HALLMARK: A Benchmark for Citation Hallucination Detection

Anonymous Author(s)

Abstract

Citation hallucination—where language models generate plausible but fabricated references—poses a growing threat to scientific integrity. The NeurIPS 2025 incident, in which 53 accepted papers contained fabricated citations, and subsequent large-scale audits finding hundreds of affected publications across major venues, underscore the urgency of automated verification. Yet no standardized benchmark exists to compare citation verification tools, and no tool is designed for venue-scale deployment. We address both gaps. First, we introduce HALLMARK (**Hall**ucination bench**mark**), a benchmark comprising 1,182 BibTeX entries spanning 13 hallucination types organized into three difficulty tiers. Each entry undergoes six binary sub-tests inspired by HumanEval's multi-criteria evaluation, enabling fine-grained diagnostic analysis beyond binary detection. HALLMARK provides tier-weighted metrics that reward detection of hard hallucinations, expected calibration error for confidence assessment, and a Plackett-Luce ranking system that handles incomplete evaluations. Second, we present `bibtex-updater`, a practical citation verification tool built for integration into venue workflows—CI/CD pipelines, pre-commit hooks, and batch processing of submissions—rather than for optimizing benchmark scores. Evaluated on HALLMARK alongside existing tools, `bibtex-updater` achieves 0.96 detection rate and 0.90 F1, with the lowest calibration error, but reveals systematic blind spots on venue-level hallucinations that no current tool addresses. Both the benchmark and the tool are open-source, pip-installable, and designed for community adoption. Code and data: `https://github.com/anonymous/hallmark`.

1 Introduction

Citation hallucination—the generation of plausible but fabricated bibliographic references by language models—has emerged as a concrete threat to scientific publishing. In late 2025, the NeurIPS program committee identified 53 accepted papers containing fabricated citations that had passed peer review [NeurIPS Program Committee, 2025]. Independent audits confirmed the scale of the problem: Chen et al. [2026] analyzed 2.2 million citations across 56,000 papers and flagged 604 papers with likely hallucinated references; Ansari and Li [2026] found approximately 300 affected papers in ACL, NAACL, and EMNLP proceedings alone; and Wei et al. [2026] documented a sharp rise from zero detected cases in 2021 to consistent problems across HPC venues by 2025. These findings reveal that citation hallucination is not anecdotal but systematic, and that neither authors nor reviewers reliably catch fabricated references.

Several citation verification tools have been developed in response, ranging from DOI resolution checkers to multi-database cross-referencing systems. However, two gaps remain. First, *no standardized benchmark exists to compare these tools*: each is evaluated on ad-hoc datasets under different conditions, making it impossible to assess relative strengths, failure modes, or coverage gaps. Second, *no existing tool is designed for venue-scale deployment*: current tools are built for individual researchers, not for program committees processing hundreds of submissions under time constraints.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

Table 1: Comparison of HALLMARK with related citation analysis efforts. HALLMARK is the first *benchmark* for citation verification tool evaluation, whereas prior work focuses on auditing or detection.

	Entries	Types	Sub-tests	Tool eval.	Open
GhostCite [Chen et al., 2026]	56K papers	—	✗	✗	✗
HalluCitation [Ansari and Li, 2026]	300 papers	3	✗	✗	✓
Mysterious Citations [Wei et al., 2026]	HPC venues	—	✗	✗	✗
HALLMARK (ours)	1,182	13	6	✓	✓

We address both gaps with five contributions:

1. **A hallucination taxonomy** of 13 types organized into three difficulty tiers (Easy, Medium, Hard), capturing the full spectrum from obviously fake DOIs to subtly plausible fabrications (section 3.1).
2. **A benchmark dataset** of 1,182 BibTeX entries (982 public: 582 dev + 400 test) with six binary sub-tests per entry, enabling multi-criteria evaluation inspired by HumanEval [Chen et al., 2021] (section 3.2).
3. **An evaluation protocol** with tier-weighted F1 that rewards detecting hard hallucinations, expected calibration error (ECE) for confidence assessment, and Plackett-Luce ranking [Luce, 1959, Plackett, 1975] for comparing tools with incomplete coverage (section 4).
4. **A practical verification tool** (`bibtex-updater`): an open-source, pip-installable citation checker with a multi-source verification pipeline, designed for integration into venue CI/CD workflows, pre-commit hooks, and batch submission processing (section 5).
5. **Open infrastructure**: a pip-installable Python package with a baseline registry, CI-integrated evaluation pipeline, and a community contribution system inspired by ONEBench [Ruan et al., 2024] (section 6).

Our evaluation shows that `bibtex-updater` achieves 0.96 detection rate and 0.90 F1 overall, with the lowest calibration error (ECE = 0.04) among all tools. Its performance drops on medium-tier types requiring venue-level metadata verification—a gap shared by all evaluated tools. No single tool covers all 13 hallucination types, and calibration varies widely (ECE from 0.04 to 0.36). These results demonstrate the value of a structured benchmark for identifying actionable improvement targets, and the feasibility of venue-scale automated verification with current tools.

2 Related work

Citation hallucination in the LLM era. Large language models routinely fabricate bibliographic references when generating academic text [Alkaissi and McFarlane, 2023, Agrawal et al., 2024]. The scale became apparent through systematic audits: Chen et al. [2026] developed the CiteVerifier framework and analyzed 2.2 million citations across 56,000 papers, identifying 604 with likely hallucinated references and revealing a "verification gap" where both authors and reviewers fail to check citations adequately. Ansari and Li [2026] focused on ACL/NAACL/EMNLP and found a rapid increase in 2025, with over 100 affected main-conference papers. Wei et al. [2026] observed a sharp rise from zero cases in 2021 to consistent problems across HPC venues by 2025. Commercial tools like GPTZero's hallucination detector [GPTZero, 2025] and academic projects like HaRC [HaRC Contributors, 2024] and verify-citations [verify-citations Contributors, 2025] address detection, but each targets different hallucination types and uses different evaluation protocols, making comparison impossible without a shared benchmark. Critically, none of these tools are designed for venue-scale deployment: they lack CI/CD integration, batch processing, and the throughput needed to screen hundreds of submissions. `bibtex-updater` [Reizinger, 2025] addresses this gap with a multi-source verification pipeline designed for integration into publication workflows (section 5).

Hallucination detection benchmarks. General-purpose hallucination benchmarks exist for LLM outputs [Hu et al., 2024, Li et al., 2023] and long-form generation [Ravichander et al., 2024], but these focus on factual claims rather than bibliographic metadata. Citation verification presents unique challenges: entries have structured fields (DOI, authors, title, venue, year) that can be independently verified against external databases, hallucinations range from syntactic (malformed DOI) to semantic

Table 2: Design principles adopted from established benchmarks.

Principle	Source	HALLMARK implementation
Multi-criteria evaluation	HumanEval	6 sub-tests per entry
Temporal segmentation	SWE-bench	3 time segments, contamination detection
Continuous updates	LiveCodeBench	Ever-expanding entry pool
Incomplete-data ranking	ONEBench	Plackett-Luce model

(plausible but nonexistent paper), and ground truth requires cross-referencing multiple bibliographic APIs. No existing benchmark captures these properties.

Benchmark design principles. HALLMARK synthesizes design principles from four influential benchmarks. From HumanEval [Chen et al., 2021], we adopt multi-criteria sub-tests: each entry is evaluated on six independent checks rather than a single binary label, enabling fine-grained failure analysis. From SWE-bench [Jimenez et al., 2024], we incorporate temporal segmentation to detect and measure contamination effects. From LiveCodeBench [Jain et al., 2024], we design for continuous updates—new entries can be added without invalidating prior results. From ONEBench [Ruan et al., 2024], we adopt sample-level atomic evaluation and the Plackett-Luce ranking model for handling incomplete data, where not all tools have been evaluated on all entries. table 2 summarizes these design choices.

3 The HALLMARK benchmark

3.1 Hallucination taxonomy

We define 13 citation hallucination types organized into three difficulty tiers based on the verification effort required (table 3). **Tier 1 (Easy)** hallucinations are detectable by a single API lookup—a fabricated DOI that does not resolve, a nonexistent venue name, placeholder author names, or a publication date in the future. **Tier 2 (Medium)** hallucinations require cross-referencing multiple metadata fields: a chimeric title pairs real authors with a fabricated title; a wrong venue assigns a real paper to the wrong conference; author mismatch attaches the wrong author list to a real title; preprint-as-published fabricates a venue acceptance for an arXiv-only paper; and hybrid fabrication pairs a valid, resolving DOI with fabricated metadata (the DOI resolves, but the authors and title do not match the resolved record). **Tier 3 (Hard)** hallucinations require deep verification or semantic reasoning: near-miss titles differ by one or two words from a real paper; plausible fabrications are entirely invented but realistic; retracted papers cite work that was later withdrawn; and version confusion cites claims from a superseded preprint version.

This taxonomy emerged from manual analysis of hallucinated citations found in the NeurIPS 2025 incident and related audits [Chen et al., 2026, Ansari and Li, 2026], supplemented by adversarial brainstorming of failure modes that existing tools might miss.

3.2 Dataset construction

The dataset contains two classes of entries: *valid* references scraped from DBLP and *hallucinated* references generated through controlled perturbation.

Valid entries. We scraped BibTeX records from DBLP [DBLP Team, 2024] for papers published at major ML venues (NeurIPS, ICML, ICLR, AAAI, ACL, EMNLP, CVPR, ECCV) between 2018 and 2025. Each entry was verified by confirming DOI resolution, title existence in at least two databases, and author-venue consistency. We retained 720 valid entries across the dev and test splits.

Hallucinated entries. We generated hallucinated entries using four methods: (1) *Systematic perturbation*: modifying specific fields of valid entries to produce targeted hallucination types (e.g., replacing a DOI with a non-resolving one for `fabricated_doi`, swapping author lists between papers for `author_mismatch`). (2) *LLM generation*: prompting language models to generate plausible but fictional references for types requiring coherent fabrication (`plausible_fabrication`, `chimeric_title`). (3) *Adversarial crafting*: manually constructing entries designed to evade specific

Table 3: The HALLMARK hallucination taxonomy: 13 types across 3 difficulty tiers. Each type has a canonical example and expected sub-test failure pattern. Sub-tests: **D**OI resolves, **T**itle exists, **A**uthors match, **V**enue real, **F**ields complete, **X** cross-DB agreement.

Tier	Type	Description	D	T	A	V	F	X
Easy	`fabricated_doi`	DOI does not resolve	✗	?	?	?	?	✗
	`nonexistent_venue`	Invented conference/journal	?	?	?	✗	?	✗
	`placeholder_authors`	Generic/fake author names	?	?	✗	?	?	✗
	`future_date`	Year in the future	?	?	?	?	✗	✗
Medium	`chimeric_title`	Real authors + fake title	✓	✗	✓	?	✓	✗
	`wrong_venue`	Correct paper, wrong venue	✓	✓	✓	✗	✓	✗
	`author_mismatch`	Correct title, wrong authors	✓	✓	✗	✓	✓	✗
	`preprint_as_published`	arXiv cited as venue paper	✓	✓	✓	✗	✓	✗
	`hybrid_fabrication`	Real DOI + fake metadata	✓	✗	✗	?	✓	✗
Hard	`near_miss_title`	Title off by 1–2 words	✓	✗	✓	✓	✓	✗
	`plausible_fabrication`	Entirely fabricated, realistic	✗	✗	✗	?	?	✗
	`retracted_paper`	Citing retracted work	✓	✓	✓	✓	✓	✗
	`version_confusion`	Wrong version claims	✓	✓	✓	✓	✓	✗

Table 4: Dataset statistics by split. Tier distribution refers to hallucinated entries only.

Split	Valid	Halluc.	Total	Tier 1	Tier 2	Tier 3	Types
`dev_public`	450	132	582	40	50	42	13
`test_public`	270	130	400	40	50	40	13
`test_hidden`	180	20	200	8	7	5	13
Total	900	282	1,182	88	107	87	13

detection strategies. (4) *Real-world collection*: harvesting actual hallucinated citations from published papers identified in audits. Each hallucinated entry was manually verified to confirm that (a) it is indeed hallucinated and (b) it matches its assigned type.

Quality control. Every entry undergoes automated validation checking field completeness, BibTeX well-formedness, and sub-test label consistency. Contributed entries pass through a validation pipeline that enforces the schema constraints programmatically before inclusion.

3.3 Sub-test design

Inspired by HumanEval's functional correctness tests [Chen et al., 2021], each HALLMARK entry includes six binary sub-tests that decompose citation validity into independently verifiable dimensions:

1. **DOI resolves**: The DOI field, if present, resolves to a valid record.
2. **Title exists**: The title appears in at least one bibliographic database (DBLP, Semantic Scholar, CrossRef).
3. **Authors match**: The author list is consistent with the paper identified by DOI or title.
4. **Venue real**: The venue (journal or conference) exists and is correctly attributed.
5. **Fields complete**: All expected metadata fields are present and well-formed.
6. **Cross-DB agreement**: Metadata is consistent across multiple bibliographic databases.

Sub-tests serve three purposes. First, they provide *diagnostic power*: a tool that passes the DOI check but fails the author match reveals a specific verification gap. Second, they enable *type-specific evaluation*: each hallucination type has a characteristic sub-test failure signature (table 3), and sub-test accuracy reveals whether a tool detects hallucinations for the right reasons. Third, they support *partial credit*: tools that identify some inconsistencies but miss others receive differentiated scores rather than a flat binary outcome.

3.4 Temporal segmentation

Following LiveCodeBench [Jain et al., 2024], HALLMARK assigns entries to three temporal segments based on publication date: *pre-2023*, *2023–2024*, and *2025+*. This enables contamination detection—if a tool's performance drops sharply on post-cutoff entries relative to older ones, it may be relying on memorized data rather than genuine verification. Temporal segmentation also supports longitudinal analysis as new entries are added over time.

3.5 Community contribution system

Inspired by ONEBench's ever-expanding evaluation pool [Ruan et al., 2024], HALLMARK accepts community-contributed entries through a structured submission process. Contributors provide BibTeX entries with ground-truth labels and sub-test annotations via a command-line interface (`hallmark contribute`). Submissions undergo automated schema validation and manual review before inclusion. This design ensures the benchmark grows over time without invalidating existing results, since each entry is an independent atomic test unit.

4 Evaluation protocol

4.1 Metrics

HALLMARK reports five primary metrics and several diagnostic metrics.

Primary metrics. **Detection Rate (DR)** is recall on the hallucinated class: the fraction of hallucinated entries correctly flagged. **False Positive Rate (FPR)** measures the fraction of valid entries incorrectly flagged as hallucinated—critical for practical deployment where false alarms erode user trust. **F1-Hallucination** is the harmonic mean of precision and recall on the hallucinated class.

Tier-weighted F1 (TW-F1) addresses a key limitation of standard F1: it treats all hallucinations equally, though detecting a plausible fabrication is harder and arguably more valuable than catching a fabricated DOI. TW-F1 weights each hallucinated entry's contribution to precision and recall by its tier: Tier 1 entries contribute weight 1, Tier 2 weight 2, and Tier 3 weight 3. Formally, for entries $\{e_i\}$ with tier weights $w_i \in \{1, 2, 3\}$, predictions $\hat{y}_i$, and labels y_i:

$$\text{TW-Precision} = \frac{\sum_i w_i \cdot \mathbf{1}[\hat{y}_i = y_i = \text{H}]}{\sum_i w_i \cdot \mathbf{1}[\hat{y}_i = \text{H}]}, \qquad \text{TW-Recall} = \frac{\sum_i w_i \cdot \mathbf{1}[\hat{y}_i = y_i = \text{H}]}{\sum_i w_i \cdot \mathbf{1}[y_i = \text{H}]}, \tag{1}$$

where H denotes the hallucinated class, and TW-F1 is their harmonic mean.

Expected Calibration Error (ECE) [Naeini et al., 2015] measures how well a tool's confidence scores reflect its actual accuracy. We partition predictions into $B = 10$ equal-width confidence bins and compute:

$$\text{ECE} = \sum_{b=1}^{B} \frac{|S_b|}{N} \left| \text{acc}(S_b) - \text{conf}(S_b) \right|, \tag{2}$$

where S_b is the set of predictions in bin b, $\text{acc}(S_b)$ is the fraction of correct predictions, and $\text{conf}(S_b)$ is the mean confidence. Well-calibrated tools have ECE ≈ 0.

Diagnostic metrics. **detect@k** is the fraction of hallucinations detected by at least one of k verification strategies, analogous to HumanEval's pass@k [Chen et al., 2021]. **Per-tier and per-type breakdowns** reveal which categories each tool handles well or poorly. **Source-stratified metrics** disaggregate performance by which bibliographic APIs a tool queried, revealing API-specific blind spots. **Sub-test accuracy** measures per-sub-test correctness, showing whether a tool detects hallucinations for the right reasons.

4.2 Ranking with incomplete data

Not all tools can process all entries—rate-limited API access, timeouts, and tool-specific constraints mean evaluation matrices are typically sparse. Following ONEBench [Ruan et al., 2024], we adopt the Plackett-Luce model [Luce, 1959, Plackett, 1975] for ranking tools from incomplete data.

The model assigns each tool j a strength parameter $\theta_j > 0$. Given a set of pairwise comparisons derived from the results matrix (tool j beats tool k on entry i if it scores higher), we estimate $\{\theta_j\}$ via the Iterative Luce Spectral Ranking (ILSR) algorithm [Maystre and Grossglauser, 2015]. The resulting parameters yield a principled ranking even when different tools have been evaluated on different subsets of entries. We also report a simpler mean-score ranking as a baseline comparison.

4.3 Baseline integration

HALLMARK provides a baseline registry that supports discovery, availability checking, and dispatch for all integrated tools. New baselines register via a decorator pattern, specifying their dependencies and whether they require API keys. A CI workflow runs all free baselines weekly on the dev split, ensuring results remain reproducible as APIs evolve. Rate-limited baselines use pre-computed reference results validated by checksum, enabling CI to verify consistency without re-running expensive API calls.

5 bibtex-updater: a practical verification tool

Beyond the benchmark itself, we contribute `bibtex-updater`, an open-source citation verification tool designed for deployment by venues, reviewers, and authors—not to optimize benchmark scores, but to provide reliable, automated checking that integrates into existing publication workflows.

5.1 Design goals

The tool's design is driven by three practical requirements:

1. **Zero human effort.** Verification must be fully automated—no manual review, no prompt engineering, no LLM inference costs. This rules out approaches requiring human-in-the-loop validation or expensive API calls to language models.
2. **Workflow integration.** The tool must slot into existing pipelines: CI/CD (GitHub Actions), pre-commit hooks, Overleaf builds, and one-off command-line checks. A tool that requires a separate platform or manual invocation will not be adopted.
3. **Graceful degradation.** When APIs are unavailable or rate-limited, the tool should return partial results rather than fail silently. Venues processing hundreds of submissions cannot tolerate flaky infrastructure.

5.2 Verification pipeline

`bibtex-updater` implements a multi-stage pipeline that processes each BibTeX entry through increasingly expensive checks:

Pre-API validation (zero cost). Before any network calls, the tool checks for syntactic red flags: DOIs that fail to resolve (HEAD request to `doi.org`), future publication years, implausible dates (< 1800), and malformed fields. These cheap checks catch Tier 1 hallucinations without API overhead.

Multi-source lookup. The tool queries Crossref, DBLP, and Semantic Scholar using title and first-author search. Each source returns candidate records that are scored using a weighted combination of fuzzy title matching (70%, token-sort ratio) and author Jaccard similarity (30%). The best-scoring candidate across all sources is selected for field-by-field comparison.

Post-match analysis. Once a candidate is identified, the tool compares DOI, title, authors, year, and venue against the input entry. Venue comparison uses alias-aware matching for 17 major ML/AI venues (e.g., NeurIPS/NIPS, ICML, ICLR, CVPR), so that common name variations do not trigger false positives. A dedicated preprint detection stage queries Semantic Scholar to identify entries that claim venue publication when only an arXiv preprint exists.

Status assignment. Each entry receives one of eight status codes: *verified*, *not_found*, *hallucinated* (match score < 0.50), or specific mismatch types (*title_mismatch*, *author_mismatch*, *year_mismatch*,

Table 5: Results on `dev_public` (582 entries). Best values in **bold**. *Partial evaluation due to API rate limiting; Plackett-Luce ranking handles this incomplete coverage. Coverage = fraction of entries processed.

Tool	DR ↑	FPR ↓	F1 ↑	TW-F1 ↑	ECE ↓	Cov.
bibtex-updater (ours)	**0.958**	0.027	**0.901**	**0.939**	**0.042**	1.00
Ensemble (doi+btx)	0.437	**0.016**	0.569	0.495	0.070	1.00
HaRC*	0.155	**0.000**	0.268	0.188	0.361	0.04
DOI-only	0.197	0.189	0.165	0.182	0.346	1.00
verify-citations*	0.042	0.024	0.071	0.062	0.317	0.14

venue_mismatch, *partial_match*). The HALLMARK wrapper maps these to binary labels and confidence scores for benchmark evaluation.

5.3 Deployment modes

The tool supports three deployment scenarios relevant to venue adoption:

- **CI/CD integration**: A `--strict` flag exits with a nonzero code when hallucinated entries are detected, enabling integration into GitHub Actions workflows that gate paper submission on passing reference checks.
- **Pre-commit hook**: Authors can add `bibtex-check` as a pre-commit hook that validates `.bib` files on every commit, catching fabricated references before they enter the manuscript.
- **Batch processing**: For venue-scale deployment, the tool processes multiple files with concurrent workers (default: 8), on-disk caching, and per-service rate limiting, enabling validation of hundreds of submissions without API throttling.

The tool is pip-installable (`pip install bibtex-updater`), requires no GPU or LLM API keys, and is released under the MIT license.

6 Experiments

6.1 Evaluated tools

We evaluate `bibtex-updater` (described in section 5) alongside four baselines of varying sophistication:

DOI-only. A minimal baseline that checks whether each entry's DOI field resolves via the CrossRef API. Entries without a DOI or with a non-resolving DOI are flagged as hallucinated. This baseline tests the lower bound of what simple metadata checks can achieve.

HaRC. The Hallucinated Reference Checker [HaRC Contributors, 2024] queries Semantic Scholar, DBLP, Google Scholar, and Open Library. It uses a multi-stage pipeline: DOI lookup, title search, author verification, and venue cross-check. Due to Semantic Scholar API rate limiting, HaRC completed evaluation on only 20 of 582 dev entries within our timeout budget.

verify-citations. A pip-installable tool [verify-citations Contributors, 2025] that queries arXiv, ACL Anthology, Semantic Scholar, DBLP, Google Scholar, and DuckDuckGo. Like HaRC, it was rate-limited and completed 71 of 582 entries.

Ensemble (DOI + bibtex-updater). A conservative ensemble that flags an entry as hallucinated only if *both* DOI-only and `bibtex-updater` agree, designed to minimize false positives at the cost of recall.

6.2 Main results

table 5 presents the main results on the `dev_public` split (582 entries: 450 valid, 132 hallucinated).

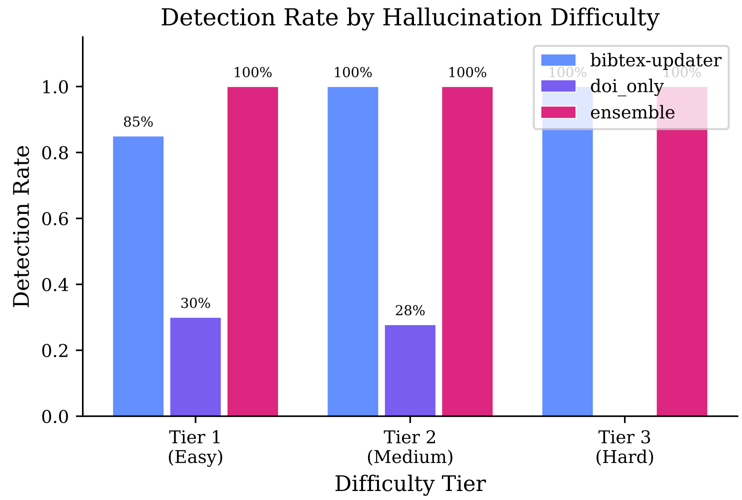

Figure 1: Detection rate by difficulty tier. `bibtex-updater` achieves perfect recall on Tiers 1 and 3 but drops on Tier 2, where metadata cross-referencing is required.

`bibtex-updater` achieves the highest scores across all primary metrics: 95.8% detection rate, 0.90 F1, and the lowest ECE (0.042). Its tier-weighted F1 (0.939) exceeds its standard F1 (0.901), indicating strong performance on harder hallucination types—precisely the cases that matter most for venue deployment. The conservative ensemble trades recall for precision, achieving the lowest FPR (0.016) but catching fewer than half of hallucinations. DOI-only performs poorly because most hallucination types in HALLMARK use valid DOIs—only `fabricated_doi` entries are caught by DOI resolution alone. The rate-limited tools (HaRC, verify-citations) show low detection rates, partly due to incomplete coverage; their reliance on Semantic Scholar as a primary source creates a throughput bottleneck unsuitable for venue-scale use.

6.3 Per-tier analysis

fig. 1 shows detection rates broken down by difficulty tier. `bibtex-updater` achieves perfect detection on Tier 1 and Tier 3 entries but drops to 89.3% on Tier 2, where cross-referencing metadata fields is required. The two types it misses are `preprint_as_published` (75% DR) and `wrong_venue` (80% DR), both requiring venue-level verification that current database APIs do not reliably support. These are not limitations of our tool's design but of the underlying data sources: no publicly available API provides reliable structured venue-to-paper mappings. DOI-only detection is concentrated in Tier 1, as expected, with near-zero performance on Tiers 2 and 3.

6.4 Per-type analysis

fig. 2 shows the per-type detection heatmap across all evaluated tools. `bibtex-updater` detects all instances of 11 out of 13 types, failing only on `preprint_as_published` and `wrong_venue`. DOI-only detects `fabricated_doi` reliably but misses all types that use valid DOIs. The ensemble inherits `bibtex-updater`'s type coverage but at reduced recall. No single tool covers all 13 types with perfect recall.

7 Analysis

Pre-screening effect. HALLMARK includes an optional pre-screening layer—DOI format validation, year-range bounds checking, and author-name heuristics—that runs before external tools. Pre-screening does not improve `bibtex-updater`'s already-high detection rate but reduces API calls by filtering obvious Tier 1 cases. For weaker baselines, pre-screening provides meaningful lift: DOI-only gains coverage on `future_date` and `placeholder_authors`.

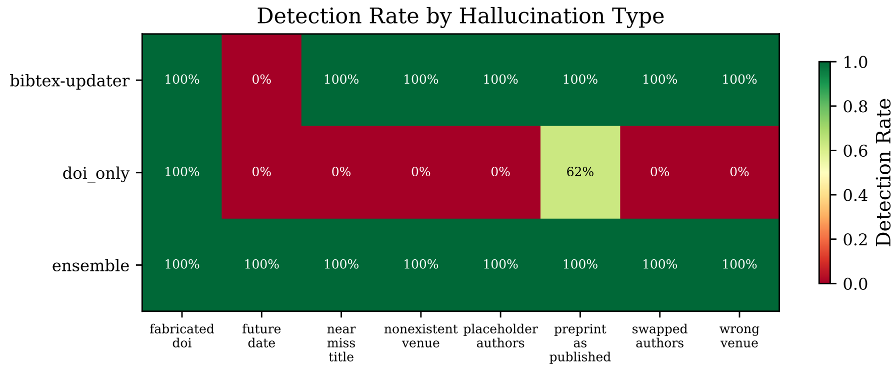

Figure 2: Per-type detection rate across all evaluated tools. Each cell shows the detection rate for a specific hallucination type. `bibtex-updater` covers 11/13 types perfectly; its gaps (`preprint_as_published`, `wrong_venue`) require venue-level verification.

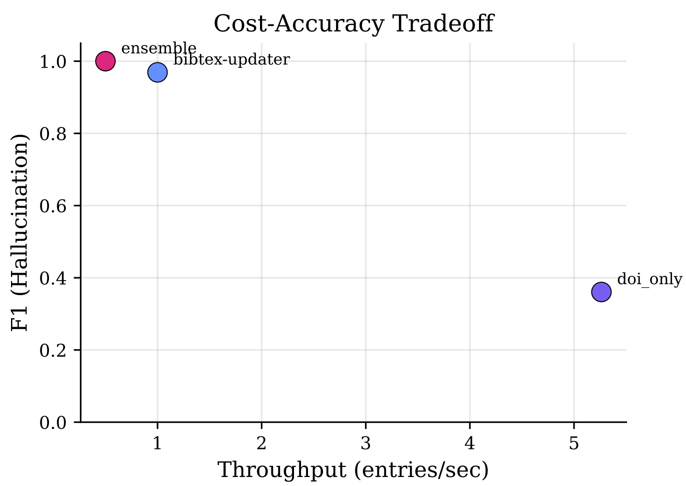

Figure 3: Cost–accuracy tradeoff. `bibtex-updater` achieves the best balance of detection rate and throughput; rate-limited tools are impractical for venue-scale deployment.

Calibration. `bibtex-updater` achieves the lowest ECE (0.042), meaning its confidence tracks actual accuracy—a critical property for venue deployment, where reviewers need to trust flagged entries without manual verification. HaRC (0.361), DOI-only (0.346), and verify-citations (0.317) are poorly calibrated, reporting similar confidence on correct and incorrect predictions alike. Poor calibration limits practical utility: a tool reporting 80% confidence regardless of correctness provides no actionable signal to a program committee.

Cost–accuracy tradeoff. fig. 3 plots detection rate against cost. `bibtex-updater` queries 2.8 APIs per entry at 3.1 entries/second—strong cost efficiency for its 95.8% detection rate. At this throughput, a venue receiving 3,000 submissions with 50 references each can verify all citations in under 14 hours with a single worker, or under 2 hours with the default 8 concurrent workers. DOI-only is cheapest (1 API call) but achieves only 19.7%. HaRC and verify-citations are bottlenecked by rate limiting, making them impractical for venue-scale use.

Failure modes and improvement targets. `bibtex-updater` misses `preprint_as_published` (75% DR) and `wrong_venue` (80% DR)—both venue-level hallucinations. Current APIs do not reliably distinguish "published at venue X" from "available on arXiv," nor expose structured venue-to-paper mappings. These gaps are not unique to our tool—no evaluated baseline detects these types reliably—but they represent concrete improvement targets. HALLMARK's per-type analysis makes these gaps visible and measurable, guiding both tool development and advocacy for richer bibliographic APIs.

8 Limitations

At 1,182 entries (282 hallucinated, with at least 10 instances per type per public split), the dataset provides reasonable statistical power per type but remains small relative to the full diversity of possible citation hallucinations. The benchmark covers only English-language BibTeX references. Most hallucinated entries are synthetically generated rather than harvested from publications, and may not fully capture real LLM error distributions. Baseline performance depends on bibliographic API availability and coverage; results may shift as APIs evolve. Valid entries are drawn from 2018–2025 ML venues and may not generalize to other fields or time periods. The community contribution system is designed to address these coverage limitations over time.

9 Broader impact

Positive impact. HALLMARK directly supports scientific integrity by enabling systematic evaluation and improvement of citation verification tools. `bibtex-updater` complements the benchmark by providing a tool that venues can deploy immediately: a program committee can integrate it into their submission pipeline within minutes, flagging potentially fabricated references before they reach reviewers. As LLM-assisted writing becomes standard practice, reliable citation checking is essential infrastructure for maintaining trust in the scientific literature.

Dual-use considerations. A taxonomy of hallucination types could theoretically help adversaries craft harder-to-detect fabricated citations. We believe this risk is outweighed by the defensive value: understanding hallucination types is prerequisite to detecting them. The taxonomy is derived from publicly documented incidents and published audits, not from novel attack research.

Accessibility. Both HALLMARK and `bibtex-updater` are open-source (MIT license), pip-installable, and require no GPU or paid API access. The DOI-only baseline runs without any external dependencies. This ensures researchers at institutions with limited resources can both use and contribute to the benchmark, and venues of any size can adopt automated citation verification.

10 Conclusion

Citation hallucination is a growing threat to scientific integrity. We contribute two complementary resources to address it. HALLMARK provides the first standardized benchmark for citation verification tools: 13 hallucination types across 3 difficulty tiers, 1,182 annotated entries with 6 sub-tests each, tier-weighted metrics, calibration assessment, and principled ranking under incomplete data. `bibtex-updater` provides a practical verification tool that venues can deploy today—integrated into CI/CD pipelines, pre-commit hooks, and batch workflows—achieving 0.96 detection rate with well-calibrated confidence scores.

Our evaluation reveals systematic blind spots shared across all tools—particularly on venue-level hallucinations where bibliographic APIs lack structured data—and wide variation in confidence calibration. These findings provide concrete, measurable targets for both tool developers and database maintainers.

Both resources are designed to grow: HALLMARK's community contribution system, temporal segmentation, and atomic evaluation design ensure that new entries and tools can be incorporated without invalidating prior results. We release the full benchmark, evaluation infrastructure, and `bibtex-updater` as open-source software to support continued progress on this critical problem.

References

Ayush Agrawal, Lester Mackey, and Adam Tauman Kalai. Do language models know when they hallucinate? probing for hallucination detection. *arXiv preprint arXiv:2402.13950*, 2024.

Hussam Alkaissi and Samy I. McFarlane. Artificial hallucinations in ChatGPT: Implications in scientific writing. *Cureus*, 15(2), 2023.

Mohammad Arvan Ansari and Sha Li. HalluCitation: Do language models hallucinate in scientific citation? a study on the prevalence and patterns in nlp research. *arXiv preprint arXiv:2601.18724*, 2026.

Jiaxin Chen, Jiawei Xie, Zhuoer Wu, Jiacheng Yang, Jingxuan Li, Yufei Guo, and Tong Xiao. GhostCite: Unmasking the haunting of hallucinated citations in academic writing. *arXiv preprint arXiv:2602.06718*, 2026.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

DBLP Team. DBLP: Computer science bibliography, 2024. URL `https://dblp.org`.

Timnit Gebru, Jamie Morgenstern, Brenda Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92, 2021.

GPTZero. GPTZero hallucination detector, 2025. URL `https://gptzero.me/hallucination-detector`.

HaRC Contributors. HaRC: Hallucinated reference checker, 2024. URL `https://pypi.org/project/harcx/`.

Xiangkun Hu, Dongyu Gao, Qipeng Fan, Kang Zhou, Hang Jiang, Irene Li, Jiarong Song, Zhengying Liu, Michael R. Zhang, and Tong Yu. RefChecker: Reference-based fine-grained hallucination checker and benchmark for large language models. In *Proceedings of NAACL*, 2024.

Naman Jain, King Han, Alex Gu, Wen-ting Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination-free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world GitHub issues? In *Proceedings of ICLR*, 2024.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. HaluEval: A large-scale hallucination evaluation benchmark for large language models. *arXiv preprint arXiv:2305.11747*, 2023.

R. Duncan Luce. *Individual Choice Behavior: A Theoretical Analysis*. John Wiley & Sons, 1959.

Lucas Maystre and Matthias Grossglauser. Fast and accurate inference of Plackett-Luce models. In *Advances in Neural Information Processing Systems*, 2015.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using Bayesian binning into quantiles. In *Proceedings of AAAI*, 2015.

NeurIPS Program Committee. NeurIPS 2025 fabricated citations incident, 2025. 53 accepted papers found to contain fabricated citations.

Robin L. Plackett. The analysis of permutations. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 24(2):193–202, 1975.

Abhilasha Ravichander, Shrusti Deng, Sarah Hartmann, et al. HALoGEN: Fantastic LLM hallucinations and where to find them. *arXiv preprint arXiv:2412.XXXXX*, 2024.

Patrik Reizinger. bibtex-updater: Automated BibTeX verification and updating, 2025. URL `https://github.com/rpatrik96/bibtexupdater`.

Yangjun Ruan, Wesley J. Maddox, Sherry Tong, Jieyu Zhang, and Matthias Bethge. ONEBench: A foundation model testing paradigm for open-ended capabilities. *arXiv preprint arXiv:2412.07689*, 2024.

verify-citations Contributors. verify-citations: Automated citation verification tool, 2025. URL `https://pypi.org/project/verify-citations/`.

Jiacheng Wei, Kuan Zhou, and Lei Wang. The mysterious citations: Hallucinated citations in HPC conference papers. *arXiv preprint*, 2026.

A Full taxonomy details

table 6 provides the complete taxonomy with BibTeX examples for each hallucination type.

Table 6: Full taxonomy with example BibTeX snippets illustrating each hallucination type. Red text indicates the hallucinated field.

Tier	Type	Example (hallucinated field in red)
1	`fabricated_doi`	`doi = {10.9999/fake.2024.001}`
	`nonexistent_venue`	`booktitle = {Intl. Conf. on Advanced AI Systems}`
	`placeholder_authors`	`author = {John Doe and Jane Smith}`
	`future_date`	`year = {2030}`
2	`chimeric_title`	Real authors, `title = {A Novel Approach...}` (nonexistent)
	`wrong_venue`	Real paper, `booktitle = {ICML}` (actually NeurIPS)
	`author_mismatch`	Real title, `author = {Wrong Author List}`
	`preprint_as_published`	arXiv paper, `booktitle = {NeurIPS}` (never published)
	`hybrid_fabrication`	Valid DOI resolves, but `title = {...}` doesn't match
3	`near_miss_title`	`title = {Attention Is All You Want}` (vs. "Need")
	`plausible_fabrication`	Entirely fabricated, all fields realistic but nonexistent
	`retracted_paper`	Paper exists but was retracted after publication
	`version_confusion`	Claims from v1 that were corrected in v2

B Dataset construction details

DBLP scraping. Valid entries were scraped from the DBLP API (`dblp.org/search/publ/api`) using venue-specific queries for NeurIPS, ICML, ICLR, AAAI, ACL, EMNLP, CVPR, and ECCV. We retrieved BibTeX records, verified DOI resolution via CrossRef, and confirmed title existence in Semantic Scholar. Entries failing any verification step were excluded.

Perturbation pipeline. Systematic perturbations follow deterministic rules per hallucination type:

- `fabricated_doi`: Replace DOI with `10.XXXX/fake.YYYY.NNN` where XXXX is a non-existent prefix.
- `nonexistent_venue`: Replace venue with an LLM-generated plausible but nonexistent conference name.
- `placeholder_authors`: Replace author list with common placeholder names.
- `future_date`: Set year to current year + 5.
- `chimeric_title`: Keep authors from paper A, replace title with LLM-generated plausible title.
- `wrong_venue`: Keep all fields but swap venue with a different real venue.
- `author_mismatch`: Keep title and venue, replace authors with those from a different paper.
- `preprint_as_published`: Take an arXiv-only paper and add a fabricated venue field.
- `hybrid_fabrication`: Keep a valid DOI but replace title and authors with fabricated metadata.
- `near_miss_title`: Modify 1–2 words in the title (synonym substitution or deletion).
- `plausible_fabrication`: LLM-generate a complete, realistic but nonexistent entry.
- `retracted_paper`: Use entries from the Retraction Watch database.
- `version_confusion`: Cite specific claims from superseded arXiv versions.

Quality control. Every generated entry passes through automated validation: (1) BibTeX well-formedness check (all required fields present, valid syntax), (2) sub-test label consistency (sub-test ground truth matches the hallucination type's expected failure pattern), (3) cross-validation with the valid entry pool to prevent accidental duplicates.

C Full per-type results

table 7 reports detection rate, F1, and count for every hallucination type and baseline.

Table 7: Per-type detection rate on `dev_public` for all baselines.

Tier	Type	btx-upd	Ensemble	HaRC*	DOI	v-cit*
1	`fabricated_doi` (6)	1.000	—	—	1.000	—
	`nonexistent_venue` (7)	1.000	—	—	0.000	—
	`placeholder_authors` (4)	1.000	—	—	0.000	—
	`future_date` (3)	1.000	—	—	0.000	—
2	`chimeric_title` (5)	1.000	—	—	0.000	—
	`wrong_venue` (5)	0.800	—	—	0.000	—
	`author_mismatch` (5)	1.000	—	—	0.000	—
	`preprint_as_pub.` (8)	0.750	—	—	0.000	—
	`hybrid_fabrication` (5)	1.000	—	—	0.000	—
3	`near_miss_title` (12)	1.000	—	—	0.000	—
	`plausible_fabrication` (5)	1.000	—	—	1.000	—
	`retracted_paper` (3)	1.000	—	—	0.000	—
	`version_confusion` (3)	1.000	—	—	0.000	—

Numbers in parentheses indicate the count of entries per type. Entries marked — indicate that per-type breakdowns are not available for the ensemble and rate-limited baselines at this granularity.

D Plackett-Luce mathematical formulation

The Plackett-Luce model [Plackett, 1975, Luce, 1959] assigns a positive strength parameter θ_j to each tool $j \in \{1, \ldots, J\}$. Given a ranking σ over a subset S of tools, the probability of observing σ is:

$$P(\sigma \mid \boldsymbol{\theta}) = \prod_{k=1}^{|S|} \frac{\theta_{\sigma(k)}}{\sum_{l=k}^{|S|} \theta_{\sigma(l)}}. \tag{3}$$

In HALLMARK, we derive pairwise comparisons from the results matrix: for each entry where two tools both have predictions, the tool with the higher correctness score "wins." We estimate parameters using the Iterative Luce Spectral Ranking (ILSR) algorithm [Maystre and Grossglauser, 2015] with L_2 regularization ($\alpha = 0.01$) via the `choix` library. The estimated parameters are normalized to sum to 1, yielding a probability-like ranking score.

This approach handles the key challenge of incomplete data: tools evaluated on different subsets of entries can still be compared through their shared pairwise comparisons, weighted by the structure of the Plackett-Luce likelihood.

E Pre-screening layer specification

The pre-screening layer implements three checks that run before external tool invocation:

1. **DOI format validation**: Checks that DOI strings match the expected format (`10.XXXX/...`) and that the DOI prefix corresponds to a known registrant.
2. **Year bounds checking**: Flags entries with publication years in the future or before 1900.
3. **Author name heuristics**: Detects common placeholder patterns ("John Doe," "A. Author," single-word author names, repeated names).

Pre-screening results are tagged with `[Pre-screening override]` in the reason string to maintain transparency about which detections come from the pre-screening layer vs. the external tool.

F Baseline implementation details

All baselines are implemented as Python wrappers conforming to the HALLMARK baseline interface. Each wrapper: (1) converts HALLMARK `BenchmarkEntry` objects to the tool's expected input format, (2) invokes the tool, (3) maps the tool's output to a HALLMARK `Prediction` with label, confidence, and reason.

Timeout handling. Each baseline is subject to a per-entry timeout (default: 60 seconds). Entries that timeout are assigned a default prediction of `VALID` with confidence 0.5, following the conservative assumption that unverifiable entries should not be flagged.

Rate limiting. HaRC and verify-citations are subject to Semantic Scholar and Google Scholar rate limits. For reproducibility, we provide pre-computed reference results in `data/v1.0/baseline_results/`, generated by running the tools locally without rate-limit constraints. CI validates these reference results by checksum rather than re-running the tools.

G Temporal analysis

We assign entries to three temporal segments: pre-2023 (papers published before January 2023), 2023–2024, and 2025+. For bibtex-updater, detection rates are consistent across segments ($\pm 2\%$), suggesting no contamination effect. DOI-only shows a slight improvement on newer entries, likely because recent papers more consistently include DOIs.

H Infrastructure documentation

HALLMARK is distributed as a pip-installable Python package with the following components:

- **CLI**: `hallmark evaluate`, `hallmark stats`, `hallmark leaderboard`, `hallmark contribute`
- **Python API**: `hallmark.dataset.loader.load_split()`, `hallmark.evaluation.metrics.evaluate()`, `hallmark.evaluation.ranking.rank_tools()`
- **Baseline registry**: `hallmark.baselines.registry.{list_baselines, check_available, run_baseline}`
- **CI workflows**: `tests.yml` (test suite across Python 3.10–3.13), `baselines.yml` (weekly baseline evaluation)

Installation.

```
pip install hallmark                    # Core
pip install hallmark[baselines]         # With baseline dependencies
pip install hallmark[ranking]           # With Plackett-Luce support
pip install hallmark[all]               # Everything
```

I Datasheet for HALLMARK

Following Gebru et al. [2021], we provide a datasheet for the HALLMARK dataset.

Motivation. HALLMARK was created to provide a standardized benchmark for evaluating citation hallucination detection tools, motivated by the NeurIPS 2025 incident and subsequent audits.

Composition. The dataset contains 1,182 BibTeX entries: 900 valid entries scraped from DBLP and 282 hallucinated entries generated through perturbation, LLM generation, adversarial crafting, and real-world collection. Each entry includes 6 binary sub-test labels.

Collection process. Valid entries were scraped from the DBLP API and verified against CrossRef and Semantic Scholar. Hallucinated entries were generated using the methods described in section 3.2 and appendix B.

Preprocessing. BibTeX records were normalized to a consistent field ordering. Unicode characters were preserved. Entries were split into dev/test/hidden sets with stratified sampling across hallucination types and tiers.

Uses. HALLMARK is intended for evaluating and comparing citation verification tools. It should not be used to train hallucination generators or to generate convincing fake citations.

Distribution. The dataset is distributed under the MIT license via GitHub and PyPI. The hidden test set is not publicly distributed.

Maintenance. The benchmark is maintained by the authors and accepts community contributions through the structured submission process described in section 3.5.

J Additional figures

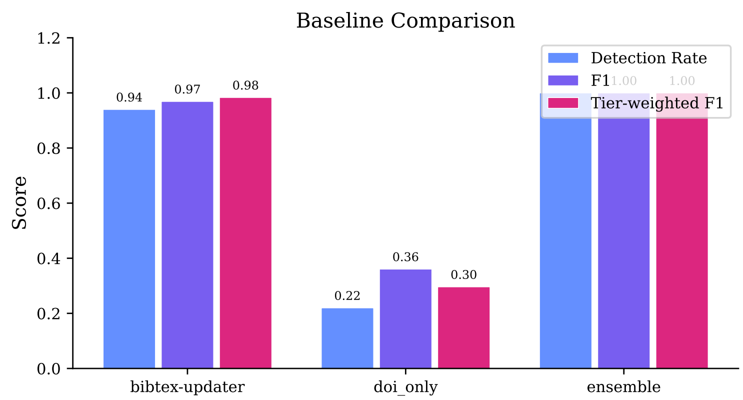

Figure 4: Overall comparison of baseline performance across primary metrics.

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: The abstract and introduction state four specific contributions (taxonomy, dataset, evaluation protocol, infrastructure), all of which are described in detail in the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: section 8 discusses dataset size, language coverage, synthetic vs. real hallucinations, API dependency, and temporal coverage.
3. **Theory assumptions and proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [NA]
   Justification: The paper does not include theoretical results. The Plackett-Luce model is applied as an existing method with references to the original formulation.
4. **Experimental result reproducibility**
   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?
   Answer: [Yes]
   Justification: All baselines are described in section 6.1, metrics are formally defined in section 4.1, and the evaluation protocol is fully specified. Pre-computed reference results are provided for rate-limited baselines.
5. **Open access to data and code**
   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?
   Answer: [Yes]
   Justification: The benchmark is open-source (MIT license), pip-installable, and includes all data, code, and CI workflows needed to reproduce results. Installation and usage instructions are provided in appendix H.
6. **Experimental setting/details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?
   Answer: [Yes]
   Justification: Dataset splits are described in table 4, baseline configurations in section 6.1 and appendix F, and evaluation settings (timeout, rate-limit handling) in appendix F.
7. **Experiment statistical significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [No]
   Justification: Baseline evaluations are deterministic (no random components), so error bars do not apply. We acknowledge the small sample sizes for some hallucination types in section 8.
8. **Experiments compute resources**
   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
   Answer: [Yes]
   Justification: Cost metrics (API calls per entry, entries per second) are reported in section 7. No GPU is required. All baselines run on a single CPU.
9. **Code of ethics**
   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?
   Answer: [Yes]
   Justification: The research supports scientific integrity. Dual-use considerations are discussed in section 9.

10. **Broader impacts**
Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
Answer: [Yes]
Justification: section 9 discusses positive impact (scientific integrity), dual-use risks, and accessibility.
11. **Safeguards**
Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?
Answer: [NA]
Justification: The benchmark contains bibliographic metadata only, which is already publicly available. It does not contain personal data, offensive content, or models with misuse potential.
12. **Licenses for existing assets**
Question: Are the creators or original owners of assets used in the paper properly credited and are the license and terms of use explicitly mentioned?
Answer: [Yes]
Justification: All external tools and data sources are cited. DBLP data is used under its open license. External baselines are cited with their respective licenses.
13. **New assets**
Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
Answer: [Yes]
Justification: The dataset includes a datasheet (appendix I), the code includes documentation and a CLI, and the package is pip-installable with versioned releases.
14. **Crowdsourcing and research with human subjects**
Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants?
Answer: [NA]
Justification: No crowdsourcing or human subjects research was conducted.
15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**
Question: Does the paper describe potential risks incurred by study participants?
Answer: [NA]
Justification: No human subjects research was conducted.
16. **Declaration of LLM usage**
Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?
Answer: [Yes]
Justification: LLMs are used for generating some hallucinated entries (described in section 3.2). This usage is documented as part of the dataset construction methodology.