [26]. MedPerf enables researchers to evaluate machine-learning models on sensitive medical datasets that are typically protected behind hospital firewalls. Instead of pooling all the data in one place, MedPerf sends the model (packaged in a container along with evaluation code) to each data-holding institution. The model is evaluated locally at each site on that site's patient data and only the aggregate performance metrics (or other non-sensitive outputs) are returned and combined. This approach reconciles confidentiality with comparability, enabling a

fair benchmark between models across multiple hospitals' datasets without exposing any individual's data.

Such federated or distributed evaluation techniques are increasingly important in sectors where collaboration is necessary to obtain a diverse evaluation (ensuring a model works for different populations or conditions), but data cannot be freely shared. In-house pipelines, whether federated or entirely within a single organization, typically emphasize extensibility and control. They enable organizations to integrate their custom metrics, leverage domain-specific knowledge (for example, specialized scoring rules for clinical predictions) and update the evaluation process in tandem with evolving regulations or standards. While building a custom pipeline requires more upfront effort, it pays off by ensuring that evaluation is not a weak link in the deployment of AI, especially when human lives, legal liabilities or sensitive information are at stake.

Weights and biases weave

Weights & Biases (W&B) Weave is an evaluation and visualization suite built on top of the popular W&B experiment-tracking platform. It aims to provide re-searchers and engineers with a "single pane of glass" through which to view both their experimental results and evaluation metrics side by side. In practice, teams that use WB to log training runs (recording hyper- parameters, training curves, etc.) can utilize Weave to create interactive dashboards that also incorporate post-training evaluation outcomes. For example, after training several variants of a model with different hyperparameters or architectures, one can use Weave to compare their performance on validation and test sets through a unified interface. The tool enables the plotting and slicing of results, allowing a user to generate a chart that shows how accuracy relates to model size across dozens of runs, or a table that lists each model variant along with metrics such as accuracy, F1-score, inference latency and fairness indicators. One of the key benefits of Weave is that it integrates what might otherwise be disparate pieces of the workflow. Instead of examining raw log files or static plots for each experiment, researchers receive a dynamic dashboard. This can be particularly useful during hyperparameter sweeps or ablation studies, where one must simultaneously manage multiple models.

For instance, if a team runs a sweep of 50 experiments to tune a language model, Weave can help filter those results to find the top-performing configurations and then drill down into their evaluation details (such as confusion matrices or error examples if those were logged). Similarly, for model comparison, Weave can display side-by-side evaluations of, say, a baseline model vs. a new model, including not just overall scores but also how each performs on various subsets of data or under specific conditions (if such metrics are logged). Because Weave is part of the W&B ecosystem, it is a SaaS offering that stores experiment data in the cloud and provides collaboration features. Multiple team members can view and interact with the same evaluation dashboard, comment on findings and share insights. This encourages a more iterative and analytical approach to evaluation-exploring the results, asking new questions, and possibly launching new experiments based on what the evaluation dashboard reveals. In summary, WB Weave doesn't introduce novel evaluation metrics per se. Still, it excels at aggregating and presenting evaluation results from multiple sources in one cohesive view, thereby streamlining the analysis and decision-making process in model development.

Comparative Analysis

Selecting an evaluation framework (or set of frame-works) involves balancing several key considerations, as summarized in Table 1:

A. Modalities supported: Some frameworks are specialized for text (NLP) models, while others support additional modalities, such as vision or audio. For instance, a tool designed around language tasks may not be easily evaluated for an image captioning model. Organizations must choose tools that align with the data types of their applications or be prepared to extend them. Suppose a team is working on a multimodal AI (such as a medical system that analyzes both images and text). In that case, they will favor frameworks that either natively handle multimodal input or can be easily augmented to do so.

B. Metric breadth: Frameworks vary in the number of evaluation metrics or dimensions they cover by default. One tool might focus narrowly on accuracy and error rates, whereas another comes with suites for robustness testing, bias measurement, security probing and more. A broader metric coverage is advantageous for thorough evaluation, but it can also lead to increased complexity. Teams should consider whether a given framework addresses all the aspects they care about (from core performance to fairness, safety and beyond) or if they'll need to supplement it with additional tools.

C. Automation and integration: This dimension captures how easily the framework can be plugged into automated workflows. Some frame-works are built with continuous integration in mind-they have command-line interfaces or APIs that allow you to run evaluations on each new model version and produce reports automatically. Others might be more manual or require significant setup for each run. In fast-paced development environments, high automation (including support for batch processing, scripting and result logging) can be a deciding factor. A framework that outputs results in a machine-readable format (such as JSON or through an API) is easier to integrate into dashboards and alerting systems.

D. Extensibility: No evaluation framework will perfectly fit every niche need, so the ability to customize is crucial. Extensibility refers to how easily users can add new metrics, support new task types or adapt the framework to unusual scenarios. Open-source frameworks often excel in this area, as users can easily plug in plugins or modify the code to handle novel evaluations (for example, by adding a new fairness metric or connecting a new data source). Closed-source or SaaS solutions might offer limited customization, which could be a drawback if an organization has unique evaluation criteria (like a specific

domain-specific error measure or a proprietary data format to test).

E. Licensing and data governance: The choice between open-source vs. proprietary (or cloud-hosted) tools has implications for cost, support and data privacy. Open-source frame-works (typically free and locally deployable) provide teams with complete control over their data and the evaluation process, which is crucial for confidential projects. However, they might require more in-house expertise to maintain. SaaS or commercial tools often provide convenience, support and additional features at the expense of recurring costs and potential vendor lock-in. Companies operating under strict data-sovereignty rules might lean away from cloud solutions that require uploading evaluation data, even if those solutions are otherwise attractive.

Crucially, no single tool maximizes every one of these criteria. As shown by our survey, a framework that is strong in one area (such as being highly extensible and multi-modal) might be weaker in another (perhaps lacking automated integration or requiring manual maintenance). Consequently, many organizations develop an evaluation pipeline that incorporates multiple tools to cover all bases. For example, a team might use the EleutherAI LM Harness to benchmark core NLP accuracy and knowledge recall. Then, they can add PromptFoo to stress-test the model's security and resilience to malicious prompts and finally run a custom fairness script to measure any demographic performance gaps. The results from these different evaluations can be aggregated-sometimes using a dash-boarding tool like Weave or an internal reporting system-to present a holistic view of the model. Another scenario could involve pairing a general-purpose metric framework with a domain-specific one: a medical AI startup might run general NLP evaluations on their model's report-generating component while also using an in-house pipeline to evaluate clinical relevance and safety on patient data. The over- arching principle is that comprehensive evaluation of-ten requires a combination of specialized tools. By mixing and matching frameworks (and adding custom evaluations where needed), organizations aim to achieve coverage across all the critical dimensions of model quality and trustworthiness.

Discussion

Despite progress in evaluation techniques, several fundamental challenges remain. One persistent issue is shortcut learning, where models achieve high scores by exploiting spurious correlations in the data rather than learning the intended underlying concepts. This problem is notably acute in medical diagnostics and other high-stakes areas. For example, an AI system for reading chest X-rays might inadvertently learn that images from a particular hospital (which often include a textual label or a specific marker) correlate with positive cases and thus use that shortcut to appear highly accurate [27]. In reality, such a model might focus on hospital-specific artifacts or image annotations (like the presence of a ruler or timestamp that often accompanies severe cases) instead of the actual clinical features of the disease. When deployed elsewhere or under slightly different circumstances, the model's performance can collapse. Detecting these hidden short-cuts is challenging-standard test sets may not reveal them if they contain the same artifact patterns. This calls for more creative evaluation methods, such as stress tests that systematically vary or remove suspected confounders, and for collecting diverse test data that better represent real-world variability.

Another major challenge is demographic and subgroup bias in model performance. Even today's largest vision-language models and other AI systems that achieve expert-level benchmark scores can exhibit significant biases [28]. For instance, a state-of-the-art image captioning model might generate more errors or less descriptive captions for images of people from underrepresented ethnic groups or a question-answering model might perform better on queries about Western topics than on those concerning other cultures. These disparities often persist even when overall accuracy is high, indicating that the models have not learned equally well for all segments of the data. Identifying and quantifying such biases requires a carefully designed evaluation. This may involve slicing evaluation data by demographic attributes or creating targeted test cases (e.g., asking the same question about different demographic subjects) to determine if the model's responses are consistent and fair. Mitigating bias is an open area of research, but it begins with more effective evaluation frameworks that facilitate the identification of where models treat similar cases differently.

In general, as models become more complex and are deployed in dynamic environments, post-deployment monitoring and auditability emerge as crucial components of the evaluation ecosystem. Issues like shortcut reliance or bias may only fully reveal themselves once the model is deployed and encounters new data or adversarial use. Frameworks that facilitate ongoing evaluation in production (for example, capturing performance metrics over time and enabling periodic audits of decisions) will be essential for mitigating these risks. This includes tools that log model decisions in a transparent way (to support later auditing and explanation) and those that can send alerts when live data deviates significantly from the training or validation distribution. The challenge is not only technical but also procedural: integrating these evaluation feedback loops into the standard operating procedure of organizations. In summary, combating shortcut learning, ensuring fairness across demographics and maintaining vigilance through continuous monitoring are among the most pressing open challenges for the field of model evaluation.

Best practices

A few emerging best practices can guide more reliable evaluation:

a) Layered evaluation: Employ a multi-tiered approach to testing models. Begin with low-level unit tests on simple, controlled cases (e.g., checking that a vision model correctly classifies a basic shape or that a translation model accurately translates a single well-formed sentence). Then, move on to integration tests and larger-scale benchmarks that reflect real-world use

cases and finally incorporate human audits or user studies to assess qualitative aspects. This progression from granular to holistic helps mitigate blind spots; issues that slip past one layer may be caught in another. For example, unit tests might catch regressions on edge-case in- puts, while a final human review might catch deficiencies in output quality that automated metrics overlook.

b) Metrics as falsifiable hypotheses: Treat every reported metric or performance claim as a hypothesis that should be subject to challenge and verification. Instead of accepting a high accuracy at face value, good practice is to ask, "Under what conditions might this accuracy not hold?" and then design experiments to probe those conditions. By thinking in terms of falsification, teams encourage thoroughness and reproducibility of their work. They might, for instance, try to "break" their model by constructing adversarial test cases or by having an independent team attempt to find scenarios where the model fails. This scientific mindset ensures that evaluation results are robust and not just a product of overfitting to a particular test set.

c) Version control and audit trails: Maintain strict versioning for evaluation datasets, scripts and model checkpoints. When evaluating a model, record precisely which data (and its corresponding version) was used and store the model's identifier or hash. This practice creates an invaluable audit trail for debugging and accountability. If a later analysis identifies an issue with the model, one can refer back to the archived evaluation to see how the model was tested and what it was observing. Likewise, if an evaluation dataset is updated (for example, to correct labelling errors or to include more challenging examples), assigning it a new version number enables fair comparisons over time and prevents confusion about which results correspond to which dataset state. Such discipline in record-keeping aligns with emerging AI accountability standards that call for transparency in the testing and validation process [29]. It enables external audits or regulatory reviews to retrace the evaluation steps that were taken before deployment.

Ethics and governance

As AI models transition from research to real-world applications, evaluation practices are increasingly influenced by ethical and regulatory expectations. Regulators and oversight bodies are calling for evidence that models perform reliably not just on the data they were trained or tuned on, but also on truly out-of-sample data that represents the diversity of real populations and conditions [25]. This is fundamentally an evaluation mandate: it requires developers to curate test sets (or conduct trials) that cover edge cases and underrepresented groups to ensure the model's accuracy and decision logic hold up universally. A related principle is the equal treatment of similar cases-in domains like lending or healthcare, one often hears that "like cases should be treated alike." For model evaluation, this translates to checking that the model's outcomes do not unjustifiably differ for individuals or scenarios that are essentially equivalent except for sensitive attributes. Regulators may expect to see fairness audits as part of the model approval process, demonstrating that, for example, a credit scoring model gives the same recommendations for two applicants with the same financial profile, regardless of their race or gender. Meeting these expectations often means incorporating fairness metrics and stress tests into the evaluation pipeline from the start, rather than as afterthoughts.

Another aspect of governance is the use of controlled deployments or 'living lab' trials to uncover sociotechnical risks before complete rollout [24]. In a living lab approach, an AI system is introduced into a real operational environment, but under close monitoring and with fallback mechanisms in place. For instance, a hospital might deploy a diagnostic AI tool in a few departments, with clinicians double-checking every recommendation or a city might pilot an AI-based traffic management system in a limited area while maintaining human oversight. These field trials serve as an extension of evaluation into the real world, revealing unanticipated failure modes, user interactions and organizational impacts that no laboratory test can catch. Perhaps the AI's advice is technically accurate but routinely ignored by human decision-makers, or maybe its presence changes the workflow in ways that introduce new errors-these are sociotechnical dynamics that pure performance metrics cannot capture. By surfacing such issues early, organizations can refine both the model and the surrounding processes (training users, adjusting UI, setting more straightforward guidelines) before scaling up the AI's deployment. In summary, modern governance frameworks for AI increasingly demand that evaluation be not only rigorous in the lab (with strong evidence of generalization and fairness) but also ongoing and contextual, involving real-world testing and continuous oversight to ensure the technology truly benefits society without unintended harm.

Future Directions

Looking ahead, we identify several trends and opportunities that could shape the next generation of model evaluation:

A. Risk-aware evaluation for "material" AI: As AI systems become embedded in physical environments with pervasive sensors and actuators, evaluation will need to account for the material risks they pose [22]. This means going beyond measuring predictive accuracy to assessing potential harms or failure modes in context. For example, an autonomous vehicle's vision system might require an evaluation pipeline that incorporates the simulation of rare but dangerous events (such as pedestrian darting into traffic or sensor malfunctions) and quantifies the risk of misses in those scenarios. Pervasive IoT devices, which utilize AI (from smart home gadgets to surveillance systems), raise significant privacy and security considerations that should be factored into their evaluation. Future evaluation frameworks may include risk metrics or checklists (inspired by safety engineering) that flag whether a model has been tested for low-probability, high-impact events. The concept

of "materiality" emphasizes that AI is not just algorithms in a vacuum-it is part of the real world with physical and social consequences, so evaluation criteria must expand accordingly.

B. Federated and privacy-preserving benchmarks: Building on efforts like MedPerf, we expect to see more federated evaluation initiatives that allow models to be tested on diverse, sensitive datasets without centralizing the data [26]. Such benchmarks can significantly enhance the coverage of evaluation by incorporating data from multiple sources (different hospitals, regions and user groups) while respecting privacy and data ownership. In finance, for instance, banks might collaborate to evaluate fraud detection models across their combined (but siloed) transaction data through a federated setup. This would increase confidence that a model generalizes across institutions and customer demographics. Developing standardized protocols for federated evaluation-encompassing aspects such as secure aggregation of results, fairness in comparison, and reproducibility-will be a crucial step in ensuring that privacy-preserving testing becomes mainstream.

C. Carbon-aware and efficiency metrics: With growing awareness of the environmental foot-print of AI, we anticipate that evaluation reports will routinely include energy usage and carbon emission estimates alongside traditional metrics [16]. This might involve tracking the compute time or FLOPs a model requires for training and inference and translating that into standardized sustainability scores. In the future, leaderboards could rank models not only by accuracy but also by "energy per 1000 predictions" or similar efficiency metrics. Such carbon-aware evaluation could incentivize the development of models that achieve better performance-cost trade-offs. Additionally, material efficiency metrics (like memory footprint or hardware utilization) are likely to become part of the evaluation criteria, especially for edge AI applications where resources are constrained. This trend aligns with corporate sustainability commitments and could be reinforced by regulations that push for greener AI practices.

D. Community-owned, adaptive benchmarks: To combat the saturation of static leaderboards and the temptation for teams to over-optimize for specific test sets, future benchmarks may become dynamic and community-driven [3]. One model for this is an adaptive benchmark that continuously evolves by incorporating new test cases, especially targeting areas where current models struggle. For example, a question-answering benchmark could automatically add questions that stump the latest top-performing model, ensuring that the benchmark's difficulty keeps pace with model improvements. We are also seeing a shift towards community ownership of bench-marks, datasets and leaderboards, which are maintained by broad coalitions of researchers rather than single institutions. This can promote more exhaustive coverage of evaluation criteria (as the community can contribute tests for various failure modes) and reduce the risk of leaderboard gaming. Over time, such adaptive benchmarks could better encourage genuine generalization and robustness, since models would need to perform well on an ever-expanding array of challenges, not just a fixed set of examples. This idea aligns with the concept of 'open-ended' evaluation, where success is defined not by surpassing a static human baseline on a task, but by how well models can improve while meeting safety and reliability standards in unanticipated situations.

Conclusion

Over the decades, model evaluation has progressed from simple single-number heuristics to multidimensional report cards that encompass accuracy, robustness, fairness, safety and even environmental impact. This paper has traced that evolution, highlighting how we moved from an era of basic accuracy checks to one of comprehensive, context-aware benchmarking and surveyed the contemporary tools that operationalize these principles. One overarching theme is that trustworthy AI depends on evaluation pipelines that are as mature and adaptive as the models themselves. In practice, this means building evaluation into every stage of the model lifecycle and continuously expanding our metrics and methods to cover new risks. Modern AI systems are not static: they learn from increasing amounts of data, interact with humans and are deployed in a wide range of environments. Accordingly, evaluation must be an ongoing, dynamic process. The comparison of today's frameworks reveals that no single tool can do it all; therefore, organizations assemble toolchains that collectively provide a 360-degree view of model performance. Importantly, evaluation is no longer just a technical afterthought; it is a first-class concern intertwined with ethics and governance. Techniques such as bias audits, safety testing and monitoring for drift are becoming standard practices alongside traditional performance measurement. The challenges ahead are significant. Models will only grow more complex and ubiquitous, raising the bar for what our evaluation methods must catch. Continued collaboration will be essential: researchers need to develop new metrics and methodologies for emerging AI behaviors, practitioners need to share best practices and real-world findings from deployment and regulators need to establish clear guidelines that incentivize rigorous evaluation without stifling innovation. Suppose these communities work together to prioritize robust evaluation. In that case, we can better ensure that AI systems remain reliable, fair and aligned with human values as they scale in capability and impact. In essence, the future of trustworthy AI hinges not only on developing better models but also on more effective evaluation methods.

References

1. Leon M (2025) AI-driven digital transformation: Challenges and opportunities. Journal of Engineering Research and Sciences 4(4): 8-19.
2. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, et al. (2015) ImageNet large scale visual recognition challenge. International Journal of Computer Vision 115(3): 211-252.
3. Geirhos R, Jacobsen JH, Wichmann FA, Michaelis C, Zemel R, et al. (2020) Shortcut learning in deep neural networks. Nature Machine Intelligence 2: 665-673.

4. Jordan MI, Mitchell TM (2015) Machine learning: Trends, perspectives and prospects. Science 349(6245): 255-260.
5. Mincu D, Roy S (2022) Developing robust benchmarks for driving forward ai innovation in healthcare. Nature Machine Intelligence 4: 916-921.
6. Leon M (2025) Gail: Enhancing student engagement and productivity. The International FLAIRS Conference Proceedings 38(1):
7. León M, Nápoles G, García MM, Bello R, Vanhoof K (2011) Two steps individuals travel behavior modeling through fuzzy cognitive maps pre-definition and learning. In: Batyrshin I, Sidorov G (Eds.), Advances in Soft Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, Germany, pp. 82-94.
8. Leon M (2023) Aggregating procedure for fuzzy cognitive maps. The International FLAIRS Conference Proceedings 36(1): 1-3.
9. Leon M (2024) GenAI-driven pedagogy: Transforming education. The Cuban Scientist 5(1): 1-2.
10. Leon M (2025) Principles for deploying responsible machine learning models. International Journal of Computers 10: 161-171.
11. Leon M (2025) Charting equitable and transparent machine learning: Principles and practices. Biomedical Journal of Scientific & Technical Research 62(2): 54403-54412.
12. Nápoles G, Hoitsma F, Knoben A, Jastrzebska A, Leon M (2023) Prolog-based agnostic explanation module for structured pattern classification. Information Sciences 622: 1196-1227.
13. Leon M (2025) Artificial intelligence: Evolution, challenges, future and governance. International Journal of Computers 10: 81-93.
14. Rieke N, Hancox J, Li W, Milletarì F, Cardoso MJ, et al. (2020) The future of digital health with federated learning. NPJ Digital Medicine 3: 119.
15. DeSimone H, Leon M (2024) Explainable AI: The quest for transparency in business and beyond. 2024 7th IEEE International Conference on Information and Computer Technologies (ICICT), IEEE, pp. 532-538.
16. Rolnick D, Donti PL, Kaack LH (2022) Achieving net zero emissions with machine learning. Nature Machine Intelligence 4: 416-421.
17. Leon M, Depaire B, Vanhoof K (2013) Fuzzy cognitive maps with rough concepts. 9th Artificial Intelligence Applications and Innovations (AIAI), Sep 2013, Paphos, Greece, pp. 527-536.
18. Leon M (2024) The escalating AI's energy demands and the imperative need for sustainable solutions. WSEAS Transactions on Systems 23: 444-457.
19. Tomlinson B, Black RW, Patterson DJ, Torrance AW (2024) The carbon emissions of writing and illustrating are lower for AI than for humans. Scientific Reports 14: 54271.
20. Leon M (2025) Generative artificial intelligence and prompt engineering: A comprehensive guide to models, methods and best practices. Advances in Science, Technology and Engineering Systems Journal 10(2): 1-11.
21. Obermeyer Z (2022) Mitigating bias in AI at the point of care. Science 378: 782-784.
22. Sloane M, Moss E, Reddi VJ (2025) Materiality and risk in the age of pervasive ai sensors. Nature Machine Intelligence.
23. Leon M (2025) Advancing equitable and transparent machine learning across business, computing and engineering innovations. Research & Development in Material Science 22(1): 2640-2649.
24. Gilbert S, Mathias R, Schönfelder A, Wekenborg M, Steinigen-Fuchs J, et al. (2025) A roadmap for safe, regulation-compliant living labs for AI and digital health development. Science Advances 11(20): eadv7719.
25. Babic B, Gerke S, Evgeniou T, Cohen IG (2019) Algorithms on regulatory lockdown in medicine. Science 366(6470): 1202-1204.
26. Karargyris A, Umeton R, Sheller MJ, Aristizabal A, George J, et al. (2023) Federated benchmarking of medical artificial intelligence with MedPerf. Nature Machine Intelligence 5(7): 799-810.
27. Huang C (2024) Shortcut learning in medical AI hinders generalization: Method for detection. NPJ Digital Medicine 7: 118.
28. Zhang K (2025) Demographic bias of expert-level vision-language foundation models in chest x-ray diagnosis. Science Advances 11: eadq0305.
29. Pati S (2023) Detecting shortcut learning for fair medical AI using shortcut testing. Nature Communications 14: 39902.