BMC Bioinformatics

SOFTWARE

Open Access

# Valsci: an open-source, self-hostable literature review utility for automated large-batch scientific claim verification using large language models

Brice Edelman[1] and Jeffrey Skolnick[1*]

*Correspondence:
skolnick@gatech.edu

[1] Georgia Tech Center for the Study of Systems Biology, Atlanta, GA, USA

## Abstract

**Background:** The exponential growth of scientific publications poses a formidable challenge for researchers seeking to validate emerging hypotheses or synthesize existing evidence. In this paper, we introduce Valsci, an open-source, self-hostable utility that automates large-batch scientific claim verification using any OpenAI-compatible large language model. Valsci unites retrieval-augmented generation with structured bibliometric scoring and chain-of-thought prompting, enabling users to efficiently search, evaluate, and summarize evidence from the Semantic Scholar database and other academic sources. Unlike conventional standalone LLMs, which often suffer from hallucinations and unreliable citations, Valsci grounds its analyses in verifiable published findings. A guided prompt-flow approach is employed to generate query expansions, retrieve relevant excerpts, and synthesize coherent, evidence-based reports.

**Results:** Preliminary evaluations across claims from the SciFact benchmark dataset reveal that Valsci significantly outperforms base GPT-4o outputs in citation hallucination rate while maintaining a low misclassification rate. The system is highly scalable, processing hundreds of claims per hour through asynchronous parallelization.

**Conclusions:** By providing an open and transparent platform for large-batch literature verification, Valsci substantially lowers the barrier to comprehensive evidence-based reviews and fosters a more reproducible research ecosystem.

**Keywords:** Claim verification, Large language models, Retrieval-augmented generation, Bibliometric scoring, Chain-of-Thought, Open-source, Batch processing, Bioinformatics

## Background

The exponential increase in scientific literature in recent years has both democratized knowledge and introduced formidable bottlenecks in evidence-based research [12]. Researchers, particularly in fields such as bioinformatics and biomedical sciences, routinely confront the challenge of rapidly verifying new hypotheses or large sets of

model-generated predictions. Classical approaches to literature review require manually searching, extracting, and synthesizing evidence, and can be slow, labor-intensive, and prone to human error [2]. The urgency of generating accurate, real-time insights has thus sparked intense interest in automating, scaling, and streamlining the claim verification process.

Scientific claim verification—the specific task of assessing the accuracy and validity of statements based on evidence extracted from existing literature—is central to this process of literature review, and therefore to advancing knowledge while safeguarding research quality and reproducibility. Its importance lies in preventing the spread of misinformation, enhancing trust in published findings, and enabling rapid, evidence-informed decision-making in research and clinical settings.

#### The rise of AI-assisted literature analysis & current gaps

Recent advancements in large language models (LLMs) have transformed various domains of natural language processing (NLP), from machine translation to summarization. Yet, when deployed as stand-alone tools for scientific claim verification, LLMs often struggle with hallucinations—fabricating data or citing non-existent sources—thereby undermining their utility as reliable evidence synthesizers [6]. Many commercial platforms, such as Scite [10], Elicit, and Consensus, have integrated NLP-based capabilities to expedite literature retrieval and summarization. However, these proprietary solutions typically operate as "black boxes" and impose financial or usage constraints, making them suboptimal for large-scale or highly specialized academic applications.

Existing open-source tools have made important strides toward transparent, reproducible literature analysis. For instance, some employ retrieval-augmented generation (RAG) to ensure that LLM outputs are grounded in texts scraped from academic repositories [8]. Others provide user-friendly environments to rank or cluster abstracts based on relevance to predefined queries. Yet, most of these frameworks either lack end-to-end pipelines—requiring users to manually feed abstracts—or cannot efficiently handle large numbers of claims concurrently. As a result, researchers aiming for high-throughput verification continue to face a patchwork of partial solutions.

#### Valsci: an end-to-end solution

In response to these limitations, we present Valsci, an open-source and fully self-hostable utility for automated scientific claim verification at scale. The source code for Valsci is distributed under the June 29, 2007 Version 3 GNU General Public License. The system is built around four core principles:

- Retrieval-Augmented Generation (RAG): Valsci seamlessly integrates with the Semantic Scholar database [7, 9] to fetch relevant abstracts and full texts, ensuring its output remains anchored in verifiable sources rather than the LLM's parametric memory.
- Structured Bibliometric Scoring: In addition to relevance, Valsci incorporates an author's H-index [5], citation counts, and estimated journal impact into an overall evidence score, providing a more nuanced view of source credibility.

- Guided Prompt Flow and Chain-of-Thought (CoT): Valsci uses specialized prompts to not only refine search queries but also systematically organize retrieved evidence into comprehensive and transparent verification reports. This approach mitigates the risk of hallucinations by requiring the LLM to cite and assess real, validated excerpts.
- High-Throughput, Asynchronous Execution: Designed for large-batch tasks, Valsci employs asynchronous parallelization in Python to manage network requests, token usage, and inference calls concurrently. With typical resource settings, the system can process hundreds of claims per hour—far outpacing both manual human reviewers and single-threaded open-source tools.

### Contributions and significance

This work aims to advance automated literature verification by offering a robust, transparent, and scalable alternative to proprietary solutions. We show that Valsci not only lowers the misclassification rate for scientific claims compared to direct outputs from base language models, but also successfully eliminates citation hallucinations. Additionally, it offers a speed increase of over thirty times under reasonable API rate limits compared to manual review by undergraduate researchers. By coupling an open-source, modular design with flexible integration options for LLM backends, Valsci empowers researchers to streamline large-scale systematic reviews, meta-analyses, and high-throughput validation of computational predictions.

### Related work

The growing complexity and volume of scientific literature have driven the development of AI-assisted tools for claim verification, literature review, and citation analysis. While many systems now integrate large language models (LLMs) with bibliographic databases, existing approaches remain constrained in scalability, cost, transparency, and automation. Valsci differs by being fully open-source, self-hostable, and designed for high-throughput, large-batch claim verification. It combines retrieval-augmented generation (RAG) with structured bibliometric scoring and guided chain-of-thought (CoT) reasoning [15], offering an end-to-end, adaptable alternative to proprietary and academic solutions.

In the NLP community, claim verification is often formalized as a classification or natural language inference (NLI) task, in which claims are evaluated against retrieved evidence, producing judgments such as supported, refuted, or inconclusive [3, 11]. State-of-the-art solutions typically integrate retrieval systems with neural inference models; however, comprehensive and scalable end-to-end open-source implementations remain limited.

***Closed-Source Tools*** One of the earliest AI-driven citation analysis tools, Scite, classifies citations as supporting, contradicting, or mentioning a referenced claim from a particular academic work. By extracting citation context, it provides a useful measure of how a study is discussed in the literature. However, Scite does not actively retrieve or analyze new sources; rather, it depends on existing citation networks. This approach limits its utility for claims that require comprehensive evidence synthesis from the broader scientific corpus. Unlike Scite, Valsci dynamically searches for relevant literature,

extracts supporting text, and evaluates the claim through structured analysis, ensuring that verification is based on the strongest available evidence rather than pre-existing citation relationships.

Closed-source LLM-assisted literature review tools such as Elicit and Consensus aim to automate evidence retrieval and synthesis but remain black-box systems with unclear weighting of retrieved sources. Elicit allows users to search for relevant literature and extract key findings but imposes constraints on how many papers can be processed, with a "Pro" plan allowing only 1,200 papers per year. Consensus, built on the same Semantic Scholar database as Valsci, offers a "Pro Analysis" feature that provides structured evidence summaries, though the exact methodology behind its synthesis remains opaque. Both tools are subscription-based, limiting accessibility and transparency. Valsci, by contrast, is both open-source and allows unrestricted batch processing at no cost, making it more suitable for large-scale systematic reviews and high-volume research workflows.

***Open-Source Tools*** Unlike these proprietary systems, open-source solutions such as LitLLM and LLAssist provide more transparent AI-assisted literature analysis. LitLLM applies retrieval-augmented generation (RAG) to ensure that LLM-generated summaries are grounded in retrieved texts, reducing the risk of hallucination [1]. However, it requires users to manually provide abstracts for input, rather than performing active literature searches. LLAssist takes a batch of abstracts and ranks them by relevance to a predefined research question but does not automatically search for or retrieve relevant papers [4].

Other open-source solutions include systems such as MultiVerS [14], which focus on document-level fact-checking in controlled scientific contexts and rely on specialized model training (i.e. resource-intensive, domain-specific fine-tuning) for tasks like annotating SciFact claims. Consequently, these systems are less generalizable to arbitrary domains and lack the end-to-end automation needed for large-scale claim verification across diverse scientific fields.

These tools serve as useful components for systematic review workflows but do not provide a fully automated end-to-end pipeline. Valsci extends beyond retrieval and ranking by autonomously handling search, evaluation, bibliometric scoring, and final synthesis into structured claim verification reports.

***Focus on Scalability*** A critical distinction between Valsci and these alternatives is its ability to process large batches of claims efficiently. Existing retrieval-based systems, whether open-source or commercial, generally handle claims sequentially or within limited pre-configured batch sizes. Valsci, leveraging asynchronous parallelization of network requests, can simultaneously process as many claims as the throughput of the configured LLM providing endpoint supports. This scalability is particularly valuable for research fields requiring high-throughput verification, such as bioinformatics and clinical guideline assessments, where researchers often need to verify thousands of claims derived from computational models.

While existing AI-driven literature tools have made notable strides, none provide an open, self-hostable, end-to-end claim verification system that integrates retrieval, evaluation, bibliometric scoring, and structured synthesis into a single pipeline. Valsci fills this gap by offering an adaptable and scalable solution that is not only more cost-effective but also ensures transparency and reproducibility in AI-assisted scientific verification.

## Implementation

The claim verification process begins with query generation, where an LLM is employed to transform a claim into multiple well-structured search queries optimized for literature retrieval. Instead of relying on simple keyword extraction, Valsci employs a strategic query expansion prompt designed to maximize the likelihood that relevant literature entries are captured by the search. This involves incorporating synonyms, alternative phrasing, and mechanistic decompositions to ensure a diverse set of relevant papers is retrieved. Additionally, the system deliberately constructs queries that target both supporting and refuting evidence, ensuring that the retrieved literature provides a balanced assessment of the claim. Queries are tailored to match the search conventions of Semantic Scholar and other academic search engines, improving retrieval efficiency across multiple disciplines. The prompt used to perform this expansion is available in Appendix A. The information flow through Valsci's modular data processing structure is depicted in Fig. 1 below.

Once queries are generated, Valsci executes them against the free Semantic Scholar Search API to obtain a corpus of potentially relevant papers. The response from the API includes the unique identifier of academic works in the Semantic Scholar Database, which is locally hosted within Valsci for fast retrieval without additional API calls. Valsci includes a utility allowing the user to download the full Semantic Scholar database as part of the system setup. The pattern of leveraging the Semantic Scholar API for relevance search in combination with Valsci's indexing utility for fast retrieval of texts from the local database is what makes effective retrieval augmented generation (RAG) against such a large corpus of data possible.

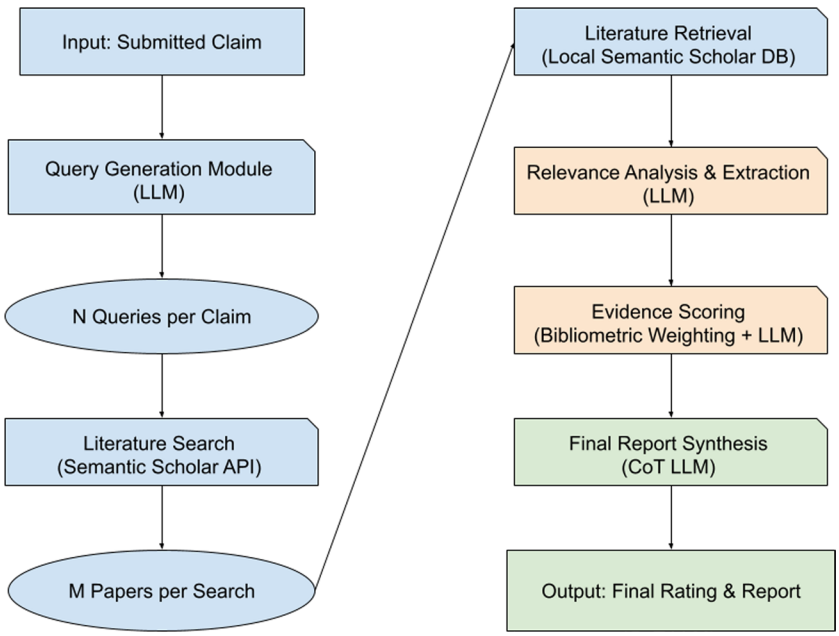


**Fig. 1** How information flows through Valsci's modular data processing structure. Information goes through the stages of retrieval (blue), analysis (orange), and synthesis (green). For each claim, up to N-times-M papers may be evaluated

To maximize access to full-text sources, the system first attempts to retrieve the work from the local Semantic Scholar Open Research Corpus (S2ORC), which contains full-text versions of millions of academic papers. By prioritizing full-text retrieval, Valsci increases the likelihood of retrieving detailed excerpts rather than relying solely on abstracts. However, to simultaneously ensure that the breadth of literature coverage is not unnecessarily limited, abstracts are still used as a fallback if the full text is not available. The retrieval engine employs deduplication techniques through an LLM call to prevent redundant papers from being analyzed multiple times. With default settings of five generated queries and a cap of up to five papers retrieved per query, the standard retrieval process typically yields between 10 and 20 papers per claim, although this configuration is adjustable based on computational resources and research requirements.

Following retrieval, each paper undergoes content extraction and relevance analysis to identify key excerpts that pertain to the claim under investigation. Using a structured LLM prompt, Valsci isolates directly relevant statements, mechanistic explanations, and empirical findings that either support or contradict the claim. These excerpts are extracted verbatim, preserving critical context, including statistical findings and methodological details. The verbatim excerpts are also paired with an explanation to provide relevant context or rationale for the excerpt's inclusion in the final analysis. The relevance of each excerpt is then assessed using a confidence score ranging from 0 to 1, ensuring that only evidence that could reasonably be considered potentially relevant is considered in later synthesis stages. If a paper is determined to have no relevance, it is classified separately and excluded from final synthesis.

To further refine the claim verification process, Valsci implements an optional evidence-scoring mechanism that ranks papers based on their scientific credibility and impact. Unlike simple relevance-based ranking, this scoring process incorporates bibliometric indicators such as the H-index of the first and last authors, the citation count of the paper, and an LLM-derived estimate of the journal's impact. The author impact metric considers the expertise of the researchers involved, while citation count provides a measure of how influential the study has been in the field. The journal impact metric is assessed using an LLM prompt aiming to recognize venue prestige based on historical publication standards and citation patterns in the model's parametric knowledge, which simplifies implementation by removing the need for integration with an external database. These three factors are weighted to produce a final evidence score, which determines the relative importance of each paper in the final assessment. The weight given to each bibliometric factor can be adjusted through the interface when submitting a claim for verification. Users who prefer not to use any bibliometric weighting can disable this feature altogether.

Once literature retrieval, relevance assessment, and evidence scoring are complete, Valsci synthesizes the results into a structured report using a guided CoT LLM prompt. This final synthesis process is designed to mimic expert-level reasoning by systematically structuring the evaluation across multiple dimensions. The report begins with a summary of supporting evidence, where the strongest corroborating papers and their key excerpts are presented. This is followed by a discussion of contradictory evidence, ensuring that conflicting findings are critically examined. The report then includes a mechanistic evaluation, exploring biological, chemical, or theoretical pathways that may

explain or refute the claim. Finally, the balance of evidence is assessed, and a claim veracity rating is assigned based on the strength and consistency of the findings.

The final claim rating follows an ordinal scale: Contradicted, Likely False, Mixed Evidence, Likely True, Highly Supported, or, if no relevant works were found, No Evidence. This rating system provides an interpretable and standardized measure of claim validity, allowing researchers to quickly determine the degree of empirical support for a given statement. For large-scale research applications, Valsci allows batch downloads of processed claim reports, enabling streamlined verification workflows across multiple projects.

The entire system is implemented in Python. The system's processing module runs as a polling loop to monitor for submitted jobs, coordinate LLM calls while staying within rate limits, and track the status of claim verification tasks during intermediate stages of analysis. API calls are performed asynchronously and only gathered when necessary using Python's asyncio library. By default, the system specifies a 0.0 temperature setting for all LLM calls without any other hyperparameter adjustments. The frontend is a graphical web interface, which can be used for submitting batches, reviewing results, and exporting files. Screenshots of the interface can be found in Appendix B. It runs comfortably on consumer hardware for users who integrate with cloud-based frontier LLM providers or other hosted LLM services on their network. Network operations, LLM queries, and file I/O are executed concurrently, enabling large-scale claim verification with minimal latency. The request management system dynamically regulates LLM usage, allowing users to tune their implementation to the specific rate limits of the LLM provider they are leveraging. Valsci is designed to be highly scalable and adaptable, supporting integration with cloud-hosted LLM APIs as well as locally deployed open-source models such as LLaMA, Deepseek-R1, and Mistral for privacy-focused research environments.

## Results

To assess Valsci's performance in large-batch scientific claim verification, we conducted experiments across three key evaluation dimensions: hallucination reduction, processing speed and efficiency, and misclassifications. We used two popular base large language models, GPT-4o (2024-11-20 version) and GPT-4o-mini (2024-07-18 version), and compared the results using these models standalone versus as the backend of the Valsci claim verification system. Processing speed was additionally evaluated against a human undergraduate researcher performing manual claim verification.

Our evaluation methodology focuses on quantifying improvements in evidence-based claim validation and computational efficiency. Experiments with both models were conducted using claims extracted from the SciFact dataset, a benchmark of scientific claims annotated with evidence from literature, developed by the Allen Institute for AI [13]. The SciFact dataset comprises claims labeled with supporting or contradicting evidence, which are critical for verifying Valsci's outputs against ground truth annotations. Specifically, we map SciFact claims labeled as "Supported" as true and "Contradicted" as false for our experimentation. To ensure accurate performance assessment, we excluded unlabeled claims (that is, claims with neither the "Supported" nor "Contradicted" label present) from the dataset, focusing solely on those with verifiable evidence. The selection

of approximately 500 claims was a deliberate choice to balance the need for a sufficient sample size to draw robust conclusions with the computational resource intensiveness of running repeated benchmarks. Importantly, our approach was developed and tested exclusively on this labeled subset, and no tweaking or adjustments were made based on the remaining, unlabeled portions of the SciFact dataset.

Our tests were run using a Microsoft Azure cloud-hosted Standard D2ads v6 (2 vcpus, 8 GiB memory) virtual machine running Debian 12.

#### Hallucination rate reduction

A major limitation of generative LLMs is their propensity for hallucination—generating plausible yet false citations, particularly in the absence of strong retrieval mechanisms. Valsci effectively eliminates hallucinations by retrieving real excerpts from Semantic Scholar into the context window, then starting the guided CoT by evaluating the available evidence from them before conducting the rest of its analysis. In a sample of 10 claims containing 211 citations, every work cited by Valsci was confirmed to be extant.

The base models, on the other hand, were asked to provide citations with the same format (paper title, lead authors, and a URL) as Valsci as part of their output. We scored the generated citations as "full hallucinations" if they appeared to be completely fabricated and as "partial hallucinations" if there were errors, but searching the internet for the title and author still yielded a similar, extant paper. Partial hallucinations were assigned half-credit in determining the hallucination rate. The comparison of citation verification between the baseline GPT-4o and Valsci is shown in Table 1 below.

In a sample of 30 papers, the GPT-4o model failed to provide a single paper without a significant error. Only three papers were scored as "partial" hallucinations. Our sample from the GPT-4o-mini model also did not include any accurate citations – the smaller model more commonly defaulted to using obvious placeholder names and URLs (such as "Jane Doe" and www.example.com) rather than the plausible-seeming but incorrect citations created by the larger model.

#### Processing speed and efficiency

Table 2, shown below, presents the average processing rate for evaluation of scientific claims and generation of structured claim verification reports by Valsci compared to an undergraduate researcher with more than an academic semester of experience working in a research setting and conducting similar work.

Valsci processes approximately 144 claims per hour under rate limits set to use no more than 10 requests and 25,000 tokens in a rolling five-second window. Even with these limits, the processing is approximately 36×faster than an undergraduate

**Table 1** Evaluation of citations provided by each literature review method in a review of ten claims

| Method | Verified citations | Partial hallucinations | Full hallucinations |
|---|---|---|---|
| GPT-4o-mini | 0 | 0 | 30 |
| GPT-4o (baseline) | 0 | 3 | 27 |
| Valsci (using GPT-4o) | **211** | **0** | **0** |

Bold indicates the best scores for the citation identification task

**Table 2** Rate of scientific claim evaluation by Valsci integrated with GPT-4o versus an undergraduate researcher

| Method | Claims processed per hour |
|---|---|
| Valsci (using GPT-4o) | 144 |
| Undergraduate Researcher | 4 |

researcher manually verifying claims. The undergraduate researcher was allowed to use publicly available academic databases and search engines but was not permitted to use generative AI for this evaluation.

We emphasize that Valsci's rate limits can be configured to maximally utilize the resources available in the environment of the user's deployment—the limiting step for processing speed is the throughput of the LLM endpoint (local- or cloud-deployed), available to the end researcher.

#### Precision, recall, F1, and uncertainty rate under various conditions

To evaluate whether accuracy is preserved during the claim verification task, we compute precision, recall, F1-score, and uncertainty rate. For clarity, we define the counts used in these calculations based on the system's classifications as follows:

- **C**: The number of claims correctly classified as true or false.
- **I**: The number of claims incorrectly classified as true or false.
- **U**: The number of claims where the system abstained from making a definitive classification (i.e., uncertain).

Using these counts, we calculate the following metrics:

- Precision: The ratio of correctly classified claims to the total number of claims where the system made a definitive classification:
- $\frac{C}{C+I}$
- Recall: The ratio of correctly classified claims to the total number of claims, including those where the system was uncertain:
- $\frac{C}{C+I+U}$
- F1 Score: The harmonic mean of precision and recall, providing a balanced measure of both:
- $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$
- Uncertainty Rate: The proportion of claims where the system abstained from making a definitive classification:
- $\frac{U}{C+I+U}$

These metrics provide a comprehensive evaluation of the system's performance. Additionally, we include columns specifying the maximum number of citations that Valsci was configured to review and whether bibliometric indicators were incorporated into the scoring.

The maximum number of citations represents the product of the number of queries generated by Valsci and the number of papers reviewed per query. Specifically, we configured Valsci for three scenarios: 3 queries with 3 papers each, 5 queries with 5 papers each, and 7 queries with 7 papers each.

Due to resource constraints, experiments varying configuration parameters and performing bibliometric ablation were conducted exclusively using the GPT-4o-mini model, chosen for its significantly lower computational cost compared to the larger GPT-4o model.

Table 3 below presents a detailed comparison of precision, recall, F1-score, and uncertainty rate across these varying conditions.

The raw results from each configuration used for calculating the above scores are available in the supplementary data hosted on Zenodo and in the "Valsci_SupplementaryData_2025_03_27.zip" additional file.

Overall, these findings indicate that Valsci consistently outperforms stand-alone LLM baselines. As shown in Table 3, using Valsci with GPT-4o yields both an increased F1 score (0.761 vs. 0.706) and a markedly lower uncertainty rate (0.267 vs. 0.360) compared to GPT-4o alone. A similar trend holds for GPT-4o-mini: integrating the model into Valsci results in a substantial jump in F1 (from 0.597 to 0.720) and a decrease in uncertainty (from 0.412 to 0.345).

Moreover, when varying Valsci's retrieval settings (i.e., the product of the number of queries and papers per query), higher citation coverage improves both recall and F1 while reducing the rate of uncertain (abstained) classifications. For instance, increasing the maximum citation count from 9 to 49 yields respective gains in F1 from 0.655 to 0.724 and a reduction in the uncertainty rate from 0.404 to 0.331. Disabling bibliometric indicators has a small but noticeable impact on performance: the system's F1 dips from 0.720 to 0.704, suggesting that weighting evidence by author impact and journal prestige can further refine final claim assessments.

In order to further validate the system on a different underlying dataset, an additional manual quality review was performed on Valsci's assessment of twenty hand-curated true claims extracted from publicly available Cochrane Reviews. Our inspection of Valsci's outputs on these claims shows reasonable analysis and chains of thought – we

**Table 3** Comparison of precision, recall, F1, and uncertainty rate across various models and system configurations

| System | Max citations | Bibliometric indicators | Precision | Recall | F1 | Uncertainty Rate |
|---|---|---|---|---|---|---|
| GPT-4o | No limit | Yes | 0.907 | 0.578 | 0.706 | 0.360 |
| Valsci (using GPT-4o) | 25 | Yes | 0.900 | **0.659** | **0.761** | **0.267** |
| GPT-4o-mini* | No limit | Yes | 0.807 | 0.473 | 0.597 | 0.412 |
| Valsci (using GPT-4o-mini) | 9 | Yes | 0.877 | 0.523 | 0.655 | 0.404 |
| Valsci (using GPT-4o-mini) | 25 | Yes | **0.909** | 0.596 | 0.720 | 0.345 |
| Valsci (using GPT-4o-mini) | 49 | Yes | 0.902 | 0.604 | 0.724 | 0.331 |
| Valsci (using GPT-4o-mini) | 25 | No | 0.886 | 0.584 | 0.704 | 0.341 |

Higher is better for Precision, Recall, and F1. Lower is better for Uncertainty Rate

*Failure to annotate after maximum of 3 retries is counted as "Uncertain" in stats calculations. Annotation failures by the standalone GPT-4o-mini occurred when the LLM became stuck generating repeated nonsense tokens up to the response length limit, which is a known issue with many small LLMs

Bold indicates the best scores for the citation identification task

include the full, unadjusted output from Valsci for these claims in the supplementary data hosted on Zenodo and in the "Valsci_SupplementaryData_2025_03_27.zip" additional file. Table 4, which includes evaluation statistics for this batch of claims, is shown below:

Of the twenty claims, Valsci was sufficiently confident to classify sixteen, and it was correct in those classifications.

## Discussion

The results presented herein demonstrate that Valsci successfully meets its primary objectives of providing an open-source, high-throughput, and verifiable framework for scientific claim verification. Critically, its integration of retrieval-augmented generation (RAG) and bibliometric scoring methods offers distinct advantages over both standalone large language models (LLMs) and existing proprietary solutions. This discussion examines how these findings extend the current state of literature verification tools, identifies limitations, and outlines opportunities for future work.

### Retrieval-augmented framework and accuracy

A key driver of Valsci's superior performance compared to the baseline GPT-4o model is its retrieval augmentation system. Rather than relying solely on parametric knowledge embedded within an LLM, Valsci always grounds its outputs in actual scientific literature. By employing guided query expansion, Valsci systematically seeks out relevant, diverse evidence. This is particularly beneficial for specialized research fields with idiosyncratic terminology, thereby broadening the scope of retrieved articles and augmenting recall. The consistent accuracy at very high throughput is striking and underscores how modern AI can not only match but occasionally exceed human capabilities in rapid literature review tasks.

Moreover, the guided chain-of-thought (CoT) synthesis approach allows Valsci to structurally integrate bibliometric indicators, context from full-text excerpts, and mechanistic insights into a coherent verification report. This layered reasoning contrasts with common single-step LLM responses, providing better transparency and interpretability. The multi-stage pipeline—encompassing query generation, excerpt extraction, bibliometric weighting, and final structured reporting—mirrors human expert workflows, but at far greater scale and throughput.

### Hallucination mitigation and bibliometric scoring

By validating each reference against a genuine retrieval from the Semantic Scholar database and employing deduplication, Valsci ensures that spurious citations do not permeate final results. The inclusion of an evidence-scoring mechanism, which integrates author-level impact (H-index), citation count, and LLM-estimated journal prestige, also

**Table 4** Precision, recall, F1, and uncertainty rate for Valsci on a set of hand-curated claims sourced from publicly available Cochrane Reviews

| System | Max Citations | Bibliometric Indicators | Precision | Recall | F1 | Uncertainty Rate |
|---|---|---|---|---|---|---|
| Valsci (using GPT-4o) | 25 | Yes | 1.000 | 0.800 | 0.889 | 0.200 |

introduces a beneficial quality-control dimension. This approach not only ranks references by credibility but inherently promotes verifiability. Papers from higher-impact journals or authors with substantial contributions in the domain intuitively hold greater weight in the final assessment, potentially reducing the influence of marginal or erroneous studies.

However, bibliometric indicators also come with caveats that merit further attention. Citation counts and H-index values can inflate due to general popularity of certain topics or self-citations, and journal-based impact metrics may overlook important findings published in less prominent venues. In fast-moving fields or interdisciplinary research areas, relying too heavily on bibliometrics can inadvertently sideline novel, high-value studies. Thus, while the inclusion of bibliometric scores bolsters validity, Valsci's ultimate goal of thoroughly capturing supporting and contradicting evidence demands that these measures be employed judiciously. Fine-tuning the weights assigned to each metric or incorporating alternative credibility heuristics could further optimize this aspect of the pipeline.

#### Scalability and throughput

Valsci's asynchronous parallelization yields a considerable throughput advantage for batch verification tasks. In experiments with rate limits set to use no more than 10 requests and 25,000 tokens in a rolling five-second window, Valsci processed claims at a rate approximately 36 times faster than undergraduate researchers, a significant boost for time-sensitive fields such as bioinformatics and clinical guideline updates. Furthermore, as discussed earlier, these rate limits can be adjusted to take maximal advantage of the LLM inference resources available in the end user's environment. This efficiency stems from orchestrating multiple processes (e.g., search requests, LLM inference, file I/O) concurrently. The capacity for researchers to integrate local or cloud-hosted OpenAI-compatible LLMs further enhances Valsci's adaptability, offering a cost-effective, flexible alternative to subscription-based platforms.

#### Limitations

Despite its strengths, Valsci has several limitations. First, although it benefits from the expansive Semantic Scholar database and the S2ORC corpus, coverage gaps remain for proprietary or paywalled journals, leading to possible incomplete evidence retrieval. Local storage of the Semantic Scholar database also requires significant hard disk space—almost two terabytes—and it must be updated periodically using the bundled utility to ensure the latest indexed works are available to the system. Moreover, while Valsci explicitly searches for both supporting and refuting sources, highly specialized claims with limited prior studies may yield sparse or no corroboration, complicating the assignment of a final veracity rating. Researchers should interpret "No Evidence" or "Mixed Evidence" findings cautiously, recognizing the dependence on existing literature availability and retrieval completeness.

Second, Valsci relies on frontier LLMs for query generation, relevance scoring, and final CoT synthesis. While substituting local or alternative LLM models can mitigate cost and privacy issues, performance may vary significantly among different LLMs,

particularly for specialized domains or non-English literature. Ensuring that the underlying LLM is robust and domain-tuned may be important to mitigate these concerns.

Lastly, interpretability and accountability remain important concerns. Although Valsci provides structured CoT outputs, discerning the rationale behind final veracity ratings may still be non-trivial for complex claims. Valsci's interpretive chain-of-thought itself is only as transparent as the language model's competence and accuracy in its articulation of intermediate reasoning. Advances in the interpretability and alignment scoring of base LLMs will allow Valsci users to better understand their outputs as well.

### Future directions

Looking ahead, several avenues could refine and extend Valsci's capabilities:

- Enhanced Bibliometric Models: Incorporating additional metrics such as altmetrics, domain-specific citation databases, or network-based analyses could yield more nuanced evidence weighting. Analysis of the LLM-derived journal impact score and comparison to methods using bibliographic databases and additional author disambiguation techniques would be helpful for optimizing the system.
- Domain Specialization of Backend Models: Domain-tuned LLMs, especially in biomedical fields, may improve query generation and excerpt extraction by leveraging specialized vocabularies and knowledge representations.
- More Sources of Evidence: Integrating data from clinical trials, chemical structures, or biological pathway databases could expand Valsci's claim verification beyond textual information from Semantic Scholar's databases.
- User-Friendly Interface Improvements: While Valsci has a graphical web interface built-in, improvements to the design and usability could encourage broader adoption by non-technical researchers.
- Explainable AI Integration: Although the system's reasoning for assigning a rating to a claim can be traced through the chain-of-thought included in the final report, advancements in explainable AI and mechanistic interpretability could be leveraged to further improve the user's confidence in the outputs.
- Additional Evaluations: The system was tested using popular closed-source models offered through Azure due to the computational resources available to our team, but further testing with different models and datasets could be insightful. It could also be interesting to evaluate the processing speed of a human with access to standard AI tools relative to Valsci's fully automated process or the human reviewer without AI access.

## Conclusion

This work introduces Valsci, an open-source, self-hostable utility that effectively bridges the gap between large language model capabilities and rigorous literature-based verification. By coupling retrieval-augmented generation with structured bibliometric scoring and guided chain-of-thought synthesis, Valsci excels in accurately assessing the veracity of scientific claims at scale. Experimental comparisons indicate that it not only dramatically reduces hallucinations relative to raw LLM outputs but also maintains high

accuracy while dramatically increasing verification speed relative to a human conducting the same task.

The system's capacity to operate asynchronously on large batches of claims empowers researchers to make full use of their resources to efficiently validate extensive sets of computational predictions, systematic review topics, or clinical guidelines. Its open design and modular architecture support custom LLM backends and evolving bibliometric metrics, providing both flexibility and transparency absent from proprietary alternatives. Although issues such as database coverage gaps, variability in LLM performance, and interpretability challenges persist, Valsci offers a valuable foundation for further innovation in automated scientific validation.

## Appendix A: Prompts

**Search query generation (system prompt):**

You are an expert at converting scientific claims into strategic literature search queries. Specifically, your queries will be used to search the Semantic Scholar database. Your goal is to generate queries that will comprehensively evaluate both supporting and contradicting evidence for a given claim.

Guidelines for Query Generation:

1 Identify core concepts and their relationships in the claim
2 Include field-specific terminology, common synonyms, and alternative phrasings
3 Decompose complex claims into testable components
4 Use only plain text search queries, no boolean operators or special syntax
5 Break hyphenated terms into separate words (e.g. "drug-resistant"—>"drug resistant")
6 Balance specificity with recall—avoid overly narrow or broad queries
7 Consider both direct evidence and mechanistic studies
8 Account for competing hypotheses and alternative explanations

Search Strategy:

– Generate queries for direct evidence testing the claim
– Include queries for underlying mechanisms and pathways
– Consider related phenomena that could provide indirect evidence
– Look for potential confounding factors or methodological challenges
– Search for systematic reviews and meta-analyses when applicable
– The queries should be sufficiently diverse to capture as much relevant information as possible and avoid overlap

Example Approach:
For "Metformin increases lifespan", you could consider:

- Direct evidence: clinical studies, epidemiological data
- Mechanisms: AMPK pathway, insulin sensitivity, mitochondrial function
- Related outcomes: mortality, age-related diseases, biomarkers of aging
- Potential confounds: diabetes status, age, concurrent medications

when generating your queries.
Output Format:

```
{
  "explanations": [
    "string explaining the rationale and strategy behind each query"
  ],
  "queries": [
    "search query strings formatted for academic databases"
  ]
}
""")
```

**Search query generation (user prompt):**
Generate {*num_queries*} strategic search queries to evaluate this scientific claim: "{*claim_text*}".
Requirements:

- Each query should be precisely formulated for Semantic Scholar database searching
- Include a mix of specific and broader search strategies
- Consider both direct evidence and mechanistic studies
- Account for different research methodologies and study types
- Use only plain text search queries, no boolean operators or special syntax
- Break hyphenated terms into separate words (e.g. "drug-resistant"—>"drug resistant")

Return results as a JSON object with 'explanations' and 'queries' arrays.
**Paper analysis (system prompt):**
You are an expert at analyzing scientific papers and evaluating their relevance to specific claims through both direct evidence and mechanistic pathways.
Guidelines for Analysis:

1. Evaluate direct evidence that supports or refutes the claim
2. Identify mechanistic evidence that strengthens or weakens the claim's plausibility
3. Examine methodology, results, and conclusions with careful attention to detail

4. Extract verbatim quotes with complete scientific context in which they are found
5. Consider study limitations and their impact on evidence quality
6. Assess both statistical and practical significance of findings
7. Note experimental conditions that may affect generalizability

Guidelines for Quote Extraction:

1. Include complete sentences or paragraphs that capture full context.
2. Maintain exact spelling, punctuation, and formatting.

Return a JSON object with:

```
{
  "relevance": float (0-1),
  "excerpts": list of relevant verbatim sentences or paragraphs (a list of strings),
  "explanations": list of explanations (a list of strings), one for each excerpt, addressing how the
excerpt relates to the claim (direct or mechanistic) and the strength and limitations of the
evidence
  "non_relevant_explanation": string (only if relevance < 0.1),
  "excerpt_pages": list of page numbers (or null if not available)
}
```

**Paper analysis (user prompt):**

Analyze this paper content for both direct and mechanistic evidence related to the following claim:

Claim: {*claim_text*}.

Paper content:

{cleaned_content}.

Tasks:

1. Determine if this paper provides relevant evidence for or against the claim
2. Extract complete, verbatim sentences or paragraphs that:

– Support or refute the claim
– Describe relevant mechanisms
– Provide essential context for understanding the evidence

3. For each sentence or paragraph:

– Explain how it relates to the claim
– Note whether it's direct evidence or mechanistic
– Include any limitations or caveats

4. If relevance < 0.1, provide a detailed explanation why
Remember:

– Include complete sentences and surrounding context
– Maintain exact wording, including statistical details

**Calculate venue impact (system prompt):**
You are an expert in academic publishing and research venues.
Estimate the impact/prestige of an academic venue on a scale of 0–10.
Consider factors like:

– Venue reputation in the field
– Publication standards and peer review
– Typical citation rates
– Publisher reputation

Return only json object with a single key "score" and a number between 0 and 10.
**Calculate venue impact (user prompt):**
Rate the academic impact and prestige of this venue:
Venue: {*paper_journal*}.
Return only json object with a single key "score" and a number between 0 and 10, where:
0–2: Low impact or predatory venues.
3–5: Legitimate but lower impact venues.
6–8: Well-respected, mainstream venues.
9–10: Top venues in the field.
**Synthesize report (system prompt):**
You are an expert scientific reviewer specializing in evaluating the plausibility of scientific claims based on evidence from academic papers and your expert knowledge. Your task is to synthesize a detailed evaluation of the claim and assign a final plausibility rating based on both your scientific knowledge and the evidence provided in the paper excerpts you receive.
The final rating you assign should be one of the following:

– Contradicted: Strong evidence refutes the claim.
– Likely False: Evidence suggests the claim is unlikely but not definitively refuted.
– Mixed Evidence: It is not clear whether the supporting or contradicting evidence is stronger.
– Likely True: The claim is supported by reasonable evidence, though it may not be definitive.
– Highly Supported: The claim is strongly supported by compelling and consistent evidence.

– No Evidence: There is no evidence to support or refute the claim in the provided papers.

When formulating your evaluation, consider the following aspects:

– Supporting Evidence: Summarize the most robust evidence that supports the claim. Be specific, referencing the findings of relevant papers and their implications.
– Caveats or Contradictions: Identify any limitations, contradictory findings, or alternative interpretations that might challenge the claim.
– Analysis: Based on your expertise, analyze the systems and structures relevant to the claim for any deeper relationships, mechanisms, or second-order implications that might be relevant.
– Assessment: Assess the balance of evidence, explaining which side is more compelling and why. Contextualize caveats but avoid undue hedging; consider the overall weight of the evidence like an expert would.
– Rating Assignment: Choose a single category from the list above that best reflects the overall strength of evidence for the claim. Assign this rating based on the preponderance of evidence, contextualizing caveats without allowing minor exceptions to overshadow the dominant trend.

Use the following process to review the claim and evidence and conduct your analysis:

First, under the explanationEssay attribute, write your thoughts as an essay with distinct paragraphs for:

– Supporting evidence.
– Caveats or contradictory evidence.
– Analysis of potential underlying mechanisms, deeper relationships, or second-order implications.
– An explanation of which rating seems most appropriate based on the relative strength of the evidence.

Once you've written the essay, read over it and analyze your logic one more time for any flaws or inconsistencies. If you find any, revise your rating and explanation accordingly in the finalReasoning attribute. Otherwise, you can reaffirm your rating and explanation in the finalReasoning attribute. This string can be as long as you need to conduct a rigorous final analysis.

Lastly, assign the final rating from the list above in the claimRating attribute.

You will receive the text of the claim and excerpts from academic papers that could support or refute the claim. Craft your evaluation, then provide a JSON response in the following format:

```
{
  "explanationEssay": "<plain text detailed essay explanation>",
  "finalReasoning": "<plain text additional reasoning for the rating>",
  "claimRating": "<rating, one of the following: Contradicted, Likely False, Mixed Evidence,
Likely True, Highly Supported, No Evidence>"
}
```

**Synthesize report (user prompt):**

Evaluate the following claim based on the provided evidence and counter-evidence from scientific papers:

Claim: {*claim_text*}.

Evidence:

{paper_summaries_text}.

## Appendix B: Screenshots

Figure 2 shown below, is a screenshot of Valsci's landing page.

Figure 3 shown below, is a screenshot illustrating the configuration options available to the user when submitting a claim processing job.

Figure 4 shown below, is a screenshot of the progress page allowing the user to monitor the system as it runs.

Figure 5, shown below, is a screenshot showing the "Batch Results" page.

Figure 6 shown below, is a screenshot of an individual report.

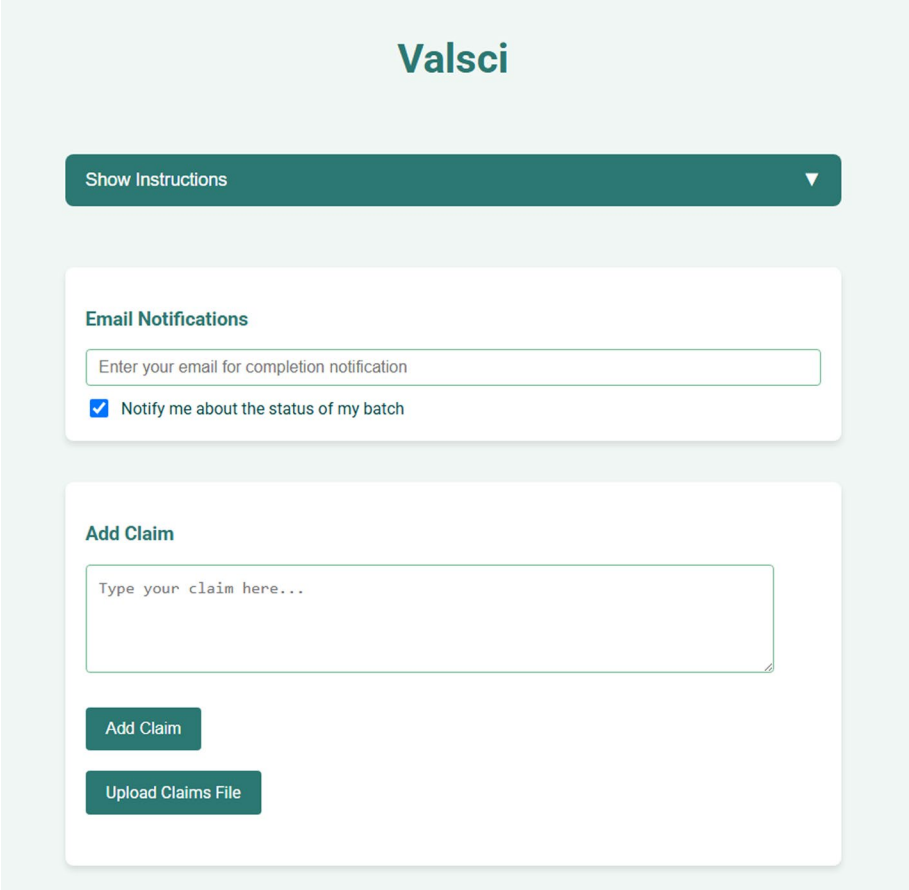


**Fig. 2** A screenshot of the Valsci landing page

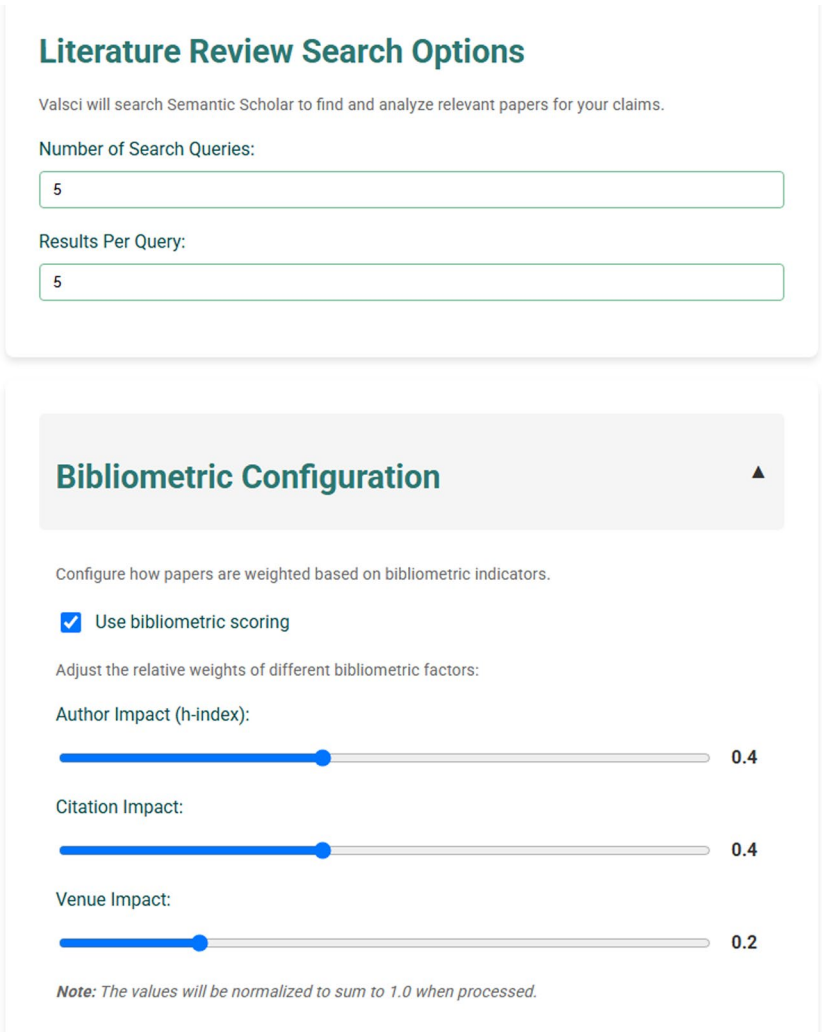


**Fig. 3** A screenshot of Valsci's job submission configuration options

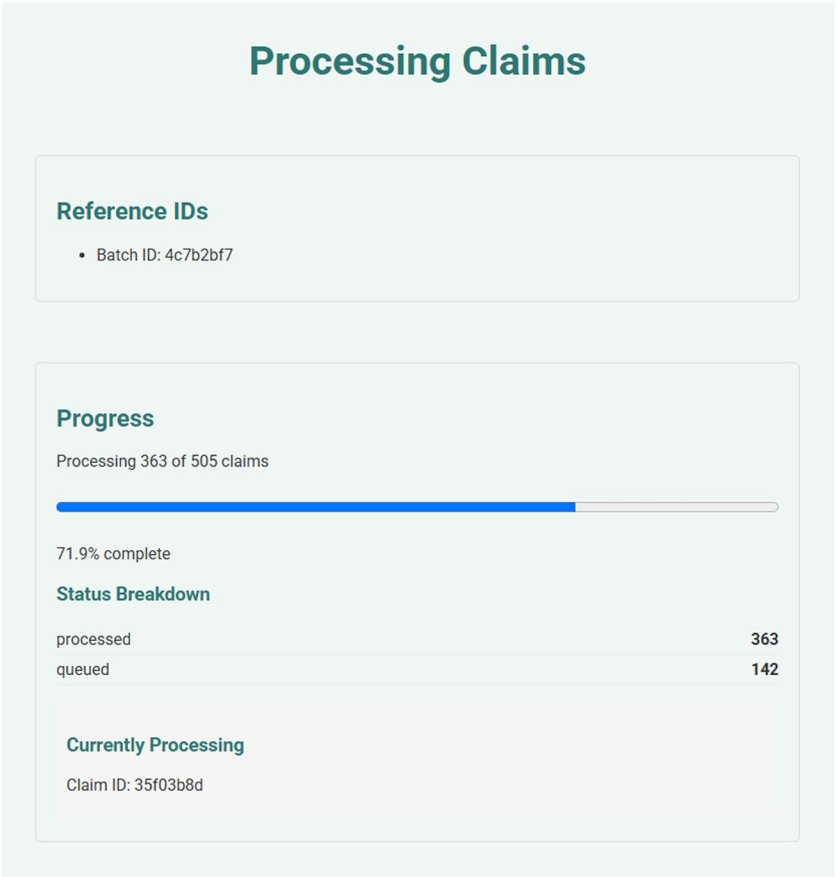


**Fig. 4** A screenshot of Valsci's progress monitoring screen

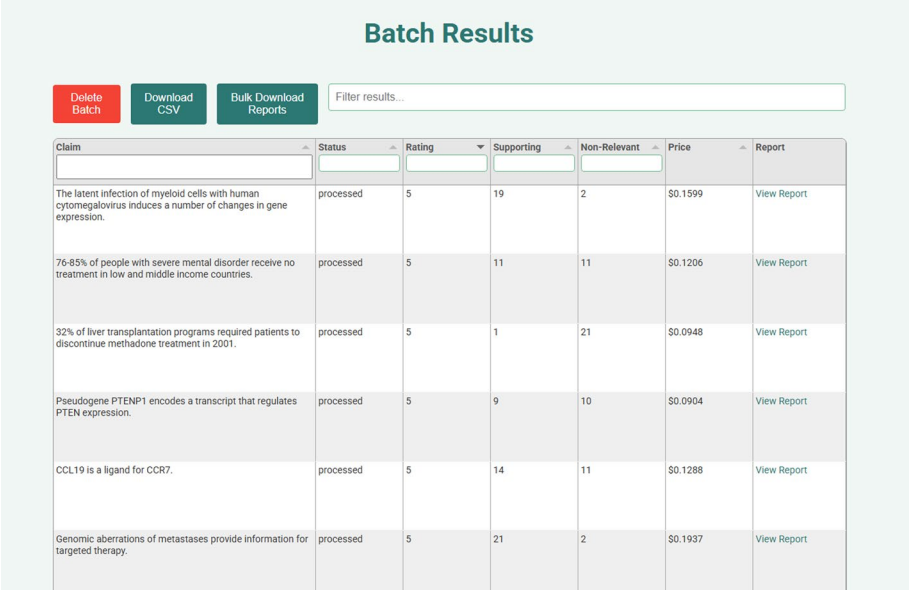


**Fig. 5** A screenshot of the Valsci Batch Results screen

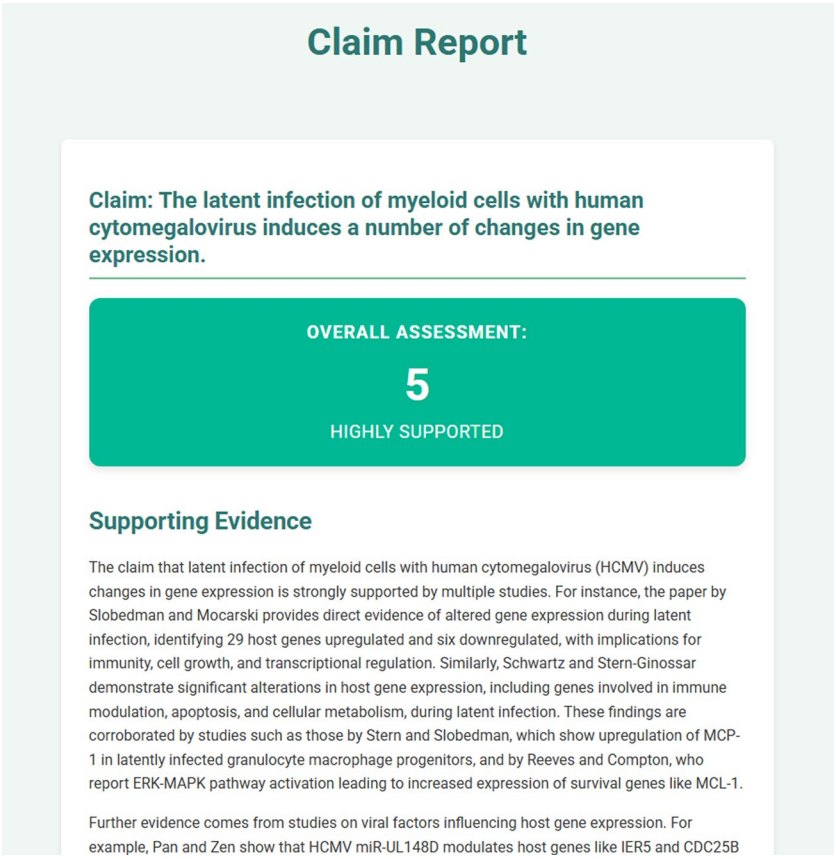


**Fig. 6** A screenshot of the top of a Valsci report

**Abbreviations**

| | |
|---|---|
| LLM | Large Language Model |
| RAG | Retrieval-Augmented Generation |
| CoT | Chain-of-Thought |
| API | Application Programming Interface |
| S2ORC | Semantic Scholar Open Research Corpus |
| GPL | General Public License |

## Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-025-06159-4.

Additional file1 (ZIP 6000 kb)

**Acknowledgements**
We thank Jessica Forness for proofreading the manuscript. We also thank our undergraduate researcher, Shaya Farahmand, for his manual literature review assessment.

**Author contributions**
BE and JS collaborated on the ideation and design of the Valsci system. BE programmed the system and provided benchmarking and data analysis. JS provided expertise to solve technical issues and guide the development effort. BE wrote the text of the manuscript and JS provided review and revisions.

**Funding**
This research was supported in part by grant GMR35-118039 of the Division of General Medical Sciences of the National Institutes of Health.

**Availability of data and materials**
The source code is available in the GitHub repository, https://github.com/bricee98/Valsci A snapshot of the source code at the time of submission and the datasets generated and analyzed during the current study are available in the Zenodo repository, https://zenodo.org/records/15098570. This data is also included as the additional file "Valsci_SupplementaryData_2025_03_27.zip".

## Declarations

**Conflict of interest**
The authors declare no competing interests.

**Ethics approval and consent to participate**
Not applicable.

**Consent for publication**
Not applicable.

Received: 26 February 2025 Accepted: 7 May 2025
Published online: 28 May 2025

**References**

1. Agarwal S, Laradji IH, Charlin L, Pal C. LitLLM: a toolkit for scientific literature review (No. arXiv:2402.01788). 2024. arXiv. https://doi.org/10.48550/arXiv.2402.01788.
2. Haddaway NR, Bethel A, Dicks LV, Koricheva J, Macura B, Petrokofsky G, Pullin AS, Savilaakso S, Stewart GB. Eight problems with literature reviews and how to fix them. Nat Ecol Evol. 2020;4(12):1582–9. https://doi.org/10.1038/s41559-020-01295-x.
3. Hanselowski A, PVS A, Schiller B, Caspelherr F, Chaudhuri D, Meyer CM, Gurevych I. *A Retrospective Analysis of the Fake News Challenge Stance Detection Task* (No. arXiv:1806.05180). 2018. arXiv. https://doi.org/10.48550/arXiv.1806.05180.
4. Haryanto CY. LLAssist: simple tools for automating literature review using large language models (No. arXiv:2407.13993). 2024. arXiv. https://doi.org/10.48550/arXiv.2407.13993.
5. Hirsch JE. An index to quantify an individual's scientific research output. Proc Natl Acad Sci USA. 2005;102(46):16569–72. https://doi.org/10.1073/pnas.0507655102.
6. Huang L, Yu W, Ma W, Zhong W, Feng Z, Wang H, Chen Q, Peng W, Feng X, Qin B, Liu T. A survey on hallucination in large language models: principles, taxonomy, challenges, and open questions. ACM Trans Inf Syst. 2025;43(2):42:1-42:55. https://doi.org/10.1145/3703155.
7. Kinney R, Anastasiades C, Authur R, Beltagy I, Bragg J, Buraczynski A, Cachola I, Candra S, Chandrasekhar Y, Cohan A, Crawford M, Downey D, Dunkelberger J, Etzioni O, Evans R, Feldman S, Gorney J, Graham D, Hu F, Weld D.S. The Semantic Scholar Open Data Platform (No. arXiv:2301.10140). 2023. arXiv. https://doi.org/10.48550/arXiv.2301.10140.
8. Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, Küttler H, Lewis M, Yih W, Rocktäschel T, Riedel S, Kiela D. Retrieval-augmented generation for knowledge-intensive NLP tasks (No. arXiv:2005.11401). 2021. arXiv. https://doi.org/10.48550/arXiv.2005.11401.

9. Lo K, Wang LL, Neumann M, Kinney R, Weld DS. S2ORC: the semantic scholar open research corpus (No. arXiv:1911.02782). 2020. arXiv. https://doi.org/10.48550/arXiv.1911.02782.
10. Nicholson JM, Mordaunt M, Lopez P, Uppala A, Rosati D, Rodrigues NP, Grabitz P, Rife SC. scite: a smart citation index that displays the context of citations and classifies their intent using deep learning. Quant Sci Stud. 2021;2(3):882–98. https://doi.org/10.1162/qss_a_00146.
11. Nie Y, Chen H, Bansal M. Combining fact extraction and verification with neural semantic matching networks (No. arXiv:1811.07039). 2018. arXiv. https://doi.org/10.48550/arXiv.1811.07039.
12. Pan RK, Petersen AM, Pammolli F, Fortunato S. The memory of science: Inflation, myopia, and the knowledge network. J Informet. 2018;12(3):656–78. https://doi.org/10.1016/j.joi.2018.06.005.
13. Wadden D, Lin S, Lo K, Wang LL, Zuylen van M, Cohan A, Hajishirzi H. Fact or Fiction: Verifying Scientific Claims (No. arXiv:2004.14974). 2020. arXiv. https://doi.org/10.48550/arXiv.2004.14974.
14. Wadden D, Lo K, Wang LL, Cohan A, Beltagy I, Hajishirzi H. MultiVerS: Improving scientific claim verification with weak supervision and full-document context (No. arXiv:2112.01640). 2022. arXiv. https://doi.org/10.48550/arXiv.2112.01640
15. Wei J, Wang X, Schuurmans D, Bosma M, Ichter B, Xia F, Chi EH, Le QV, Zhou D.: Chain-of-thought prompting elicits reasoning in large language models. In: Proceedings of the 36th International Conference on neural information processing systems, 2022; 24824–24837.

## Publisher's Note