Enhancing Scientific Reference Accuracy in Large Language Models: An Evaluation of Agentic Frameworks

1. Executive Summary

Large Language Models (LLMs) are rapidly transforming scientific research and technological development, offering capabilities from data analysis to literature synthesis. However, their propensity to generate inaccurate or "hallucinated" scientific references poses a significant threat to academic integrity and the credibility of AI-driven scientific advancement. These errors, ranging from incorrectly formatted citations to entirely fabricated sources, undermine the foundational principles of verifiability and scholarly discourse. This report investigates the problem of LLM citation inaccuracies, focusing on the evaluation of an agentic framework integrated with bibliographic software (e.g., EndNote, Mendeley, Zotero) and academic databases (e.g., Google Scholar, Web of Science, PubMed, CrossRef) as a potential solution.

The analysis reveals that LLMs struggle with scientific references due to their probabilistic nature, limitations in training data (biases, outdated information, lack of access to paywalled databases), and inherent difficulties in generating precisely structured data. The proposed agentic framework aims to mitigate these issues by enabling LLMs to interact with external tools and databases, thereby verifying information, retrieving accurate metadata, and ensuring correct formatting. This approach leverages the LLM's language understanding with the specialized capabilities of bibliographic software and the vast, curated knowledge within academic databases.

A comparative analysis with alternative methods—including Retrieval Augmented Generation (RAG) and its variants (CRAG, Self-RAG), fine-tuning, hybrid LLM-symbolic systems, knowledge graphs, and post-processing validation tools—indicates that while each offers partial solutions, a comprehensive agentic framework orchestrating these components holds the most promise. RAG can provide up-to-date information, fine-tuning can improve an LLM's baseline citation capabilities, symbolic systems can enforce structural rigor, knowledge graphs can offer verifiable factual backbones, and post-processing tools can catch residual errors.

However, the agentic framework solution is not without significant challenges. Technical feasibility hinges on the availability and stability of APIs for diverse bibliographic tools and databases, which is currently inconsistent. The complexity of developing, maintaining, and operating such systems introduces considerable

overhead and potential latency. Issues of data consistency from multiple sources, scalability, security, privacy, and potential bias inherited from tools or data also require careful consideration.

Despite these challenges, the potential benefits of an agentic approach—improved accuracy, automation of tedious bibliographic tasks, enhanced verifiability, and adaptability—are substantial. Future development should focus on modular design, robust error handling, prioritizing verifiable sources, ensuring transparency in agent operations, and incorporating human-in-the-loop mechanisms. Further research into standardized bibliographic APIs, benchmark datasets for citation accuracy, and advanced hybrid models is crucial. Ultimately, while agentic systems can significantly augment scientific workflows, human expertise and oversight remain indispensable for ensuring the integrity of scholarly communication.

2. Introduction: The Crisis of Credibility in LLM-Generated Scientific References

The proliferation of Large Language Models (LLMs) marks a paradigm shift in how scientific research and technological innovation are approached. These sophisticated AI systems are increasingly being integrated into diverse stages of the scientific lifecycle, from initial data analysis and hypothesis generation to assisting in the drafting of scholarly articles and conducting comprehensive literature reviews.[1] Platforms like Microsoft Discovery even envision teams of specialized AI agents collaborating with researchers on complex tasks such as advanced knowledge reasoning and experimental simulation, highlighting the transformative potential of these technologies.[3] As LLMs become more embedded in these high-stakes domains, the accuracy and reliability of their outputs are not merely desirable but critical.

However, a significant impediment to the trustworthy application of LLMs in science is their documented struggle with generating accurate scientific references. LLMs are prone to a phenomenon known as "hallucination," where they produce information that is false, misleading, or unverifiable, yet present it with a high degree of confidence.[4] In the context of scientific referencing, this often manifests as "extrinsic hallucinations"—the fabrication of references that do not exist, the incorrect attribution of authorship, or the generation of invalid Digital Object Identifiers (DOIs).[6] Some research indicates that LLMs can hallucinate up to 27% of the time, with factual errors appearing in as much as 46% of their output.[9] These are not minor inaccuracies; they represent a fundamental flaw in the LLM's ability to engage with scholarly literature authentically. For instance, studies have shown LLMs producing

citations that are correctly formatted but refer to entirely fictional content.[7]

The implications of such inaccuracies for scientific integrity and technological advancement are profound. Fabricated or incorrect citations can mislead researchers, causing them to base new work on non-existent or erroneous foundations, thereby undermining the rigor of the scientific process.[4] This erosion of trust extends beyond individual errors; it can systematically distort the scientific record. For example, LLMs have been observed to reinforce the "Matthew effect" in citations by disproportionately favoring already highly cited papers, potentially marginalizing newer or less mainstream research.[1] The dissemination of fake information or inaccurate summaries can mislead not only the scientific community but also the public, diminishing overall trust in both AI and scientific institutions.[6] This problem is particularly acute as the very foundation of scholarship rests on the ability to accurately cite and build upon prior work. Citations serve to acknowledge the contributions of other researchers, provide evidentiary support for claims, ensure the verifiability of scientific findings, and foster a traceable, coherent scholarly conversation.[10] When LLMs fail to uphold these standards, they threaten the core tenets of scientific communication.

The widespread adoption of LLMs generating faulty citations could lead to a systemic erosion of verifiability and intellectual honesty, potentially culminating in a "citation crisis" where the provenance of scientific ideas becomes obscured. If LLMs are used to synthesize literature and hallucinate citations, they might inadvertently create plausible-sounding but fictional narratives or connections between concepts and researchers, leading to a "semantic drift" in scientific understanding. This is especially dangerous in interdisciplinary research, where reliance on LLM-generated summaries of unfamiliar fields could propagate false premises based on non-existent sources, thereby hindering genuine cross-disciplinary progress. The integrity of the scientific endeavor hinges on addressing this crisis of credibility.

3. Understanding LLM Shortcomings in Scientific Referencing

The difficulties Large Language Models (LLMs) face in accurately generating scientific references stem from a combination of their fundamental architecture, the nature of their training data, and the specific challenges posed by bibliographic information.

Fundamental Reasons for LLM Struggles:

- **Probabilistic Nature and Next-Token Prediction:** At their core, LLMs are sophisticated pattern matchers that generate text by predicting the next token (a word or sub-word unit) in a sequence based on the statistical patterns learned

from their vast training data.[4] They do not "understand" content or verify facts in a human sense. This probabilistic generation means that while they can produce fluent and often plausible-sounding text, including reference strings, these outputs are based on statistical likelihood rather than factual correctness or actual knowledge of the cited work.[4] An LLM might generate a perfectly formatted BibTeX entry for a paper that doesn't exist simply because the sequence of tokens *looks like* a valid entry it has seen many times. This inherent mechanism means LLMs may generalize from limited data or extrapolate incorrect facts when attempting to fulfill a request for a citation.[6] While next-token prediction is an optimization function for language modeling, it does not inherently equip LLMs with the capability for accurate factual recall or structured data generation necessary for reliable referencing.[13]

- **Training Data Limitations:** The data used to train LLMs significantly influences their capabilities and limitations regarding scientific references.
    - **Biases and Misinformation:** LLMs are trained on massive datasets, often scraped from the internet, which inevitably contain errors, outdated information, and biases.[4] These models can replicate and even amplify such misinformation.[5] If the training data includes incorrectly cited papers or discussions of non-existent research, the LLM may learn these erroneous patterns.
    - **Outdated Knowledge/Knowledge Cutoff:** LLMs possess knowledge only up to their last training date (knowledge cutoff).[4] This means they are often unaware of the most recent publications, a critical limitation in rapidly evolving scientific fields. When queried about recent work, they might generate speculative or outdated references rather than admitting a lack of current information.[5]
    - **Lack of Access to Paywalled/Specialized Databases:** A significant portion of scholarly literature resides behind paywalls or in specialized academic databases. Most LLMs are not trained on this content and cannot access these databases in real-time during inference.[4] This severely limits their "knowledge" of the full scope of scientific research, often restricting them to open-access resources and potentially omitting seminal or highly relevant subscription-based publications.
- **Inherent Difficulties with Structured Data Generation:** Bibliographic information, especially in formats like BibTeX or CSL-JSON, is highly structured, requiring precise adherence to syntax, field names, and data types. LLMs, designed primarily for generating fluent natural language, often struggle with the rigid, rule-based nature of such structured data.[15] Their probabilistic, token-by-token generation mechanism is not well-suited for ensuring that every

field, comma, or curly brace in a BibTeX entry is perfectly correct according to a formal grammar. This "structured data blindspot" means that even if an LLM has some representation of a paper's metadata, it may fail to output it in a valid and complete structured format. Research into enabling LLMs to better handle structured data like SQL queries or data-to-text tasks is ongoing, underscoring that this is a non-trivial challenge.[15]

Types and Prevalence of Citation Errors:

These fundamental shortcomings lead to several distinct types of citation errors:

- **Extrinsic Hallucinations:** This is perhaps the most damaging type of error, where the LLM fabricates information with no basis in reality. This includes generating plausible but non-existent DOIs, inventing author lists, citing papers that were never written, or misattributing findings to incorrect sources.[4] These are often presented with high confidence, making them particularly deceptive.
- **Intrinsic Hallucinations:** In this case, the LLM might correctly cite an existing paper but misrepresent its content. This could involve providing an inaccurate summary, attributing claims to the paper that it does not make, or misinterpreting its findings.[4] While the reference itself is real, its use in the LLM's output is misleading.
- **Formatting Errors:** Even when an LLM has access to some correct bibliographic details, it may fail to format them according to standard citation styles (e.g., APA, MLA, Chicago) or structured formats like BibTeX. This can range from minor punctuation errors to completely malformed entries.
- **Quantifying Errors:** Studies have attempted to quantify these errors, revealing a significant problem. For example, some research indicates factual errors in up to 46% of LLM outputs.[9] One analysis found that half of LLM-generated search results lacked citations, and of those with citations, only 75% actually supported the claims made.[4] Furthermore, LLMs may exhibit biases in citation patterns, such as over-citing already prominent papers (the Matthew effect), rather than reflecting a balanced view of the literature.[1] Metrics like sensitivity (changes in prediction due to prompt rephrasing) and consistency (variation in predictions for similar inputs) are being developed to better understand these failure modes.[18]

The "plausibility trap" is a significant concern: LLMs often generate errors, such as correctly formatted but entirely fictional DOIs or author lists that seem appropriate for a given field, which are difficult to detect without careful verification.[4] This realism makes it easier for such errors to propagate, especially when users are not experts in the specific domain or are conducting cursory reviews. In multi-turn interactions or

complex tasks where an LLM builds upon its previous outputs, initial subtle inaccuracies can compound, leading to more substantial citation errors downstream, a problem exacerbated by the models' inherent contextual memory limitations.[4]

4. Proposed Solution: Agentic Frameworks for Enhanced Citation Accuracy

To address the significant shortcomings of Large Language Models (LLMs) in generating accurate scientific references, a promising approach involves the development of agentic frameworks. These frameworks aim to augment LLMs with capabilities for planning, tool use, and interaction with external information sources, thereby creating more robust and reliable citation generation systems.

Overview of Agentic AI: Principles and Architecture

Agentic AI refers to systems designed to perceive their environment, reason about information, formulate plans, and execute actions autonomously or semi-autonomously to achieve specific goals.[19] Unlike standard LLMs that primarily generate responses based on input prompts and internal knowledge, AI agents can interact with external tools, databases, and even other agents.[4] An agentic framework provides the foundational structure for these interactions, defining protocols for communication, coordination, reasoning, and decision-making.[19] In the context of citation accuracy, an agentic system would leverage an LLM as its core reasoning engine but enhance its capabilities by allowing it to actively seek, verify, and integrate information from authoritative bibliographic sources.

Core Components of an Agentic Citation System

A robust agentic system for scientific referencing would typically comprise several key components:

1. **LLM Core:** The central LLM provides natural language understanding, query interpretation, text generation, and the ability to formulate plans and decide on tool usage.
2. **Memory Module:** This component endows the agent with both short-term memory (for ongoing task context) and long-term memory (for storing learned information, past interactions, user preferences, and successful strategies).[22] This allows the agent to improve over time and maintain consistency across interactions.
3. **Planning Module:** This module is responsible for decomposing complex citation tasks (e.g., "find and cite three key papers on topic X published after 2020 in APA

style") into a sequence of smaller, manageable sub-tasks.[19] Examples include identifying keywords, selecting appropriate databases, formulating queries, extracting metadata, verifying information, and formatting the citation.

4. **Tool Use Module:** This critical module enables the agent to interact with external software, APIs, and databases.[4] For citation management, these tools would include connectors to bibliographic software and academic databases. Agentic frameworks like LangChain are designed to facilitate such integrations.[22]

Mechanism: Integrating LLMs with Bibliographic Software and Academic Databases

The core of the proposed solution lies in the agent's ability to dynamically interact with a suite of specialized bibliographic resources.

- Interaction with Bibliographic Software (EndNote, Mendeley, Zotero):
  An AI agent could connect to a user's personal or shared bibliographic libraries to check if a reference already exists, retrieve its metadata, or add newly verified references.
  - **EndNote:** While direct, comprehensive public API access for deep library manipulation by third-party agents appears limited based on current information [27], integrations exist for specific functionalities like "Cite While You Write" in Microsoft Word, which operates in a cloud environment.[28] Programmatic interaction might be possible through intermediary libraries or specific software development kits like Aspose.Words for.NET for document-embedded EndNote data [29], but a general-purpose, open API for full library interaction by an agent is not clearly documented in the provided materials. This presents a potential challenge for seamless integration.
  - **Mendeley:** Mendeley offers a more promising avenue for agentic integration, with a dedicated Developer Portal detailing Core API Resources and Datasets API Resources, including SDKs.[30] These APIs allow authorized programmatic access to user documents, files, group libraries (including private ones), annotations, and folders.[31] This structured access is well-suited for an agent to query, retrieve, and manage bibliographic entries.
  - **Zotero:** Zotero also provides robust API capabilities, as evidenced by existing integrations like the ZoteroRetriever in LangChain.[33] This retriever uses the Zotero API to connect to personal and group libraries, search for items by various parameters (query, item type, tag, etc.), and retrieve full text if available. The Zotero API's support for detailed search syntax and access to library content makes it a strong candidate for agentic interaction.
- Leveraging Academic Databases (Google Scholar, Web of Science, PubMed,

CrossRef, DBLP, Semantic Scholar, etc.):
The agent would query these authoritative databases to find publications, verify bibliographic details (authors, title, year, venue), retrieve DOIs, abstracts, citation counts, and potentially access full text for contextual verification.

- **Google Scholar:** While a powerful search tool, Google Scholar does not offer an official, robust API for programmatic access.[35] Existing methods rely on web scraping or third-party APIs (e.g., Scrapingdog, SerpAPI), which can be less stable, more costly, or have limitations for large-scale, reliable agentic interaction.[35] It does support export in formats like BibTeX and EndNote.[37]
- **Web of Science (WoS):** As a leading citation database, WoS provides APIs (e.g., Web of Science Starter API, and more comprehensive APIs for subscribers) that allow programmatic access to its Core Collection for document metadata, citation counts, author profiles, and journal impact data.[39] This makes WoS a highly valuable and reliable tool for an agent.
- **PubMed:** Frequently cited as a trusted database for medical literature [4], PubMed offers robust APIs (e.g., Entrez Utilities) for searching and retrieving biomedical literature metadata, making it an essential resource for agents working in health sciences.
- **CrossRef & DBLP:** CrossRef is fundamental for DOI registration and metadata lookup, offering a REST API crucial for verifying DOIs and retrieving metadata.[43] DBLP Computer Science Bibliography provides extensive bibliographic information for computer science and offers a SPARQL endpoint and other API access, as demonstrated by its use in systems like GPTscholar.[44] These are indispensable for an agent focused on computer science literature.
- **Semantic Scholar:** Offers APIs (e.g., title-search API) that can be leveraged by agents, as seen in the Ai2 Paper Finder, for semantic search and citation tracking.[45]

Workflow: How an Agentic System Would Generate and Validate References

An agentic citation system would likely follow a multi-step workflow:

1. **Input & Initial Generation:** The user provides text requiring a citation, or asks the agent to find and cite sources for a specific claim. The agent's core LLM might make an initial attempt to generate the citation or identify candidate papers.
2. **Information Extraction & Query Formulation:** The agent analyzes the user's request or the LLM's initial output to extract key entities (e.g., author names, title keywords, publication year, topic). It then formulates structured queries for relevant tools.

3. **Tool Selection & Execution:** Based on the query and available tools, the planning module selects the most appropriate tools (e.g., Mendeley API to check local library, Web of Science API for verification, CrossRef API for DOI lookup). The agent executes these tools.
4. **Data Retrieval & Reconciliation:** The agent retrieves bibliographic data from one or more sources. This data may be incomplete or conflicting (e.g., different author lists from different databases). The agent must then employ strategies to reconcile these discrepancies, potentially by prioritizing more authoritative sources or using fuzzy matching techniques.[46]
5. **Verification:** The agent verifies the existence and accuracy of the publication. This includes validating the DOI, cross-referencing metadata across multiple databases, and potentially checking if the content of the paper (if accessible via full text or abstract) supports the claim it is being cited for.
6. **Interaction with User's Bibliography Software:** The agent queries the user's connected bibliographic software (e.g., Zotero library via Zotero API) to check if the verified reference already exists. If it does, the agent can use the existing entry. If not, and with user permission, it can add the newly verified and structured reference to the user's library.
7. **Formatting & Output:** Using the verified and reconciled metadata, the agent formats the citation according to the user's required style (e.g., APA, MLA, Vancouver) or generates a complete BibTeX entry. This step could involve an internal LLM prompt or, more robustly, interaction with a Citation Style Language (CSL) processor.[48]
8. **Self-Correction/Refinement & User Interaction:** If verification fails at any step, or if ambiguities remain that the agent cannot resolve, it can re-evaluate its plan, try alternative tools or queries, or present the issue and options to the user for clarification or decision.[4] This iterative refinement is a hallmark of agentic behavior.

This systematic approach, combining the LLM's language capabilities with the structured data access and verification afforded by external tools, aims to significantly reduce the incidence of hallucinated or inaccurate citations.

The integration of these diverse bibliographic sources and tools presents a notable "bibliographic data impedance mismatch." LLMs primarily process and generate unstructured natural language, while bibliographic databases and software APIs deal with highly structured, often heterogeneous, and sometimes inconsistent data formats. An agentic framework must therefore incorporate sophisticated data mapping, normalization, and conflict resolution strategies to bridge this gap

effectively. This is not merely about making API calls, but about managing a complex data integration and transformation pipeline. The current fragmented landscape of APIs for these tools—ranging from well-documented and robust (like Mendeley, Zotero, Web of Science) to limited or non-existent (like a public, comprehensive EndNote API or an official Google Scholar API)—poses a significant hurdle. A truly scalable and universally effective agentic solution might necessitate or drive the development of more standardized "bibliographic agent" APIs, offering common functionalities across platforms and simplifying agent development.

Beyond simple retrieval and formatting, an advanced agentic system could evolve into a "citation quality guardian." This would involve not only verifying the existence of a source but also assessing its appropriateness and quality in context—evaluating if it is peer-reviewed, from a reputable venue, current enough for the claim, or if it genuinely supports the assertion being made. This requires deeper semantic understanding and access to qualitative metadata, such as journal impact factors or peer-review status available in databases like Web of Science.[39]

The following table provides an overview of key bibliographic software and academic databases, assessing their potential for integration into such an agentic framework.

Table 1: Overview of Bibliographic Software and Academic Database APIs for Agentic Integration

Tool/Database	API Availability/Type	Key Data Provided	Potential for Agentic Interaction	Known Limitations/Challenges
EndNote	Limited public API for deep library access; specific integrations exist (e.g., Word plugin via AWS) [27]	Metadata, Cite-While-You-Write functionality	Low to Medium	Lack of a clear, comprehensive public API for general agentic interaction; reliance on specific integrations or intermediary libraries.[29]
Mendeley	Yes (Core API Resources, Datasets API	Metadata, PDFs (user-authorized), annotations,	High	API changes have occurred (e.g., retiring

	Resources via Mendeley Developer Portal, SDKs) [30]	folders, group libraries (public & private)		profile search) [30]; requires user OAuth for private library access.
Zotero	Yes (Zotero API, PyZotero library, LangChain ZoteroRetriever) [33]	Metadata, full-text links, attachments, collections, tags, group libraries (public & private with API key)	High	API search behavior may require LLM to refine queries for optimal results when used in agentic frameworks.[34]
Google Scholar	No official API; web scraping or third-party APIs (e.g., Scholarly, Scrapingdog, SerpAPI) [35]	Metadata, abstracts, citation counts, links to full text, BibTeX/EndNote /RefMan export [37]	Low to Medium	Unofficial methods can be unstable, costly, or subject to rate limits/blocking; reliability for large-scale agentic use is questionable.[35]
Web of Science	Yes (WoS Starter API, Expanded API, Article Match Retrieval API, etc., via Clarivate Developer Portal) [41]	Comprehensive metadata, citation counts, author profiles, journal metrics, DOIs, abstracts, full-text links	High	Requires subscription for full access; API keys needed; different APIs for different levels of data access.[42]
PubMed	Yes (Entrez Utilities E-utils API) [4]	Biomedical literature metadata, abstracts, MeSH terms, full-text links (PubMed Central)	High	Primarily focused on biomedical domain; requires understanding of Entrez query syntax.

CrossRef	Yes (REST API) [43]	DOI registration and resolution, extensive metadata (authors, titles, publication dates, venue, abstracts, funding)	High	Primarily for DOI-centric lookups and metadata retrieval; not a discovery search engine like Google Scholar or WoS.
DBLP	Yes (SPARQL endpoint, XML data dumps, search API) [44]	Computer science bibliographic data (authors, titles, venues, year, DOIs, BibTeX)	High	Primarily focused on computer science; SPARQL querying can be complex for an LLM to generate reliably without careful prompting and validation.[44]
Semantic Scholar	Yes (API for paper search, author details, citation graph, etc.) [45]	Paper metadata, abstracts, citation data, author information, influential citation counts, TLDRs	High	API has rate limits; data coverage might vary compared to WoS or Scopus for some fields.

This table underscores the variable landscape of API accessibility and functionality, which is a central factor in determining the practical feasibility and robustness of the proposed agentic solution.

5. Comparative Analysis: Agentic Frameworks vs. Alternative Approaches

While an agentic framework integrating bibliographic software and academic databases presents a comprehensive vision, several alternative or complementary approaches exist for tackling LLM citation inaccuracies. Each offers distinct mechanisms, strengths, and weaknesses.

Retrieval Augmented Generation (RAG) and its Variants

- **Core Mechanism:** RAG enhances LLMs by first retrieving relevant information from an external knowledge source (e.g., a vector database of research papers, academic database APIs) and then providing this information as context to the LLM when generating a response.[50] This grounds the LLM's output in factual data, reducing hallucinations and allowing access to information beyond its training cut-off.[4] RAG systems can be programmed to cite the sources used in generation.[50]
- **Application to Citations:** For citations, RAG can retrieve actual bibliographic metadata, abstracts, or even full-text snippets from indexed scientific documents or by querying academic databases directly.[52] This retrieved data then informs the LLM's generation of the citation and surrounding text. For example, a RAG system could extract BibTeX data directly from PDFs using tools like GROBID and then use this structured data to answer queries or generate bibliographies.[56]
- **Corrective RAG (CRAG):** CRAG enhances RAG by adding a retrieval evaluation step. If retrieved documents are deemed irrelevant or of low quality by an evaluator (which can be another LLM), CRAG can trigger corrective actions, such as reformulating the query or performing a web search to find better sources before generation.[62] This could be useful if an initial database lookup for a specific paper fails or returns ambiguous results.
- **Self-Reflective RAG (Self-RAG):** Self-RAG trains an LLM to adaptively decide *when* to retrieve information and to generate "reflection tokens" that critique its own output and the relevance/supportiveness of retrieved passages.[68] This allows for more fine-grained control over the generation process and provides explicit citations with a self-assessment of their validity, enhancing verifiability. Self-RAG has shown significant gains in improving factuality and citation accuracy for long-form generations.[68]
- **ScholarCopilot:** This is an example of an agentic RAG framework specifically designed for academic paper writing. It iteratively integrates text generation with citation retrieval by having the LLM generate a special `` token when a citation is needed. This pauses generation, triggers retrieval from a citation database, and then feeds the retrieved reference content back to the LLM to continue generation. It is designed to handle BibTeX output and uses contrastive learning to optimize its retrieval mechanism.[55]
- **Strengths:** Provides access to up-to-date and domain-specific information, reduces hallucinations by grounding responses, can provide source attribution.
- **Weaknesses:** Performance heavily depends on the quality and relevance of retrieved documents ("garbage in, garbage out"). Retrieval itself can be

challenging for complex queries. Basic RAG doesn't inherently verify the retrieved information's correctness, only its relevance to the query.

Fine-tuning LLMs

- **Core Mechanism:** This involves further training a pre-trained LLM on a smaller, curated dataset specific to a particular task or domain.[53] The goal is to adapt the LLM's knowledge, style, or behavior.
- **Application to Citations:** LLMs can be fine-tuned on large datasets of scientific papers paired with their correct, structured bibliographic metadata (e.g., BibTeX entries, CSL-JSON).[74] This could teach the LLM to better recognize patterns in bibliographic data and generate more accurate and properly formatted citations directly. Fine-tuning can also focus on specific sub-tasks like "citation intent classification" (understanding why a citation is made) [77] or "citation faithfulness detection" (verifying if a claim is supported by a citation).[78] For example, ChIP-GPT fine-tunes Llama models to extract and summarize metadata from the Sequence Read Archive.[75] Research also explores fine-tuning for generating BibTeX from scientific papers using arXiv data and extracted BibTeX.[76]
- **Datasets:** The success of fine-tuning heavily relies on the availability of high-quality, diverse, and large-scale training datasets. Creating such datasets for bibliographic information, covering numerous styles and edge cases, is a significant challenge.[79] While general-purpose instruction fine-tuning datasets exist [82], specialized, comprehensive BibTeX generation datasets are not readily available.
- **Strengths:** Can improve the LLM's inherent ability to handle specific tasks like citation formatting, potentially reducing the need for extensive retrieval for common cases. Can adapt the LLM to specific stylistic requirements.
- **Weaknesses:** Can be resource-intensive (data collection, computation for training).[84] Knowledge remains static after fine-tuning unless retrained. Risk of "catastrophic forgetting" where fine-tuning on a narrow task degrades performance on other tasks.[85] Performance is highly dependent on the quality and representativeness of the fine-tuning dataset.

Hybrid LLM-Symbolic Systems

- **Core Mechanism:** These systems combine the pattern-recognition and natural language strengths of LLMs with the precision, rule-following capabilities, and verifiability of symbolic AI methods (e.g., formal logic, rule engines, parsers).[15]
- **Application to Citations:**
  - **Validation/Generation using Formal Grammars:** LLMs could generate citation data (e.g., in a preliminary structured format or as natural language

descriptions of bibliographic fields), which is then validated or transformed by a symbolic parser adhering to a formal grammar for BibTeX or CSL-JSON.[90] Conversely, a grammar could guide the LLM's generation process to ensure syntactically correct output.
    - **CSL Processing:** An LLM could extract or generate raw bibliographic metadata, which is then passed to a Citation Style Language (CSL) processor (like citeproc-js [48] or citeproc-lua [92]) for formatting according to specific citation styles.[93] The LLM would handle metadata extraction and content understanding, while the CSL engine enforces strict formatting rules.
    - **HybridMind Strategy:** This approach involves an LLM meta-selecting whether to use natural language reasoning or symbolic language reasoning (e.g., generating Python code or First-Order Logic) for a given problem.[86] For citations, this could mean choosing between direct LLM generation for simple cases versus generating a structured representation for a symbolic formatter for complex styles or validation tasks.
- **Strengths:** Symbolic components can enforce strict structural correctness and adherence to formatting rules, areas where LLMs often fail. Offers greater verifiability and interpretability for the rule-based parts.
- **Weaknesses:** Developing and integrating symbolic components can be complex. Symbolic systems can be brittle and may not handle ambiguity or variation as well as LLMs. Defining comprehensive formal grammars for all bibliographic nuances is challenging.

Knowledge Graphs (KGs)

- **Core Mechanism:** KGs represent information as a network of entities (e.g., papers, authors, venues) and their relationships (e.g., CITES, HAS_AUTHOR, PUBLISHED_IN).[87] LLMs can be used to construct KGs from unstructured text (like research papers) and can also query KGs to retrieve structured, factual information.[96]
- **Application to Citations:** A bibliographic KG can serve as a highly structured and verifiable source of truth for citation details. LLMs can query this KG (e.g., by generating SPARQL or Cypher queries from natural language) to retrieve metadata for a specific paper or to verify the components of a citation.[44] For instance, GPTscholar uses an LLM to generate SPARQL queries for the DBLP KG to retrieve academic literature.[44] The KGT framework uses LLMs to reason on KG schemas to mitigate hallucinations.[108]
- **Schema Design:** A comprehensive bibliographic KG schema would include entities for papers, authors (with variants), affiliations, venues (journals, conferences, books with IDs), publishers, and explicit relationships like cites,

is_authored_by, is_affiliated_with, published_in, along with attributes for all standard BibTeX fields (year, volume, issue, pages, DOI, URL, abstract, keywords, etc.).[87]

- **Strengths:** Provide a structured, verifiable, and interconnected source of bibliographic data. Can help resolve ambiguities (e.g., author disambiguation) and uncover relationships. Facilitate complex queries that span multiple entities.
- **Weaknesses:** KG construction is a complex and potentially costly process, especially at scale.[109] Keeping KGs up-to-date with the rapidly growing body of scientific literature is a major challenge. LLM-generated queries for KGs can still be error-prone.

Post-processing and Validation Tools

- **Core Mechanism:** These are external tools or secondary LLM checks designed to validate, correct, or complete citations generated by a primary LLM or extracted from other sources.[52]
- **Application to Citations:**
  - **CiteFix:** Employs methods like keyword matching, semantic context matching, BERTScore, and LLM-based matching to cross-check LLM-generated factual claims and their attributed citations against the content of retrieved source documents in a RAG system. It has shown to improve citation accuracy significantly.[52]
  - **Semantic Citation Validation Tool:** Uses NVIDIA NIM microservices and fine-tuned models to perform deep semantic analysis of claims against referenced texts, classifying citations as supported, partially supported, unsupported, or uncertain.[112]
  - **SourceCheckup:** An automated agent-based pipeline that evaluates the relevance and supportiveness of sources cited in LLM responses, particularly for medical queries.[115]
  - **Other Tools:** Tools like Originality.ai offer fact-checking capabilities.[116] CiteAssist automates BibTeX generation from preprints and embeds them in PDFs.[113] Bibtex Autocomplete uses online databases (via DOI or title) to complete fields in existing BibTeX entries.[117]
- **Strengths:** Can catch and correct errors made by the primary LLM. Can be computationally lighter than full agentic reasoning for every citation if applied selectively.
- **Weaknesses:** Adds an extra step to the workflow. The accuracy of the post-processing tool itself becomes critical. May not catch all types of errors, especially subtle semantic ones or completely fabricated (but plausible)

references if the tool relies on the same limited knowledge base.

The various approaches to improving LLM citation accuracy—RAG, fine-tuning, symbolic methods, KGs, and post-processing—each possess unique strengths but also inherent limitations. This suggests that the most robust and effective solutions will likely emerge from hybrid systems that strategically combine these techniques. For example, an LLM fine-tuned for basic bibliographic awareness could operate within an agentic RAG framework to access external databases and tools, leverage a KG for verifying structured data, and utilize a symbolic CSL processor for precise style enforcement. Such convergence is already hinted at in discussions comparing RAG and fine-tuning, where a combined approach is often recommended for optimal performance.[53]

However, each layer of sophistication adds to the "cost of accuracy." Simple RAG might be relatively inexpensive to implement compared to the extensive data curation and computational resources required for deep fine-tuning or the construction and maintenance of a comprehensive bibliographic knowledge graph.[52] Complex agentic systems with multiple specialized tools and LLM calls also incur significant development and operational overhead.[24] Therefore, the selection and combination of these methods will necessitate a careful trade-off analysis based on the desired level of citation accuracy versus available resources and acceptable complexity.

Furthermore, as these hybrid systems evolve, the concept of a definitive "truth source" for citations becomes multifaceted. Is the ultimate authority the LLM's fine-tuned knowledge, the content of documents retrieved via RAG, the structured assertions within a KG, or the output of a rule-based CSL processor? These sources may occasionally conflict, and a mature agentic system will require robust strategies for reconciling these discrepancies, potentially involving confidence scoring, source prioritization, or even human arbitration.[46]

Table 2: Comparative Overview of LLM Citation Improvement Approaches

Approach	Core Mechanism	Key Strengths for Citation Accuracy	Key Weaknesses/Challenges for Citation Accuracy	Estimated Impact on Citation Accuracy	Key Supporting Information

Agentic Framework with Tools	LLM plans and uses external bibliographic software (EndNote, Zotero, Mendeley) & academic databases (WoS, PubMed, CrossRef) for verification & generation.	Comprehensive verification, access to curated user libraries & authoritative databases, potential for automation of full workflow, adaptability.	High development complexity, API dependency & variability, potential latency, data reconciliation from multiple sources, cost.	High	3
Retrieval Augmented Generation (RAG) - Basic	LLM retrieves information from an external knowledge base (e.g., indexed papers) before generating citations.	Access to up-to-date/domain-specific info, reduces hallucination by grounding, can cite sources.	Dependent on retrieval quality, retrieved info may still be inaccurate or incomplete, basic RAG doesn't deeply verify.	Medium to High	50
Corrective RAG (CRAG)	RAG with a retrieval evaluator; corrects/augments retrieval (e.g., via web search) if initial documents are poor.	Improves robustness of RAG if initial retrieval is weak, adaptive knowledge sourcing.	Adds complexity and latency to RAG, web search quality can vary.	High	62
Self-Reflective RAG (Self-RAG)	LLM adaptively retrieves,	Enhanced factuality & citation	Requires specialized LLM training,	High	68

	generates reflection tokens to critique its own output & support from retrieved passages, provides citations with self-assessment.	accuracy, adaptive retrieval, verifiable outputs with support scores.	complexity in managing reflection tokens.		
Fine-tuning	Training a pre-trained LLM on domain-specific datasets of papers and correct citations (e.g., BibTeX).	Improves LLM's inherent ability for specific citation formats/styles, better understanding of bibliographic nuances.	Requires large, high-quality curated datasets (often scarce for citations), knowledge becomes static, risk of overfitting or catastrophic forgetting.	Medium to High	[73]
Hybrid LLM-Symbolic Systems	Combining LLM NLU with symbolic reasoners (e.g., parsers, CSL processors, formal grammars) for structure & rule enforcement.	Precision in formatting, adherence to strict syntax (e.g., BibTeX, CSL), verifiable rule-based processing.	Integration complexity, symbolic systems can be brittle, defining comprehensive grammars is hard.	High (for formatting/structure)	[15]

Knowledge Graphs (KGs)	Representing bibliographic data as entities & relations; LLMs query KG or use it for verification.	Provides structured, verifiable factual backbone, helps resolve ambiguities, supports complex queries.	KG construction & maintenance is costly & complex, LLM-KG querying can be error-prone.	High (for data verification)	44
Post-processing & Validation Tools	External tools or secondary LLM checks to validate, correct, or complete LLM-generated citations.	Catches residual errors, can be computationally lighter if applied selectively, specific tools for specific error types (e.g., CiteFix, Semantic Citation Validation).	Adds workflow step, accuracy depends on the tool itself, may not catch all error types.	Medium to High	52

6. Critical Review of the Agentic Framework Solution

The proposal to use an agentic framework, integrating LLMs with bibliographic software and academic databases, holds considerable promise for addressing the pervasive issue of citation inaccuracies. However, a thorough assessment reveals both significant strengths and formidable challenges.

Strengths and Potential Benefits:

1. **Improved Accuracy and Reliability:** The primary allure of an agentic framework is its potential to significantly enhance citation accuracy. By enabling the LLM to iteratively query authoritative databases like Web of Science or PubMed, validate DOIs via CrossRef, cross-reference metadata, and even consult the user's own curated EndNote, Mendeley, or Zotero libraries, the system can move beyond probabilistic generation towards fact-based citation construction.[4] Platforms like Microsoft Discovery already demonstrate how specialized AI agents combined

with knowledge engines can drive outcomes with improved accuracy by reasoning over complex, contextual graphs and providing detailed source tracking.[3]

2. **Automation of Literature Management Tasks:** Beyond mere citation generation, an agentic system could automate many tedious aspects of literature review and bibliography management. This includes searching for relevant papers based on user queries, retrieving full metadata, checking for updates to existing references, ensuring consistent formatting across a document, and organizing references within the user's bibliographic software.[19]
3. **Enhanced Verifiability and Trust:** By providing clear provenance for each piece of bibliographic information (i.e., indicating which database or tool verified a specific field) and employing self-correction mechanisms, agentic systems can make the citation process more transparent. This transparency, coupled with improved accuracy, is crucial for building user trust in LLM-assisted academic work.[3] The ability of an agent to show its "work" in retrieving and verifying a citation is a step towards mitigating the black-box nature of LLM outputs.
4. **Adaptability and Learning:** Agentic frameworks can be designed to learn from interactions and feedback. If a user corrects a citation generated by the agent, or if an automated check flags an error that the agent initially missed, this feedback can be used to refine the agent's future strategies, tool selection, or query formulations.[20] This adaptive capability is key for handling the evolving landscape of scientific publishing and user preferences.

Weaknesses, Challenges, and Limitations:

1. **Technical Feasibility of Robust API Integrations:** The entire agentic framework hinges on seamless and reliable interaction with a multitude of external systems, each with its own API (or lack thereof).
    - **API Availability and Stability:** As detailed in Table 1 (Section 4), the API landscape for bibliographic tools and databases is highly variable. While some (Mendeley, Zotero, Web of Science) offer relatively mature APIs [30], others like EndNote have less clear public API access for deep library integration [27], and crucial resources like Google Scholar lack official, robust APIs, forcing reliance on potentially unstable web scraping or costly third-party services.[35] The functionality, reliability, and terms of service for these APIs can also change, posing ongoing maintenance challenges. This "chain of dependency" makes the agentic system vulnerable: the failure or degradation of a single critical API could cripple the entire citation workflow.
    - **Authentication and Authorization:** Securely managing user credentials and permissions for accessing private bibliographic libraries (e.g., a user's Zotero

library) and subscription-based academic databases is a complex security and implementation challenge.
    - **Rate Limits and Usage Costs:** Many APIs impose rate limits on requests and may have associated costs, especially for commercial databases or high-volume LLM calls for verification and reasoning steps.[36] This could make the agentic solution expensive to operate at scale.
2. **Complexity and Overhead of Agentic Systems:**
    - **Development Complexity:** Designing, building, and rigorously testing a multi-component agentic system that orchestrates an LLM with numerous tools, manages state, plans, and self-corrects is significantly more complex than developing a standalone LLM application or a simpler RAG pipeline.[24] Workflows can rapidly grow in complexity, becoming difficult to manage and debug.[121]
    - **Computational Overhead & Latency:** Each step in the agentic workflow—LLM calls for planning or query generation, API calls to external databases, data processing, verification logic—adds computational overhead and latency.[24] A multi-step verification process for a single citation could become unacceptably slow for users accustomed to near-instantaneous LLM responses.
    - **Maintenance:** The numerous interconnected components (LLM versions, API connectors, tool logic, database schemas) require continuous monitoring and maintenance to ensure they remain compatible and functional.
3. **Data Consistency and Reconciliation:** Academic databases and user libraries often contain conflicting, incomplete, or inconsistently formatted bibliographic data for the same publication (e.g., variations in author name spellings, different publication dates, missing DOIs). The agent must incorporate sophisticated logic for data reconciliation, disambiguation, and merging, which is a non-trivial data quality problem.[14] The "semantic gap" in tool orchestration is also a concern: an LLM might correctly identify the need to use a tool (e.g., a DOI lookup) but may misunderstand the tool's specific parameters, output format, or error messages, leading to incorrect tool usage and flawed results, even if the API itself is functional.
4. **Scalability:** The sheer volume of scientific literature and the potential number of users and requests pose a significant scalability challenge. Ensuring that the agentic system can perform efficiently and cost-effectively as usage grows requires careful architectural design and resource management.[24]
5. **Security, Privacy, and Bias Concerns:**
    - **Data Privacy:** When an agent accesses a user's private bibliographic library or queries databases based on user input, sensitive information may be

processed. Ensuring data privacy, secure handling of credentials, and compliance with regulations like GDPR is paramount, especially when data is shared with multiple third-party tools or APIs.[21]
    - **Inherited Bias:** The agentic system can inherit biases from its core LLM (trained on potentially biased web data) or from the external databases it queries. For example, if a database over-represents publications from certain regions or in certain languages, the agent's suggestions might reflect this bias.[1]
6. **Cost Implications:** The cumulative costs of development, ongoing maintenance, API subscription fees (for some databases), LLM inference for reasoning and generation, and computational resources for agentic processing can be substantial.[118]
7. **Risk of Over-engineering:** For simple citation tasks or by users who only need a quick, approximate reference, a full-fledged agentic framework might represent an over-investment in complexity and resources if simpler methods like a well-prompted LLM with basic RAG, or a dedicated fine-tuned model, could suffice with "good enough" accuracy for those specific needs.[24] The potential for "automation bias amplification" also exists: if an agent, perceived as highly intelligent, produces a citation that is well-formatted (e.g., via a CSL processor) but factually incorrect, researchers might be less inclined to manually verify it compared to a clearly malformed LLM output, thereby increasing the risk of error propagation.[1]

Table 3: Critical Evaluation of Agentic Framework for Scientific Citation

Aspect	Potential of Agentic Framework	Key Challenges/Limitations	Illustrative Supporting Information
Accuracy Improvement Potential	High; through multi-source verification, DOI validation, contextual checks, and interaction with curated databases/libraries.	Dependency on quality/availability of external source APIs; complexity of data reconciliation; LLM's ability to correctly interpret tool outputs.	[3]
Scalability for	Medium to High; can query multiple	API rate limits; processing speed for	[24]

Diverse Literature	databases covering vast literature.	numerous sources; cost of extensive querying; keeping KG/indexes updated.	
Integration Feasibility with Existing Tools	Variable; Good for tools with robust APIs (Mendeley, Zotero, WoS, CrossRef). Poor for tools with limited/no public APIs (EndNote, Google Scholar).	API fragmentation; authentication complexities; maintenance of diverse integrations.	[27]
Cost of Development & Operation	High; complex system design, multiple LLM calls, potential API fees, ongoing maintenance.	Significant initial investment and recurring operational expenses.	[24]
Maintainability & Upgradability	Medium; modular design can help, but dependent on stability of many external APIs and LLM versions.	Changes in external APIs or LLM behavior can break integrations; requires continuous monitoring.	[24]
Data Privacy & Security	Medium; depends on implementation. Can be designed with privacy in mind (e.g., local processing where possible, secure API calls).	Risks from sharing user data with multiple third-party tools; GDPR compliance; secure credential management.	[21]
Bias Mitigation Capabilities	Medium; can be designed to query diverse sources, but may inherit biases from LLM or queried databases.	Difficult to eliminate all sources of bias; requires careful selection of data sources and potentially bias detection mechanisms.	[1]

User Trust & Adoption	Medium to High; if proven reliable and transparent, could be highly adopted. Initial skepticism likely.	Requires demonstrating consistent accuracy and providing explainable outputs to overcome automation bias.	[1]

In conclusion, while the agentic framework offers a theoretically powerful solution to LLM citation inaccuracies, its practical realization is fraught with significant technical, operational, and financial challenges. Its success will depend on careful architectural design, the evolution of API ecosystems, and a balanced approach to automation and human oversight.

7. Recommendations and Future Directions

Addressing the challenge of LLM-generated citation inaccuracies requires a multi-faceted approach. While agentic frameworks offer a promising path, their successful development and deployment necessitate adherence to best practices and continued research into overcoming current limitations.

Best Practices for Developing and Deploying Agentic Citation Systems:

1. **Modular Design:** Architect the agentic system with clear separation of concerns. Different agents or modules should handle specific sub-tasks such as query understanding, tool selection, metadata extraction from specific databases (e.g., a PubMed agent, a CrossRef agent), DOI verification, interaction with local bibliography software, formatting via CSL processors, and conflict reconciliation.[19] This modularity facilitates development, testing, maintenance, and future upgrades of individual components.
2. **Robust Error Handling and Fallback Mechanisms:** Given the reliance on external APIs and the potential for incomplete or conflicting data, agents must have sophisticated error handling. This includes retrying failed API calls, implementing timeouts, and having predefined fallback strategies (e.g., if a primary database is unavailable, query a secondary one; if DOI verification fails, attempt title/author search).
3. **Prioritize Verifiable and Authoritative Sources:** Design agents to preferentially query and trust information from high-authority databases like Web of Science, PubMed, CrossRef, and established institutional repositories. The system should have a mechanism to weigh the reliability of different sources when reconciling conflicting information.

4. **Transparency and Explainability in Operation:** For users to trust the agent's output, the system should be able to provide an "audit trail" or explanation of its actions. This could involve logging which sources were queried, what information was retrieved, how conflicts were resolved, and how the final citation was constructed. Techniques like Self-RAG's reflection tokens [68] or the source tracking in systems like Microsoft Discovery [3] offer models for such "explainable agency." This transparency is vital for debugging, building user confidence, and allowing users to understand the provenance of a citation.
5. **User-in-the-Loop (Human-AI Collaboration):** Especially in the early stages of deployment and for critical applications like scholarly publishing, incorporate mechanisms for human review, correction, and feedback.[4] The agent should be able to flag ambiguous cases or low-confidence results for user intervention. User corrections can also serve as valuable data for retraining or fine-tuning the agent's decision-making processes.
6. **Continuous Evaluation and Monitoring:** Implement a rigorous and ongoing evaluation framework to benchmark the system's performance on citation accuracy, completeness, formatting correctness, latency, and robustness against diverse inputs and edge cases.[128] Standardized metrics and challenging test datasets are essential for tracking improvements and identifying regressions.

Areas for Further Research:

1. **Standardized APIs for Bibliographic Tools and Databases:** The current fragmented and often limited API landscape for bibliographic software and academic databases is a major bottleneck.[27] Research and advocacy efforts should be directed towards the development of open, comprehensive, and standardized APIs that offer common functionalities (e.g., advanced search, metadata retrieval, entry addition/modification, style formatting) across different platforms. This would significantly simplify agent development and enhance interoperability.
2. **Benchmark Datasets for Scientific Citation Accuracy:** There is a pressing need for large-scale, high-quality, and diverse benchmark datasets specifically designed for training and evaluating LLMs and AI agents on scientific citation generation, verification, and formatting tasks. These datasets should cover a wide range of disciplines, citation styles (including BibTeX and CSL-JSON outputs), and include examples of common errors and edge cases.[76] The CiteCheck dataset for Chinese citation faithfulness is a step in this direction.[78]
3. **Advanced Hybrid and Composable Models:** Future research should explore more tightly integrated hybrid systems. This could involve agentic frameworks that orchestrate fine-tuned LLMs (for improved baseline generation and domain

understanding), Retrieval Augmented Generation (for accessing current and external knowledge), Knowledge Graphs (for structured factual verification and relationship discovery), and symbolic processors (like CSL engines for precise style enforcement). The development of "citation style agents"—smaller, specialized models or symbolic engines expert in particular complex citation styles (e.g., legal or highly specific journal styles)—that can be invoked by a primary agent, is a promising direction.

4. **Proactive Error Detection and Prevention:** Current approaches often focus on post-hoc correction. Research into agents that can proactively identify potential citation errors based on contextual ambiguity, query vagueness, or known LLM weaknesses could lead to more efficient and reliable systems. This might involve the agent prompting the user for clarification before attempting a difficult citation task.
5. **Agent-Driven Discovery of Novel or Obscure References:** While the primary goal is accuracy, a sophisticated agent with broad access to diverse databases (including full-text repositories) and advanced semantic search capabilities could also assist researchers in discovering relevant but "long-tail" or serendipitous references that traditional search methods might miss. This would elevate the agent from a mere citation manager to a genuine research discovery assistant.
6. **Ethical Guidelines and Governance for Agentic Scientific Assistants:** As AI agents become more autonomous and integrated into the scientific process, clear ethical guidelines and governance frameworks are needed. These should address issues of accountability, bias, data privacy, and the potential for misuse in academic contexts.

The Indispensable Role of Human Oversight:

It is crucial to underscore that while agentic systems and other AI-driven solutions can significantly augment and automate the process of scientific referencing, human expertise and critical judgment remain indispensable. LLMs and AI agents are tools, and like any tool, they can be misused or produce flawed outputs. In the context of scholarly publishing, where accuracy, intellectual honesty, and the integrity of the scientific record are paramount, final validation by human researchers is non-negotiable. The most effective path forward lies in developing robust human-AI collaborative workflows, where AI agents handle the laborious tasks of searching, retrieving, and initial formatting, while human researchers provide the critical oversight, contextual understanding, and final approval.

8. Conclusion

The increasing integration of Large Language Models into scientific research workflows presents both unprecedented opportunities and significant challenges. Among the most critical challenges is the propensity of LLMs to generate inaccurate, incomplete, or entirely fabricated scientific references, a phenomenon that directly undermines the credibility of AI-assisted scholarship and the foundational principles of scientific verifiability. The impact of such errors is far-reaching, potentially leading to the propagation of misinformation, the erosion of trust in scientific findings, and the misdirection of research efforts.

This report has investigated the root causes of these LLM shortcomings, attributing them to their probabilistic generation mechanisms, limitations inherent in their training data, and their difficulties in handling the precise, structured nature of bibliographic information. As a potential solution, the concept of an agentic framework—wherein an LLM is augmented with the ability to plan, use external tools (bibliographic software like EndNote, Mendeley, Zotero), and query authoritative academic databases (such as Web of Science, PubMed, CrossRef)—has been critically evaluated.

The analysis indicates that such an agentic framework offers a theoretically robust pathway toward significantly improving scientific reference accuracy. By enabling iterative verification, cross-referencing against multiple trusted sources, and leveraging the specialized functionalities of existing bibliographic tools, these agents could automate and enhance the reliability of citation generation and management. However, the practical realization of this vision is contingent upon overcoming substantial hurdles. These include the current fragmentation and limitations of APIs for bibliographic software and databases, the inherent complexity and computational overhead of sophisticated agentic systems, the challenges of reconciling inconsistent data from diverse sources, and persistent concerns regarding data privacy, security, and potential biases.

Compared to other approaches like standalone Retrieval Augmented Generation, fine-tuning, symbolic systems, knowledge graphs, or post-processing tools, the agentic framework has the potential to be the most comprehensive by orchestrating these varied techniques. Yet, this comprehensiveness comes at the cost of complexity. Simpler methods may offer partial solutions with lower overhead for specific use cases.

Ultimately, the journey towards trustworthy LLM-assisted scientific research requires a multi-pronged strategy. This includes continued advancements in LLM architectures and training methodologies, the development of more open and standardized APIs for scientific information resources, the creation of robust benchmark datasets for

evaluating citation accuracy, and the fostering of human-AI collaborative models that leverage the strengths of both. While technology offers powerful tools to mitigate citation errors, the critical judgment and diligent oversight of human researchers will remain indispensable to upholding the integrity and rigor of scientific communication in the age of artificial intelligence.

Works cited

1. How Deep Do Large Language Models Internalize Scientific Literature and Citation Practices? - arXiv, accessed May 23, 2025, https://arxiv.org/html/2504.02767v1
2. Large language models in oncology: a review, accessed May 23, 2025, https://bmjoncology.bmj.com/content/4/1/e000759
3. Transforming R&D with agentic AI: Introducing Microsoft Discovery ..., accessed May 23, 2025, https://azure.microsoft.com/en-us/blog/transforming-rd-with-agentic-ai-introducing-microsoft-discovery/
4. How to Reduce LLM Hallucinations with Agentic AI (Simple ..., accessed May 23, 2025, https://magnimindacademy.com/blog/how-to-reduce-llm-hallucinations-with-agentic-ai-simple-techniques-for-making-large-language-models-more-reliable/
5. LLM hallucinations: Complete guide to AI errors | SuperAnnotate, accessed May 23, 2025, https://www.superannotate.com/blog/ai-hallucinations
6. ijcaonline.org, accessed May 23, 2025, https://ijcaonline.org/archives/volume187/number4/gautam-2025-ijca-924909.pdf
7. Citation Accuracy Challenges Posed by Large Language Models - PMC, accessed May 23, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC12037895/
8. HalluLens: LLM Hallucination Benchmark - arXiv, accessed May 23, 2025, https://arxiv.org/html/2504.17550v1
9. Hallucinations - Artificial Intelligence - Research Guides at Queen's ..., accessed May 23, 2025, https://guides.library.queensu.ca/c.php?g=740510&p=5351519
10. libguides.mit.edu, accessed May 23, 2025, https://libguides.mit.edu/citing#:~:text=It's%20important%20to%20cite%20sources,researchers%20and%20acknowledging%20their%20ideas
11. Citing Sources: What are citations and why should I use them? - Library Guides, accessed May 23, 2025, https://guides.lib.uw.edu/research/citations/citationwhat
12. Are LLMs just predicting the next token? : r/ArtificialInteligence - Reddit, accessed May 23, 2025, https://www.reddit.com/r/ArtificialInteligence/comments/1jo3o69/are_llms_just_predicting_the_next_token/
13. Why Claiming LLMs are merely "next token predictors" is a fundamental misunderstanding : r/singularity - Reddit, accessed May 23, 2025, https://www.reddit.com/r/singularity/comments/199y2xk/why_claiming_llms_are_merely_next_token/

14. Pitfalls of LLMs - Learn Prompting, accessed May 23, 2025, https://learnprompting.org/si/docs/basics/pitfalls
15. Why do LLMs struggle to understand structured data from relational databases, even with RAG? How can we bridge this gap? - Reddit, accessed May 23, 2025, https://www.reddit.com/r/LLMDevs/comments/1ixa80j/why_do_llms_struggle_to_understand_structured/
16. [2402.13284] Structure Guided Large Language Model for SQL Generation - arXiv, accessed May 23, 2025, https://arxiv.org/abs/2402.13284
17. [2401.10186] Beyond Traditional Benchmarks: Analyzing Behaviors of Open LLMs on Data-to-Text Generation - arXiv, accessed May 23, 2025, https://arxiv.org/abs/2401.10186
18. What Did I Do Wrong? Quantifying LLMs' Sensitivity and Consistency to Prompt Engineering - ACL Anthology, accessed May 23, 2025, https://aclanthology.org/2025.naacl-long.73.pdf
19. Agentic Frameworks: A Guide to the Systems Used to Build AI ..., accessed May 23, 2025, https://www.moveworks.com/us/en/resources/blog/what-is-agentic-framework
20. What is agentic AI? - Red Hat, accessed May 23, 2025, https://www.redhat.com/en/topics/ai/what-is-agentic-ai
21. What is Agentic AI? | UiPath, accessed May 23, 2025, https://www.uipath.com/ai/agentic-ai
22. Top Agentic LLM Frameworks for Smarter AI Automation, accessed May 23, 2025, https://www.bacancytechnology.com/blog/agentic-llm-frameworks
23. Agentic Retrieval-Augmented Generation: A Survey on Agentic RAG - arXiv, accessed May 23, 2025, https://arxiv.org/html/2501.09136v1
24. Are Agentic Frameworks an Overkill? - Blog, accessed May 23, 2025, https://blog.premai.io/are-agentic-frameworks-an-overkill-benefits-challenges-and-alternatives/
25. LLM Agent Evaluation: Assessing Tool Use, Task Completion ..., accessed May 23, 2025, https://www.confident-ai.com/blog/llm-agent-evaluation-complete-guide
26. (PDF) Agentic Large Language Models, a survey - ResearchGate, accessed May 23, 2025, https://www.researchgate.net/publication/390354708_Agentic_Large_Language_Models_a_survey
27. How to Use EndNote for Reference Management: A Step-by-Step ..., accessed May 23, 2025, https://editverse.com/how-to-use-endnote-for-reference-management-a-step-by-step-guide/
28. Application Information for EndNote Cite While You Write for Word Online by Clarivate - Microsoft 365 App Certification, accessed May 23, 2025, https://learn.microsoft.com/en-us/microsoft-365-app-certification/word/clarivate-endnote-cite-while-you-write-for-word-online
29. Set Endnote Options | Aspose.Words Document Processing API - Tutorials, accessed May 23, 2025, https://tutorials.aspose.com/words/net/working-with-footnote-and-endnote/set-

endnote-options/
30. Mendeley Developer Portal, accessed May 23, 2025, https://dev.mendeley.com/
31. API Objects - Mendeley Developer Portal, accessed May 23, 2025, https://dev.mendeley.com/overview/core_resources.html
32. Authorization Code Flow - Mendeley Developer Portal, accessed May 23, 2025, https://dev.mendeley.com/reference/topics/authorization_auth_code.html
33. Zotero - LangChain, accessed May 23, 2025, https://python.langchain.com/docs/integrations/providers/zotero/
34. ZoteroRetriever | 🦜 LangChain, accessed May 23, 2025, https://python.langchain.com/docs/integrations/retrievers/zotero/
35. Google Scholar API - Scrapfly, accessed May 23, 2025, https://scrapfly.io/blog/google-scholar-api-and-alternatives/
36. 3 Best Google Scholar APIs To Checkout in 2025 - Scrapingdog, accessed May 23, 2025, https://www.scrapingdog.com/blog/best-google-scholar-apis/
37. Exporting References from Google Scholar - Sourcely, accessed May 23, 2025, https://www.sourcely.net/resources/exporting-references-from-google-scholar
38. How to use Google Scholar: the ultimate guide - Research - Paperpile, accessed May 23, 2025, https://paperpile.com/g/google-scholar-guide/
39. Web of Science Core Collection, accessed May 23, 2025, https://webofscience.help.clarivate.com/Content/wos-core-collection/wos-core-collection.htm
40. Saving and Exporting Marked Lists - Web of Science, accessed May 23, 2025, https://webofscience.zendesk.com/hc/en-us/articles/20135824927505-Saving-and-Exporting-Marked-Lists
41. Resources for Librarians and Administrators: APIs user guides - Clarivate LibGuides, accessed May 23, 2025, https://clarivate.libguides.com/c.php?g=1140539&p=10606994
42. clarivate/wosstarter_python_client: Web of Science Starter API Python Client - GitHub, accessed May 23, 2025, https://github.com/clarivate/wosstarter_python_client
43. RAG-Instruct: Boosting LLMs with Diverse Retrieval-Augmented Instructions. - DBLP, accessed May 23, 2025, https://dblp.org/rec/journals/corr/abs-2501-00353
44. Enhancing LLMs with Knowledge Graphs for Academic ... - SciTePress, accessed May 23, 2025, https://www.scitepress.org/publishedPapers/2024/129905/pdf/index.html
45. Introducing Ai2 Paper Finder | Ai2, accessed May 23, 2025, https://allenai.org/blog/paper-finder
46. How does AI deal with conflicting information? - Milvus, accessed May 23, 2025, https://milvus.io/ai-quick-reference/how-does-ai-deal-with-conflicting-information
47. How does AI deal with conflicting information? - Zilliz Vector Database, accessed May 23, 2025, https://zilliz.com/ai-faq/how-does-ai-deal-with-conflicting-information
48. A JavaScript implementation of the Citation Style Language (CSL) https://citeproc-js.readthedocs.io - GitHub, accessed May 23, 2025,

https://github.com/Juris-M/citeproc-js
49. Research Guides: Citation Indexes: Scopus & Web of Science - Samuel C. Williams Library, accessed May 23, 2025, https://library.stevens.edu/c.php?g=1121889&p=8631431
50. Retrieval-augmented generation - Wikipedia, accessed May 23, 2025, https://en.wikipedia.org/wiki/Retrieval-augmented_generation
51. What Is Retrieval-Augmented Generation aka RAG | NVIDIA Blogs, accessed May 23, 2025, https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/
52. CiteFix: Enhancing RAG Accuracy Through Post-Processing Citation Correction - arXiv, accessed May 23, 2025, https://arxiv.org/html/2504.15629
53. RAG vs. Fine-Tuning: Choosing the Right Approach for Your LLM - Signity Solutions, accessed May 23, 2025, https://www.signitysolutions.com/blog/rag-vs-fine-tuning
54. RAG vs Fine Tuning LLMs: The Right Approach for Generative AI - Aisera, accessed May 23, 2025, https://aisera.com/blog/llm-fine-tuning-vs-rag/
55. ScholarCopilot: Training Large Language Models for Academic Writing with Accurate Citations - arXiv, accessed May 23, 2025, https://arxiv.org/html/2504.00824v1
56. A Retrieval-Augmented Generation Framework for Academic Literature Navigation in Data Science - arXiv, accessed May 23, 2025, https://arxiv.org/html/2412.15404v1
57. RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture - arXiv, accessed May 23, 2025, https://arxiv.org/html/2401.08406v2
58. Using the Retrieval-Augmented Generation to Improve the Question-Answering System in Human Health Risk Assessment: The Development and Application - MDPI, accessed May 23, 2025, https://www.mdpi.com/2079-9292/14/2/386
59. Doing RAG on PDFs using File Search in the Responses API - OpenAI Cookbook, accessed May 23, 2025, https://cookbook.openai.com/examples/file_search_responses
60. accessed December 31, 1969, https://arxiv.org/pdf/2412.15404.pdf
61. arxiv.org, accessed May 23, 2025, https://arxiv.org/pdf/2412.15404
62. Corrective RAG Workflow - LlamaIndex, accessed May 23, 2025, https://docs.llamaindex.ai/en/stable/examples/workflow/corrective_rag_pack/
63. RAG Vs CRAG: Leading The Evolution Of Language Models - CustomGPT.ai, accessed May 23, 2025, https://customgpt.ai/crag-vs-rag-the-evolution-of-rag/
64. arXiv:2401.15884v3 [cs.CL] 7 Oct 2024, accessed May 23, 2025, https://arxiv.org/pdf/2401.15884
65. 6 retrieval augmented generation (RAG) techniques you should know - LogRocket Blog, accessed May 23, 2025, https://blog.logrocket.com/rag-techniques/
66. Citation-Enhanced Generation for LLM-based Chatbots - arXiv, accessed May 23, 2025, https://arxiv.org/html/2402.16063v4
67. accessed December 31, 1969, https://arxiv.org/pdf/2402.16063.pdf
68. arxiv.org, accessed May 23, 2025, https://arxiv.org/html/2310.11511
69. Self-RAG: AI That Knows When to Double-Check - Analytics Vidhya, accessed

May 23, 2025, https://www.analyticsvidhya.com/blog/2025/01/self-rag/
70. Self-RAG: Learning to Retrieve, Generate and Critique through Self-Reflection, accessed May 23, 2025, https://selfrag.github.io/
71. accessed December 31, 1969, https://arxiv.org/pdf/2504.00824.pdf
72. arxiv.org, accessed May 23, 2025, https://arxiv.org/pdf/2504.00824
73. Fine-tuning large language models (LLMs) in 2025 - SuperAnnotate, accessed May 23, 2025, https://www.superannotate.com/blog/llm-fine-tuning
74. (PDF) iTRI-QA: a Toolset for Customized Question-Answer Dataset Generation Using Language Models for Enhanced Scientific Research - ResearchGate, accessed May 23, 2025, https://www.researchgate.net/publication/389314954_iTRI-QA_a_Toolset_for_Customized_Question-Answer_Dataset_Generation_Using_Language_Models_for_Enhanced_Scientific_Research
75. ChIP-GPT: a managed large language model for robust data extraction from biomedical database records | Briefings in Bioinformatics | Oxford Academic, accessed May 23, 2025, https://academic.oup.com/bib/article/25/2/bbad535/7600389
76. Seeing the Forest for the Trees: A Large Scale, Continuously Updating Meta-Analysis of Frontier LLMs - arXiv, accessed May 23, 2025, https://arxiv.org/html/2502.18791v1
77. Can LLMs Predict Citation Intent? An Experimental Analysis of In-context Learning and Fine-tuning on Open LLMs - arXiv, accessed May 23, 2025, https://arxiv.org/html/2502.14561v1
78. arxiv.org, accessed May 23, 2025, https://arxiv.org/abs/2502.10881
79. An introduction to preparing your own dataset for LLM training - AWS - Amazon.com, accessed May 23, 2025, https://aws.amazon.com/blogs/machine-learning/an-introduction-to-preparing-your-own-dataset-for-llm-training/
80. FAIR in LLM Datasets - Data Management for Research - CMU LibGuides, accessed May 23, 2025, https://guides.library.cmu.edu/researchdatamanagement/FAIR_llmdatasets
81. A survey of datasets in medicine for large language models - OAE Publishing Inc., accessed May 23, 2025, https://www.oaepublish.com/articles/ir.2024.27
82. mlabonne/llm-datasets: Curated list of datasets and tools for post-training. - GitHub, accessed May 23, 2025, https://github.com/mlabonne/llm-datasets
83. 10 Datasets for Fine-Tuning Large Language Models aka LLMs - Open Data Science, accessed May 23, 2025, https://opendatascience.com/10-datasets-for-fine-tuning-large-language-models-llm/
84. Improving large language model applications in biomedicine with retrieval-augmented generation: a systematic review, meta-analysis, and clinical development guidelines | Journal of the American Medical Informatics Association | Oxford Academic, accessed May 23, 2025, https://academic.oup.com/jamia/article/32/4/605/7954485
85. Faithfulness and Accuracy: How Fine-Tuning Shapes LLM Reasoning, accessed

May 23, 2025, https://d3.harvard.edu/faithfulness-and-accuracy-how-fine-tuning-shapes-llm-reasoning/
86. HybridMind: Meta Selection of Natural Language and Symbolic Language for Enhanced LLM Reasoning - arXiv, accessed May 23, 2025, https://arxiv.org/html/2409.19381v5
87. Knowledge Graphs and Their Reciprocal Relationship with Large Language Models - MDPI, accessed May 23, 2025, https://www.mdpi.com/2504-4990/7/2/38
88. accessed December 31, 1969, https://arxiv.org/pdf/2409.19381v5.pdf
89. arxiv.org, accessed May 23, 2025, https://arxiv.org/pdf/2409.19381
90. Can LLMs Help Create Grammar?: Automating Grammar Creation for Endangered Languages with In-Context Learning - ACL Anthology, accessed May 23, 2025, https://aclanthology.org/2025.coling-main.681.pdf
91. Grammar Prompting for Domain-Specific Language Generation with... - OpenReview, accessed May 23, 2025, https://openreview.net/forum?id=B4tkwuzeiY¬eId=BaPOkLI42Y
92. Bibliography formatting with citation-style-language - CTAN, accessed May 23, 2025, https://ctan.math.illinois.edu/biblio/citation-style-language/citation-style-language-doc.pdf
93. Primer — An Introduction to CSL — Citation Style Language 1.0.1-dev documentation, accessed May 23, 2025, https://docs.citationstyles.org/en/stable/primer.html
94. Bibiliography style with Quarto documents - Stack Overflow, accessed May 23, 2025, https://stackoverflow.com/questions/75306550/bibiliography-style-with-quarto-documents
95. Running the Processor — citeproc-js 1.1.73 documentation - Read the Docs, accessed May 23, 2025, https://citeproc-js.readthedocs.io/en/latest/running.html
96. Knowledge Graphs with LLMs: Optimizing Decision-Making - Addepto, accessed May 23, 2025, https://addepto.com/blog/leveraging-knowledge-graphs-with-llms-a-business-guide-to-enhanced-decision-making/
97. LLM-Powered Knowledge Graphs for Enterprise Intelligence and Analytics - arXiv, accessed May 23, 2025, https://arxiv.org/html/2503.07993v1
98. Pseudo-Knowledge Graph: Meta-Path Guided Retrieval and In-Graph Text for RAG-Equipped LLM - arXiv, accessed May 23, 2025, https://arxiv.org/html/2503.00309v1
99. Constructing Knowledge Graphs From Unstructured Text Using LLMs - Neo4j, accessed May 23, 2025, https://neo4j.com/blog/developer/construct-knowledge-graphs-unstructured-text/
100. Construction of Journal Knowledge Graph Based on Deep Learning and LLM - MDPI, accessed May 23, 2025, https://www.mdpi.com/2079-9292/14/9/1728
101. Knowledge Graphs & LLMs: Multi-Hop Question Answering - Neo4j, accessed

May 23, 2025, https://neo4j.com/blog/developer/knowledge-graphs-llms-multi-hop-question-answering/
102. Leverage Knowledge Graph and Large Language Model for Law Article Recommendation: A Case Study of Chinese Criminal Law - arXiv, accessed May 23, 2025, https://arxiv.org/html/2410.04949v2
103. Enhancing Large Language Models with Knowledge Graphs - DataCamp, accessed May 23, 2025, https://www.datacamp.com/blog/knowledge-graphs-and-llms
104. [2409.04181] Combining LLMs and Knowledge Graphs to Reduce Hallucinations in Question Answering - arXiv, accessed May 23, 2025, https://arxiv.org/abs/2409.04181
105. From Chat to Publication Management: Organizing your related work using BibSonomy & LLMs - Institut für Informatik, accessed May 23, 2025, https://www.informatik.uni-wuerzburg.de/datascience/projects/nlp/?tx_extbibsonomycsl_publicationlist%5Baction%5D=download&tx_extbibsonomycsl_publicationlist%5Bcontroller%5D=Document&tx_extbibsonomycsl_publicationlist%5BfileName%5D=2401.09092.pdf&tx_extbibsonomycsl_publicationlist%5BintraHash%5D=c864f01366ecfaea30420f7c4799baf8&tx_extbibsonomycsl_publicationlist%5BuserName%5D=dmir&cHash=955940f499a78d706395897159eb3b0f
106. How to Build a Knowledge Graph for AI - DataStax, accessed May 23, 2025, https://www.datastax.com/guides/knowledge-graph-ai
107. Leveraging Medical Knowledge Graphs Into Large Language Models for Diagnosis Prediction: Design and Application Study - JMIR AI, accessed May 23, 2025, https://ai.jmir.org/2025/1/e58670
108. a knowledge graph–enhanced LLM framework for pan-cancer question answering | GigaScience | Oxford Academic, accessed May 23, 2025, https://academic.oup.com/gigascience/article/doi/10.1093/gigascience/giae082/7943459
109. Large Language Model–Driven Knowledge Graph Construction in Sepsis Care Using Multicenter Clinical Databases: Development and Usability Study - PMC, accessed May 23, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC11986385/
110. CiteFix: Enhancing RAG Accuracy Through Post-Processing Citation Correction - arXiv, accessed May 23, 2025, https://arxiv.org/html/2504.15629v1
111. [2504.15629] CiteFix: Enhancing RAG Accuracy Through Post-Processing Citation Correction - arXiv, accessed May 23, 2025, https://arxiv.org/abs/2504.15629
112. Developing an AI-Powered Tool for Automatic Citation Validation Using NVIDIA NIM, accessed May 23, 2025, https://developer.nvidia.com/blog/developing-an-ai-powered-tool-for-automatic-citation-validation-using-nvidia-nim/
113. CiteAssist: A System for Automated Preprint Citation and BibTeX Generation - ACL Anthology, accessed May 23, 2025, https://aclanthology.org/2024.sdp-1.10.pdf
114. CiteFix: Enhancing RAG Accuracy Through Post-Processing Citation

Correction, accessed May 23, 2025, https://www.researchgate.net/publication/391020117_CiteFix_Enhancing_RAG_Accuracy_Through_Post-Processing_Citation_Correction
115. An automated framework for assessing how well LLMs cite relevant medical references - PMC - PubMed Central, accessed May 23, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC12003634/
116. AI Detector - Most Accurate AI Content Checker for ChatGPT - Originality.ai, accessed May 23, 2025, https://originality.ai/ai-checker
117. Python package to autocomplete bibtex bibliographies - GitHub, accessed May 23, 2025, https://github.com/dlesbre/bibtex-autocomplete
118. RAG vs. Fine-Tuning: How to Choose - Oracle, accessed May 23, 2025, https://www.oracle.com/artificial-intelligence/generative-ai/retrieval-augmented-generation-rag/rag-fine-tuning/
119. LLM RAG vs Fine Tuning: Which Method Is Best? | Blog | Tonic.ai, accessed May 23, 2025, https://www.tonic.ai/blog/llm-rag-vs-fine-tuning
120. RAG vs. LLM fine-tuning: Which is the best approach? - Glean, accessed May 23, 2025, https://www.glean.com/blog/rag-vs-llm
121. Performant LLM Agentic Framework for Conversational AI - arXiv, accessed May 23, 2025, https://arxiv.org/html/2503.06410v1
122. Agentic AI for Scientific Discovery: A Survey of Progress, Challenges, and Future Directions, accessed May 23, 2025, https://arxiv.org/html/2503.08979v1
123. LLM-Powered AI Agent Systems and Their Applications in Industry - arXiv, accessed May 23, 2025, https://arxiv.org/html/2505.16120v1
124. LLM Agents: How They Work and Where They Go Wrong - Holistic AI, accessed May 23, 2025, https://www.holisticai.com/blog/llm-agents-use-cases-risks
125. Agentic AI Frameworks in SMMEs: A Systematic Literature Review of Ecosystemic Interconnected Agents - Preprints.org, accessed May 23, 2025, https://www.preprints.org/manuscript/202504.1797/v1
126. LangChain, accessed May 23, 2025, https://www.langchain.com/
127. LangGraph - LangChain, accessed May 23, 2025, https://www.langchain.com/langgraph
128. LLM Evaluation: Frameworks, Metrics, and Best Practices | SuperAnnotate, accessed May 23, 2025, https://www.superannotate.com/blog/llm-evaluation-guide
129. Mastering LLM Techniques: Evaluation | NVIDIA Technical Blog, accessed May 23, 2025, https://developer.nvidia.com/blog/mastering-llm-techniques-evaluation/
130. LLM Evaluation: Key Metrics, Best Practices and Frameworks - Aisera, accessed May 23, 2025, https://aisera.com/blog/llm-evaluation/
131. Enhancing NLP Robustness and Generalization through LLM-Generated Contrast Sets: A Scalable Framework for Systematic Evaluation and Adversarial Training - ResearchGate, accessed May 23, 2025, https://www.researchgate.net/publication/389714029_Enhancing_NLP_Robustness_and_Generalization_through_LLM-Generated_Contrast_Sets_A_Scalable_Framework_for_Systematic_Evaluation_and_Adversarial_Training

132. Enhancing the Robustness of LLM-Generated Code: Empirical Study and Framework - arXiv, accessed May 23, 2025, https://arxiv.org/html/2503.20197v1
133. The Ultimate Guide to Fine-Tuning LLMs from Basics to Breakthroughs: An Exhaustive Review of Technologies, Research, Best Practices, Applied Research Challenges and Opportunities (Version 1.0) - arXiv, accessed May 23, 2025, https://arxiv.org/html/2408.13296v1
134. Open-Sourced Training Datasets for Large Language Models (LLMs) - Kili Technology, accessed May 23, 2025, https://kili-technology.com/large-language-models-llms/9-open-sourced-datasets-for-training-large-language-models