Auto-BenchmarkCard: Automated Synthesis of Benchmark Documentation

Aris Hofmann[1,2], **Inge Vejsbjerg**[2], **Dhaval Salwala**[2], **Elizabeth M. Daly**[2]

[1]IBM, Boeblingen, Germany,
[2]IBM Research, Dublin, Ireland
aris.hofmann@ibm.com, ingevejs@ie.ibm.com, dhaval.vinodbhai.salwala@ibm.com, elizabeth.daly@ie.ibm.com

Abstract

We present Auto-BenchmarkCard, a workflow for generating validated descriptions of AI benchmarks. Benchmark documentation is often incomplete or inconsistent, making it difficult to interpret and compare benchmarks across tasks or domains. Auto-BenchmarkCard addresses this gap by combining multi-agent data extraction from heterogeneous sources (e.g., Hugging Face, Unitxt, academic papers) with LLM-driven synthesis. A validation phase evaluates factual accuracy through atomic entailment scoring using the FactReasoner tool. This workflow has the potential to promote transparency, comparability, and reusability in AI benchmark reporting, enabling researchers and practitioners to better navigate and evaluate benchmark choices.

Introduction

Benchmarks are vital in AI for standardizing tasks, enabling model evaluation, tracking progress, and setting baseline expectations (Reuel et al. 2024). Appropriate benchmarks systematically detect, assess, and mitigate risks (Sokol et al. 2024). Unsuitable benchmarks may risk leaving failure modes undetected, leading to deployment with unverified, poorly understood behaviors. Choosing appropriate benchmarks ensures models are evaluated on relevant tasks, avoiding inaccurate assessments and missed risks.

Benchmark documentation is often limited, requiring developers to parse source code or consult reference papers to understand the details of the benchmark. Recently, (Sokol et al. 2024) proposed a standardized representation of benchmark metadata, drawing inspiration from existing standards such as model cards (Mitchell et al. 2019) and dataset documentation frameworks like Croissant (Akhtar et al. 2024). Sokol's framework defines key benchmark metadata such as "purpose," "methodology," and "risks" in order to support informed selection, clearer stakeholder communication, and better understanding of objectives and limitations. However, creating benchmark cards by hand is time- and labor-intensive, posing a significant barrier to widespread adoption across the community. A large-scale analysis of Model Cards shows frequent completion of "Model Description" and "Training Procedure," but critical fields—evaluation, limitations, and risks—are often left blank (Huggingface 2025). Regarding risks, (Rao et al. 2025) found that only 14% of AI model cards mention risks, and 96% of those were identical. This imbalance highlights the practical challenges of achieving comprehensive documentation through manual effort alone. To mitigate this, we introduce an automated workflow aimed at generating BenchmarkCards. The system employs a multi-agent architecture to extract relevant information from heterogeneous sources, including Unitxt, Hugging Face repositories, and associated academic publications. Content is structured into a BenchmarkCard based on Sokol's schema (Sokol et al. 2024) and validated for factual consistency against source data.

System Overview

The workflow has three phases: *Extraction*, *Composition*, and *Validation*, illustrated in Figure 1. Access is provided via a Python CLI, and the system[1] is available in open source.

Extraction Phase: This phase collects structured benchmark data from multiple sources using modular custom agent tools. The implementation currently supports Unitxt but can be adapted for other standards like lm-eval-harness. Users begin by specifying a benchmark identifier for the *Unitxt Tool*, built upon the Unitxt library (Bandel et al. 2024), which searches its catalog and retrieves the corresponding *UnitxtCard*. The retrieved card is then parsed to identify cited materials (e.g., metrics, templates) and retrieve related supplementary cards, with the result returned in JSON format. Next, the *Extractor Tool* extracts identifiers from the JSON such as the Hugging Face repository ID and the publication URL for subsequent processing. The *Hugging Face Tool* then extracts metadata from the benchmark's repository. Finally, the *Docling Tool* (Livathinos et al. 2025) processes the benchmark's associated research publication, converting it into machine-readable markdown format.

Composition Phase: The extracted data is passed to a large language model (LLM), which generates a complete BenchmarkCard by filling predefined sections such as purpose, methodology, and limitations. Once the initial card is generated, the system passes it to the *Risk Atlas Nexus* framework (Bagehorn et al. 2025). The risk identifier component flags potential risks based on a structured risk taxon-

[1]https://github.com/IBM/ai-atlas-nexus-demos/tree/main/auto-benchmarkcard

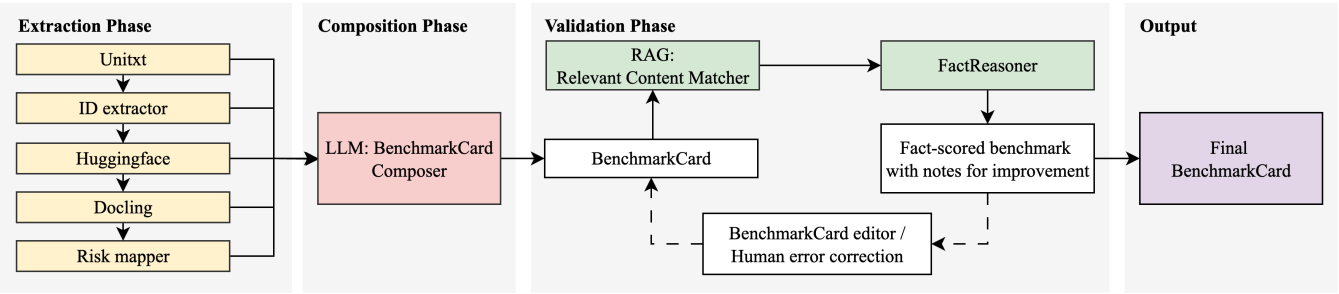

Figure 1: Auto-BenchmarkCard workflow overview

omy. These risks are incorporated into the BenchmarkCard, adding a governance-informed layer to the technical content.

Validation Phase: Validation plays a critical role by introducing a structured approach to verifying the factual accuracy of the initial BenchmarkCard. To address consistency challenges, especially conflicting details, we use *FactReasoner*, a probabilistic framework for assessing factual consistency via natural language inference (Marinescu et al. 2025). We extend the package with custom components for atomization and context retrieval, specifically adapted to the BenchmarkCard format. The validation process begins by breaking down the BenchmarkCard into atomic statements: small, self-contained units of meaning that can each be checked for accuracy. This step is performed using an LLM with a prompt tailored to the structure and content of BenchmarkCards. In contrast to generic atomization, this approach ensures that the resulting statements are not only minimal but also and explicitly designed to be fact-checkable.

To assess each statement's validity, we reference content from the Extraction Phase. The extracted metadata can be extensive, so evaluation by comparing each atomic statement against the full knowledge base is inefficient. To solve this, we index all extracted metadata used to compose the BenchmarkCard in a vector database. The retrieval module combines sparse keyword search and dense vector similarity to retrieve the relevant evidence for each atomic statement from the indexed knowledge documents. Retrieved chunks are graded and re-ranked by an LLM based on their relevance to the atomic statement. The paired atomic statements and their corresponding retrieved evidence are then passed to *FactReasoner*, which assigns an entailment score between 0 and 1. A score close to 1 indicates that the statement is strongly supported by the source material (i.e., factually correct), a score near 0 indicates contradiction, and a score around 0.5 reflects a neutral or unverifiable claim. Atomic statements with low entailment scores are flagged for potential correction. Two remediation strategies are available: Automated Revision and Human-in-the-Loop Correction. *Automated Revision:* Flagged sections are passed to an LLM along with their relevant context, allowing for targeted regeneration of specific BenchmarkCard fields with increased accuracy compared to the initial generation step, which relied on the full, unfiltered output of the Extraction Phase. *Human-in-the-Loop Correction:* Flagged fields are routed for manual review and correction by human annotators.

This process results in a BenchmarkCard that has been automatically generated and validated for factual correctness, with the final card and workflow inputs/outputs provided as JSON.

Limitations

The workflow has several limitations. First, performance is constrained by the completeness and quality of the extracted input. If insufficient data is retrieved from sources such as Hugging Face, the LLM may struggle to generate an accurate BenchmarkCard due to gaps in documentation. Second, factual correctness does not guarantee comprehensiveness or practical relevance. While the validation process ensures that each statement is grounded in evidence, it does not assess whether the content sufficiently covers all important aspects of the benchmark. Comprehensiveness refers to the extent to which the generated output addresses all relevant information, ensuring that no critical detail is omitted. In contrast, factuality only measures whether a given claim is accurate based on the available evidence, without introducing hallucinated content. This distinction becomes particularly important in cases where the generated content, though factually accurate, overlooks more central or representative information. For example, given a context stating "The main languages of the benchmark are English and Spanish, but some questions also address Portuguese," a generated field listing only "Portuguese" as the benchmark language would pass a factuality check but fail in terms of comprehensiveness. It highlights a key limitation: accurate statements may still be misleading if they omit relevant information. To address this, one direction for future work would be to implement a dedicated evaluation step for comprehensiveness.

Outlook

Our workflow produces a BenchmarkCard that is based on information extracted from multiple sources. Its factual alignment with the underlying data is assessed, and fields with low alignment scores are flagged for human intervention or revised by an LLM. While the workflow is specifically designed for BenchmarkCard generation, its modular architecture is applicable to similar tasks involving structured data extraction, LLM-based generation, and factuality validation. As such, it can be adapted to a wide range of use cases in automated documentation, summarization, and metadata synthesis across AI governance and other domains.

References

Akhtar, M.; Benjelloun, O.; Conforti, C.; Foschini, L.; Giner-Miguelez, J.; Gijsbers, P.; Goswami, S.; Jain, N.; Karamousadakis, M.; Kuchnik, M.; et al. 2024. Croissant: A metadata format for ml-ready datasets. *Advances in Neural Information Processing Systems*, 37: 82133–82148.

Bagehorn, F.; et al. 2025. AI Risk Atlas: Taxonomy and Tooling for Navigating AI Risks and Resources. *arXiv preprint arXiv:2503.05780*.

Bandel, E.; et al. 2024. Unitxt: Flexible, Shareable and Reusable Data Preparation and Evaluation for Generative AI. *arXiv preprint arXiv:2401.14019*.

Huggingface. 2025. What's the AI Community Being Transparent About? https://docs.google.com/presentation/d/1PPSpl-RIcqyZrhahXyqzhJxBXyiAQRjORNcde2RI9rY. Accessed: 2025-07-24.

Livathinos, N.; Auer, C.; Lysak, M.; Nassar, A.; Dolfi, M.; Vagenas, P.; Ramis, C. B.; Omenetti, M.; Dinkla, K.; Kim, Y.; Gupta, S.; de Lima, R. T.; Weber, V.; Morin, L.; Meijer, I.; Kuropiatnyk, V.; and Staar, P. W. J. 2025. Docling: An Efficient Open-Source Toolkit for AI-driven Document Conversion. technical report arXiv:2501.17887, arXiv.

Marinescu, R.; et al. 2025. FactReasoner: A Probabilistic Approach to Long-Form Factuality Assessment for Large Language Models. *arXiv preprint arXiv:2502.18573*.

Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I. D.; and Gebru, T. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, 220–229.

Rao, P. S.; Šćepanović, S.; Zhou, K.; Bogucka, E. P.; and Quercia, D. 2025. RiskRAG: A Data-Driven Solution for Improved AI Model Risk Reporting. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 1–26.

Reuel, A.; Hardy, A. F.; Smith, C.; Lamparth, M.; Hardy, M.; and Kochenderfer, M. J. 2024. BetterBench: Assessing AI Benchmarks, Uncovering Issues, and Establishing Best Practices. *CoRR*, abs/2411.12990.

Sokol, A.; Moniz, N.; Daly, E.; Hind, M.; and Chawla, N. 2024. BenchmarkCards: Large Language Model and Risk Reporting. *arXiv preprint arXiv:2410.12974*.