SciSciGPT: Advancing Human-AI Collaboration in the Science of Science

Erzhuo Shao[1,2,3,4], Yifang Wang[1,2,3,5*], Yifan Qian[1,2,3,5*], Zhenyu Pan[4], Han Liu[4], Dashun Wang[1,2,3,4,5†]

[1]Center for Science of Science and Innovation, Northwestern University, Evanston, IL, USA
[2]Ryan Institute on Complexity, Northwestern University, Evanston, IL, USA
[3]Northwestern Innovation Institute, Northwestern University, Evanston, IL, USA
[4]McCormick School of Engineering, Northwestern University, Evanston, IL, USA
[5]Kellogg School of Management, Northwestern University, Evanston, IL, USA
*These authors contributed equally
[†]Correspondence to: dashun.wang@kellogg.northwestern.edu

Abstract

The increasing availability of large-scale datasets has fueled rapid progress across many scientific fields, creating unprecedented opportunities for research and discovery while posing significant analytical challenges. Recent advances in large language models (LLMs) and AI agents have opened new possibilities for human-AI collaboration, offering powerful tools to navigate this complex research landscape. In this paper, we introduce *SciSciGPT*, an open-source, prototype AI collaborator that uses the science of science as a testbed to explore the potential of LLM-powered research tools. *SciSciGPT* automates complex workflows, supports diverse analytical approaches, accelerates research prototyping and iteration, and facilitates reproducibility. Through case studies, we demonstrate its ability to streamline a wide range of empirical and analytical research tasks while highlighting its broader potential to advance research. We further propose an LLM agent capability maturity model for human-AI collaboration, envisioning a roadmap to further improve and expand upon frameworks like *SciSciGPT*. As AI capabilities continue to evolve, frameworks like *SciSciGPT* may play increasingly pivotal roles in scientific research and discovery, unlocking further opportunities. At the same time, these new advances also raise critical challenges, from ensuring transparency and ethical use to balancing human and AI contributions. Addressing these issues may shape the future of scientific inquiry and inform how we train the next generation of scientists to thrive in an increasingly AI-integrated research ecosystem.

1 - Introduction

Scientific advances are foundational to improving quality of life, driving global health outcomes, and fostering growth and prosperity[1–6]. Understanding the mechanisms underlying these advances is critical for shaping effective science policies and empowering scientists to address high-risk and high-impact questions. The field of the science of science (SciSci) has emerged to tackle this challenge[1,7,8], leveraging interdisciplinary approaches to explore how science is conducted, funded, and applied. SciSci has seen rapid growth, partly fueled by the increasing availability of large-scale datasets that capture a wide array of activities in science and innovation[9–17], from the inner workings of science to its upstream investments and downstream societal impacts. These advances mirror broader progress in computational social science[18], where increasingly sophisticated datasets and computational methods are enabling researchers to analyze complex systems of human behavior, dynamics, and interactions.

However, the very advances in data and tools that make this research possible also introduce significant technical challenges. The growing scale and complexity of datasets, coupled with the rapid evolution of computational methods, create barriers to entry for researchers and demand substantial technical expertise. At the same time that science is becoming more complex, individual expertise is becoming more narrowly focused, leading to an increase in specialization[19–21]. Together, these challenges highlight the need for new approaches to help researchers efficiently navigate, analyze, and derive insights from these rich data sources[22].

Recent advances in large language models (LLMs) and Artificial Intelligence (AI) agents have opened new possibilities for advancing human-AI collaboration[23–25], offering potential tools to navigate the complex and rapidly evolving research landscape. Recent studies show that LLMs are increasingly adept at performing high-level cognitive tasks, including in-context learning[26], complex reasoning[27,28], planning, tool usage[29,30], and coding[31–34]. Researchers have begun harnessing these capabilities, using LLMs as central controllers in autonomous task-executing LLM agents across various domains, including retrieval-augmented generation[35,36] and automated data science[37–40].

These developments suggest the potential to leverage LLM agents for SciSci research. An effective LLM agent in this context would be able to understand the SciSci literature, the data available to use for research, and the tools and methods for analysis and visualization. It would organize and execute progressive workflows for SciSci research questions, taking on the technical workload and supporting a low-code or no-code research process. If designed appropriately, such a system could substantially increase research efficiency, lower barriers to entering the field, facilitate reproducibility, and support early-stage exploration and idea generation. Moreover, its capabilities and reach could expand further as LLMs continue to evolve.

In this paper, we present our initial effort to explore LLM agents' potential in this realm, including developing *SciSciGPT* as a proof-of-concept AI collaborator, under the guidance of a comprehensive LLM Agent capability maturity model. *SciSciGPT* is an AI collaborator for the science of science. It offers a chat interface for public use at https://sciscigpt.com that functions

similarly to ChatGPT alongside a fully open-source implementation at https://github.com/erzhuoshao/SciSciGPT, ensuring full transparency and enabling other researchers to reproduce and build on the work. Our framework incorporates a range of functionalities: retrieving pertinent SciSci publications based on user inquiries, writing code to extract data from complex databases, conducting data analytics using advanced methods, creating visualizations of results and insights, and evaluating its own analytical and visual outputs. By combining these capabilities into a seamless, AI-powered research workflow, *SciSciGPT* lowers technical barriers, enhances efficiency, and enables a new mode of human-AI collaboration in SciSci. Here, we offer an overview of *SciSciGPT*'s architecture and assess its efficacy, including case studies that showcase *SciSciGPT*'s ability to support and enhance research efforts.

It is important to emphasize that our intent is to develop *SciSciGPT* as a prototype. While its early results appear promising, *SciSciGPT*'s performance and value are expected to grow with the advancement of LLMs–particularly their complex reasoning abilities–and with ongoing refinements to the *SciSciGPT* framework. Furthermore, while this paper focuses on the science of science as a testbed, *SciSciGPT* offers a generalizable framework for advancing human-AI collaboration across diverse fields. The open-source nature of *SciSciGPT* allows researchers to flexibly adapt and extend the tool to meet their specific needs. With appropriate adjustments and the integration of domain-specific knowledge, *SciSciGPT* could be applied to other scientific domains, particularly in data-intensive domains and disciplines traditionally less reliant on computational methods, which may enable more interdisciplinary research and collaborations.

To this end, we further propose an LLM Agent capability maturity model to envision a roadmap for developing AI research collaborators, which encompasses four key maturity levels: functional capabilities, workflow orchestration, memory architecture, and human-AI collaborative paradigms. As a proof-of-concept of the capability maturity model, *SciSciGPT* embodies several key features from the model, and the proposed maturity model provides a framework to guide further developments and extensions, offering a strong foundation for agentic AI system development across broad research environments.

Overall, *SciSciGPT* represents an initial—yet important—step toward broader and more efficient exploration of data-driven insights in research. Our work makes three key contributions. (1) By integrating AI capabilities and agents, we develop *SciSciGPT* as an AI collaborator for science of science, which makes AI-driven research assistance accessible and practical. (2) We validate *SciSciGPT*'s effectiveness through case studies, comparisons, and expert interviews. (3) We propose an LLM Agent capability maturity model, providing a general framework for AI system development for human-AI collaborations. Ultimately, by fostering a deeper partnership between humans and machines, *SciSciGPT* opens new possibilities for innovation and discovery in the field of the science and science, and beyond.

2 - Related Work

SciSciGPT builds on recent advances in LLMs which have shown remarkable capacities for code generation[31–34], tool use, and planning. Frameworks such as Toolformer[29] and ReAct[30], for example, have pioneered new ways to harness LLMs for tool usage, and various cutting-edge planning methodologies[27,28,41] have showcased LLMs' ability to break tasks down into specific procedures.

SciSciGPT also benefits from advances in Retrieval-Augmented Generation (RAG), which enables LLMs to retrieve relevant external information in real time[42,43]. This enhances response accuracy by mitigating LLMs' tendency to hallucinate or generate incorrect information in specialized domains[44–48] and by helping overcome limitations imposed by knowledge cutoffs that create gaps in their understanding[49]. First introduced by Lewis et al.[50], RAG has evolved from early frameworks like ReAct[30] and MRKL[51] to more sophisticated approaches, including Self-Ask[52], SELF-RAG[53], and PaperQA[35,36], enabling systems to handle complex queries with multi-step reasoning and fact verification. Further innovations, such as HyDE[54] and Chain-of-Note[55], enhance retrieval accuracy and information integration.

Researchers have leveraged these developments to create autonomous LLM data agents—integrated systems that combine code generation, tool use, planning, and RAG to orchestrate tasks in a wide range of fields. In the data science domain, two primary types of LLM-based agents are particularly relevant:

> (1) Code-writing agents are designed specifically for code writing tasks[56], autonomizing the generation of code for data science projects. These frameworks include TaskWeaver[57], Data-Copilot[58], and DA-Agent[37], which enhance data analysis capabilities through Python sandbox integration or enable database interaction and external knowledge extraction. DS-Agent[38] integrates LLM agents with case-based reasoning, leveraging Kaggle's expert knowledge for automated machine learning. LAMBDA[40] develops a multi-agent system with specialized programmer and inspector roles, while Data Interpreter[39] uses hierarchical graph modeling and programmable node generation to support a wide range of machine-learning tasks.
>
> (2) Co-scientist pipelines in the data science field are multi-agent frameworks designed to emulate the research process. They follow specific procedures to generate ideas, write code, interpret results, and generate reports. For instance, Lu et al.[59] developed an AI scientist for machine learning research, an AI system designed to automate the entire research process, from idea generation and experimentation to paper writing. Similarly, Schmidgall et al.[60] introduced AgentLaboratory, a framework that simulates collaborative machine learning research by using LLM agents to automate tasks across the research pipeline, from idea generation to reporting.

While these models have demonstrated the capacity of LLMs to generate effective code and the usefulness of multi-agent systems for research tasks, these applications are often focused on machine learning tasks. They do not include custom data repositories that allow for the data

insight exploration that *SciSciGPT* facilitates, and few have a self-reflection mechanism for iterative improvement. Moreover, these co-scientist pipelines are fully automated, whereas *SciSciGPT* is designed to be transparent and interactive. It is intentionally not fully automated, serving instead as a conversational AI collaborator that allows for iterative human-AI collaborations to explore and extract data-driven findings.

SciSciGPT is further distinguished by its focus on advancing research and discovery in a specific research domain, which requires an integrated understanding of the literature and relevant datasets, measurement approaches, and empirical methods and toolkits. *SciSciGPT* uses the field of the science of science as a testbed. This multidisciplinary field offers a rapidly expanding evidence base and insights on science and innovation, leveraging rich sources of data and a range of computational tools. By infusing the agentic features of LLMs, including code generation, tool use, planning, and reasoning, with domain-specific knowledge and expertise, including SciSci literature, datasets, and empirical methods, *SciSciGPT* aims to offer a prototype of a new form of human-AI collaboration. From this perspective, *SciSciGPT* may be viewed as a meso-level LLM-based research agent—neither too general nor too specific. It is capable of answering a range of research questions with greater depth than general agents while maintaining transparency in its methodology and offering domain-specific knowledge and toolkits that are tailored to the unique analytical needs of the domain researchers.

3 - *SciSciGPT*

3.1 - System overview

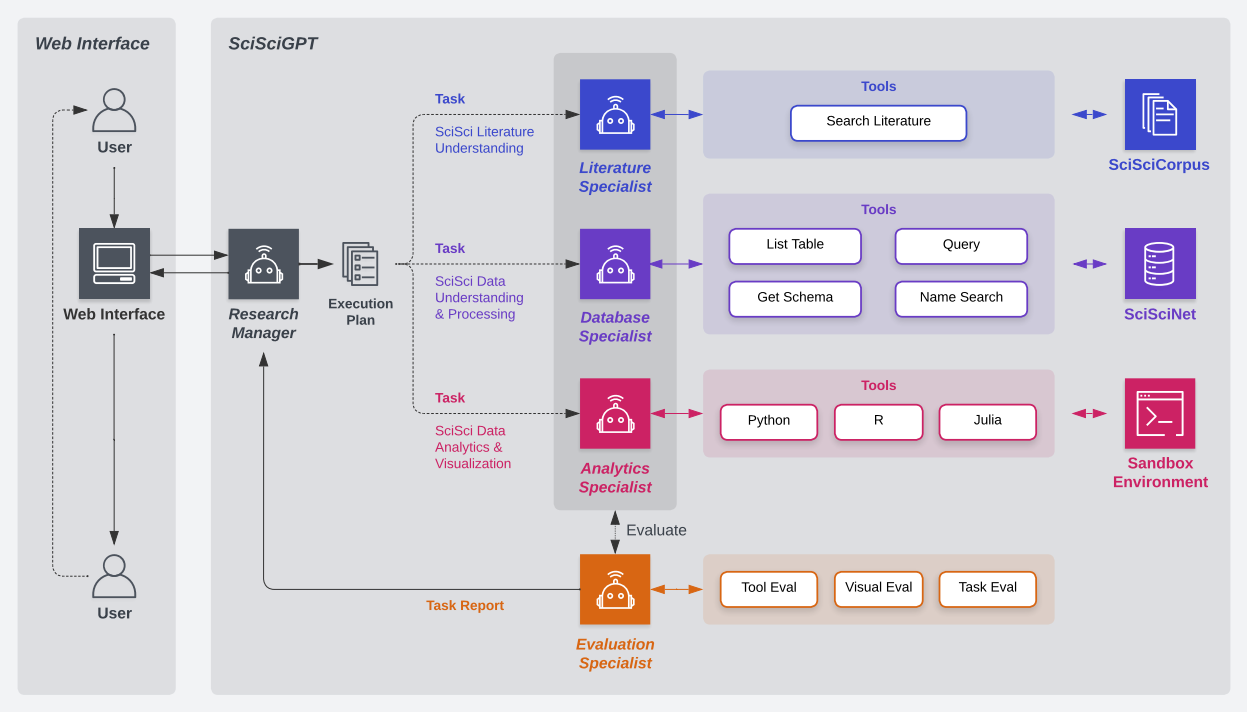


Figure 1: *SciSciGPT* System Architecture. This diagram illustrates the modular design of *SciSciGPT*, an AI collaborator for the science of science. Users submit requests through a web chat interface to the *ResearchManager* agent, which breaks user requirements down into tasks and delegates them to the appropriate specialist agents, *LiteratureSpecialist*, *DatabaseSpecialist*, *AnalyticsSpecialist*, and *EvaluationSpecialist*. These specialists provide assistance with literature understanding, data processing, data analytics, visualization, and quality assessment through their interactions with tools, data sources, and sandbox environments. Each then returns its results to the *ResearchManager* to manage the workflow.

SciSciGPT is a multi-agent AI system designed to serve as a research collaborator for science of science researchers and practitioners. Drawing inspiration from the core research tasks of domain researchers, *SciSciGPT* comprises five specialized modules, each focusing on a distinct component of the research workflow:

- The ***ResearchManager*** agent functions as a project leader and central coordinator. It orchestrates the research workflow, breaking complex research questions down into tasks and assigning them to the four specialist agents listed below.
- The ***LiteratureSpecialist*** agent focuses on comprehension and synthesis, searching for and organizing relevant information from the SciSci literature.
- The ***DatabaseSpecialist*** agent handles data processing tasks, managing complex data extraction, transformation, and basic statistics across scholarly databases. This agent is equipped to interact with a comprehensive scholarly data repository.

- The ***AnalyticsSpecialist*** agent focuses on statistical analysis and modeling, implementing empirical methods and analytical techniques and generating visualizations to support empirical investigations.
- The ***EvaluationSpecialist*** agent assesses the quality, relevance, and rigor of *SciSciGPT*'s analyses, visualizations, and methodological choices, allowing the system to identify potential improvements and adjust its approach iteratively.

When the *ResearchManager* receives a research question, it formulates an execution plan, assigning tasks to appropriate specialists. Each specialist agent formulates sub-plans, invokes tool use, and engages in iterative reasoning until the task is completed. As each plan is executed, the *EvaluationSpecialist* is invoked to assess progress across multiple levels, guiding the specialist's next step. After the specialist finishes each task, the control returns to the *ResearchManager* for subsequent task allocation and execution. This hierarchical structure supports flexible task decomposition and delegation for any user question, enabling SciSci researchers to interact seamlessly with the system through conversation, refine their research questions, and explore different approaches as needed. This conversational, multi-agent architecture enables domain-specific functionalities while maintaining the original LLM's general capability, such as instruction following, question answering, and common-sense reasoning.

3.2 - Case studies

To illustrate the functionality and value of this multi-agent system, we present two case studies that showcase how researchers can leverage this tool in real-world scenarios. These examples highlight the interaction between the user and the system, the workflow, the methodological approach, and the tangible outcomes that *SciSciGPT* produces.

Case study #1: Collaboration network among Ivy League universities

(See full chat history in SN 5.1)

Imagine the following research question: What does scientific collaboration look like among Ivy League universities? This question might be asked by a SciSci researcher who studies scientific collaboration and teamwork, an increasingly important area in the field. Research shows that great breakthroughs today rarely stem from lone geniuses; rather, they disproportionately emerge from collaborative efforts that often transcend institutional or geographic boundaries[1,7,8,61]. This question could also be asked by a practitioner, such as an institutional leader who is interested in quantitative answers to the question that could inform efforts to foster more strategic partnerships.

To answer the question using conventional approaches, the researcher would need to consider all papers published by each of the Ivy League universities, filter out papers that feature collaborations between at least two of these universities, and calculate the frequency of co-authorship for each pair of universities. As co-authorship analyses are often represented as

networks, the researcher might also consider creating a visualization of the collaboration network among the eight universities. Each node would represent a university, and the links between them would denote the collaborative strength (i.e., the number of papers that were co-authored by two universities). As part of this process, the researcher would need to identify the necessary datasets, write scripts to query the data and extract information, compute the measures of collaboration, and apply network analysis tools for visualization, which requires specialized expertise in network science[62]. In total, this task could take a researcher hours to complete, depending on their experience and skill set.

To see *SciSciGPT* tackle this task, we gave it the following prompt:

Human Input
`Generate a network for collaborations among Ivy League Universities between 2000 and 2020. Optimize its colors and annotations.`

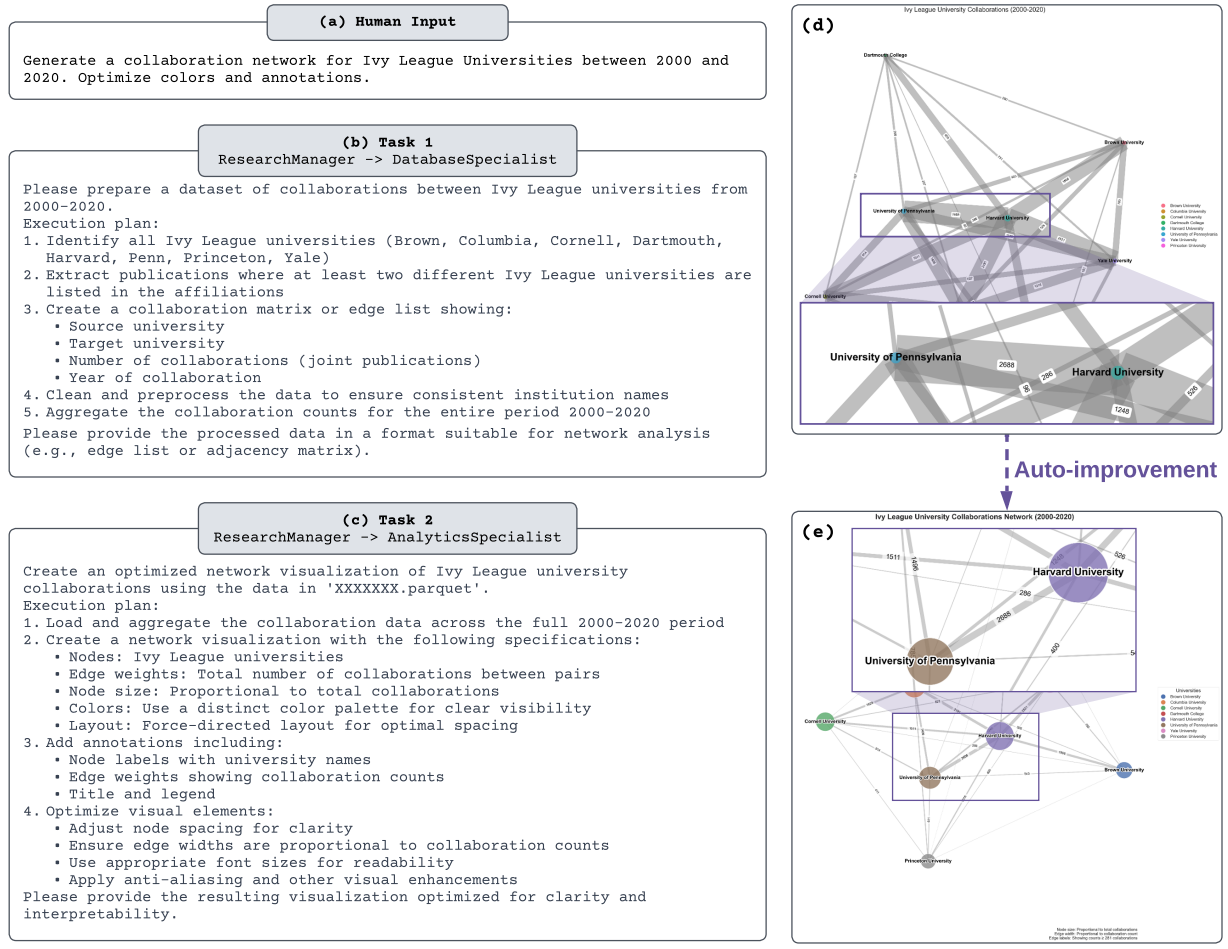


Figure 2: *SciSciGPT*'s visualization of Ivy League university collaborations. In response to the human input (a), the *ResearchManager* decomposed the request and then delegated the data extraction task (b) and visualization task (c) to the *DatabaseSpecialist* and *AnalyticsSpecialist,* respectively. The *AnalyticsSpecialist* created an initial visualization (d), and the system refined the figure through two rounds of improvements to generate a final

visualization (e) with an enhanced color scheme, proportional node sizes, and clearer text annotations. Note: The zoomed window in (e) was added manually for clarity.

The workflow began with the *ResearchManager*, which identified key requirements for the request, including data acquisition, network construction, and visualizations based on reasoning through meta-prompting[63]. The *ResearchManager* agent then broke down the input question into high-level tasks to delegate to other agents. First, it asked the *DatabaseSpecialist* to prepare a collaboration dataset with a specified data schema and provided a list of executable steps, including identifying pairs of Ivy League universities, filtering by publication time, and cleaning and aggregating the data (see Fig. 2b and Chat History in SN 5.1 for more details). In response, the *DatabaseSpecialist* executed this task in three steps: (1) it explored the database to identify relevant schemas and tables; (2) it used specialized tools that standardized the university names to ensure consistent institutional identification; and (3) it wrote the SQL queries and queried the data through complex SQL operations with Common Table Expressions (CTEs). After conducting this data extraction procedure and structuring the data, the *DatabaseSpecialist* saved the extracted data to a temporary file.

As the *DatabaseSpecialist* moved through this process, the *EvaluationSpecialist* assessed the agent's performance after each step, giving it a score as well as suggestions for improvements. For example, the *EvaluationSpecialist* gave the first tool call a score of 0.8, which is high enough for the agent to continue to the next step. Once the *DatabaseSpecialist* completed the entire task, the *EvaluationSpecialist* performed a more systematic assessment of the specialist's workflow, providing an overall score and generating a detailed report that reviewed the delegated task, documented key methodological choices and challenges, and assessed the quality of its output. The *EvaluationSpecialist* then forwarded the complete workflow and assessment report to the *ResearchManager*.

After receiving the assessment report, the *ResearchManager* delegated the visualization task to the *AnalyticsSpecialist*, instructing it to use the extracted data and providing a list of actionable steps for loading the data, constructing and visualizing the network, and optimizing the annotation and visual elements (Fig. 2c). The *AnalyticsSpecialist* then initiated a dynamic visualization workflow, using Pandas for data loading, NetworkX for graph construction, and Matplotlib to create the initial visualization. As with the *DatabaseSpecialist*'s work, the *EvaluationSpecialist* provided a multimodal assessment of each step, with a caption, feedback, score, and suggestions for improvements that the *AnalyticsSpecialist* could use to redo the visualization. After the first visualization attempt, for instance, the *EvaluationSpecialist* gave it a score of 0.75, indicating that a revision was needed, and suggested improvements to edge weights, labeling, and annotations. The *AnalyticsSpecialist* used this iterative refinement and debugging process across multiple cycles to continuously enhance the figure, improving the size of elements, colorization, annotations, legends, and other aesthetic parameters. Fig. 2d,e presents the *AnalyticsSpecialist*'s first visualization attempt and its output after two more iterations of this automated refinement process. As this last figure received a high score of 0.85 from the *EvaluationSpecialist*, the *ResearchManager* determined that no additional tasks were necessary and finalized the response, summarizing the workflow and synthesizing a final answer for the user.

In this case study, *SciSciGPT* successfully processed and visualized collaboration patterns among Ivy League universities, producing a network visualization that communicates both institutional productivity through node sizes and collaboration intensity through edge weights. The case study highlights not only *SciSciGPT*'s automation of complex workflows, but also its ability to execute quality checks and refine its results through iterative improvements.

Just as researchers using conventional data science methods often develop follow-up questions after considering their initial findings, researchers may have additional questions after examining *SciSciGPT*'s output. In this case, for example, a researcher might be interested in a more in-depth exploration of the research fields involved in Ivy League collaborations. Or they may be interested in writing an op-ed on university collaboration using these findings. SN 5.1 presents *SciSciGPT*'s responses to these follow-up questions.

Case study #2: Multimodal replication of existing findings

(See full chat history in SN 5.2)

Now imagine another researcher who is reading a SciSci paper and is curious about the interpretation and replication of the findings. This scenario is typical for researchers at various career stages. For example, active researchers who want to build on a particular finding often begin by replicating key results, and junior researchers who are just entering the field frequently find that replicating the primary findings serves as a valuable learning exercise. More broadly, the growing emphasis on open science[64,65] has made the replication of existing results and findings increasingly important.

In this case, imagine the researcher is reading the paper, "*Large Teams Develop and Small Teams Disrupt Science and Technology*"[66], and they are intrigued by its main finding, depicted in Fig. 2a of the paper[66]. The figure shows that median citations increase with team size while the average disruption percentile decreases with team size. Recognizing *SciSciGPT*'s multimodal capabilities, we simply take a screenshot of the figure, upload it to *SciSciGPT*, and give it the following prompt to instruct it to interpret and replicate the findings using data from its repository:

Human Input
`Interpret this figure. Redo the analysis using your database. Create a similar visualization.`

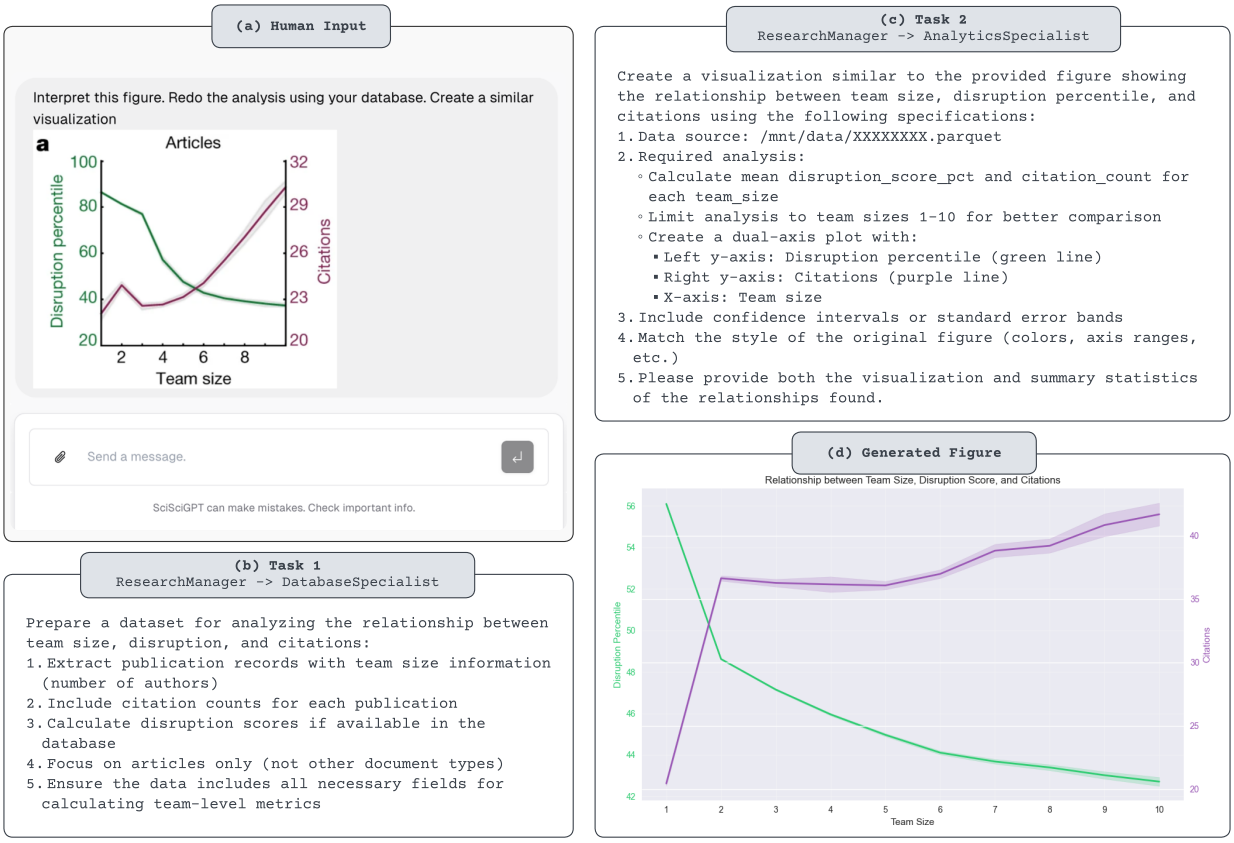


Figure 3: *SciSciGPT*'s replication of a figure from a published paper. The user input (a) includes Figure 2(a) from Wu et al.[66] and instructions to interpret the figure, redo the analysis, and create a similar visualization. *SciSciGPT* broke the request down into tasks (b) and (c) for the *DatabaseSpecialist* and *AnalyticsSpecialist*, respectively. (d) presents *SciSciGPT*'s final output. While the exact data points in the two figures differ due to variations in database timeframes and geographical coverage, *SciSciGPT* successfully replicated the trade-off between citation impact and disruption.

After receiving the figure and replication request (Fig. 3a), *SciSciGPT* coordinated a systematic response. First, the *ResearchManager* examined the figure, assessing the technical elements (the dual-axis visualization), trend patterns, and confidence intervals. It then broke down the user request into specific tasks and delegated the data extraction task to the *DatabaseSpecialist* (Fig. 3b).

The *DatabaseSpecialist* surveyed all available data tables and examined their schema. After mapping the database architecture, the *DatabaseSpecialist* crafted SQL queries to extract data from more than 9 million papers, including their citations, disruption percentile measures, team sizes, and other relevant metrics, storing them in a temporary parquet file. After the *EvaluationSpecialist* assessed these steps, returning a high score of 0.95, the *ResearchManager* directed the *AnalyticsSpecialist* to recreate the dual-axis visualization (Fig. 3c).

The *AnalyticsSpecialist* responded by loading the parquet file from *DatabaseSpecialist* and using it to calculate the average impact by team size, with confidence intervals, and create the visualization (Fig. 3d). The *EvaluationSpecialist* systematically considered the data

representation, visual design, scientific insight, and technical execution. As the *EvaluationSpecialist*'s rating met the threshold for continuation, the *AnalyticsSpecialist* proceeded to calculate additional statistics describing the relationship between team size, citation impact, and disruption scores, including correlation coefficients and the percentage change. The *ResearchManager* then synthesized the final results of this analysis and visualization task for the user.

Here again, after receiving these results, a researcher may have various follow-up questions. A researcher may be interested in further examining the initial result using more advanced statistical methods. For example, they might consider using OLS regression and Propensity Score Matching (PSM) to investigate whether the result still holds after controlling various confounding factors (see SN 5.2). Or they may be interested in replicating the same visualization using *SciSciGPT*'s data but calculating the impact metrics, like disruption scores, from scratch during runtime rather than allowing *SciSciGPT* to use its pre-defined impact metrics from the SciSciNet database for computational simplification. In this case, the researcher can simply instruct *SciSciGPT* to compute the disruption score, explaining the calculation using natural language (see SN 5.3).

Together, these case studies demonstrate how *SciSciGPT*'s multi-agent framework orchestrates diverse functionalities, including understanding user requests, breaking down tasks into concrete steps, retrieving relevant data, analyzing data, creating visualizations, comprehending the literature, evaluating its performance, and making iterative improvements.

4 - Methods

4.1 - SciSciGPT architecture

SciSciGPT supports efficient data-driven insight extraction by integrating three modules:

1. **Database repository.** The database repository includes (1) a scholarly data lake organized into a relational database (i.e., Google Big Query), and (2) a corpus of SciSci publications that we have chunked, embedded, and organized into a vector database.
2. **Multi-agent AI system**. *SciSciGPT* is built on a hierarchical multi-agent SciSci collaborator framework. This multi-agent system serves as its core (Fig. 1).
3. **Web interface.** The user-friendly chat interface at https://sciscigpt.com enables users to collaborate with the AI system through multi-turn conversations to generate insights, refine analyses, and reach empirically validated conclusions.

We describe the architecture in greater detail below.

4.2 - Database repository

SciSciGPT's data infrastructure enables seamless interaction with scholarly data lakes to support data analysis. It is designed to build on comprehensive databases such as SciSciNet[9] or OpenAlex[10], open-source scholarly data lakes that encompass most of the data and linkages needed for SciSci research, and to integrate with SciSciCorpus, a curated database of literature in the field. *SciSciGPT* also maintains the ability to integrate with other data sources[10,11,13,67].

SciSciNet encompasses over 134 million scientific publications and millions of external linkages to funding sources and public uses. As such, it contains data capturing the essential elements of scientific research, including publications, authors, affiliations, upstream funding, and downstream impacts. We use Google BigQuery, a cloud-based, high-performance relational database, to manage SciSciNet's interconnected data tables. We implemented several refinements to the SciSciNet database to enhance its integration with *SciSciGPT*. First, since *SciSciGPT* is currently a prototype, we limited the data scope to papers published in the United States to optimize computational efficiency. Further, we structured the database into 19 tables to ensure that it accurately reflects the relationships between entities, and we wrote and incorporated descriptions that map tables and columns to established SciSci concepts to enable *SciSciGPT* to interpret the data. The resulting repository encompasses more than 11 million research papers, 78 million citation relationships, and numerous other quantifiable metrics of scientific activity. Fig. 4 presents the database architecture.

SciSciCorpus contains a corpus of publications as a vector database that the system uses to access prior knowledge in the field. To create SciSciCorpus, we assemble a collection of SciSci papers, download PDFs, and employed GROBID[68] (GeneRation Of BIbliographic Data) to extract and parse the full text into natural paragraphs. We then used the OpenAI API to generate 2-3 sentence summaries of each paragraph and classified each paragraph into one of a predefined set of categories, such as abstract, methodology, results, and discussion. This

taxonomic structure, while not necessarily aligned with the organization of the original document, provides a standardized framework for *SciSciGPT*'s content navigation. Each paragraph is then projected into an embedding space and indexed into a vector database for effective RAG during runtime. SN 1 contains more details regarding the processing procedures and schemas for these databases. Note that the inclusion of SciSciNet and SciSciCorpus serves as an initial framework, and given the open-source nature of SciSciGPT, users may adjust, replace, or extend this corpus to align with their specific preferences and research use cases.

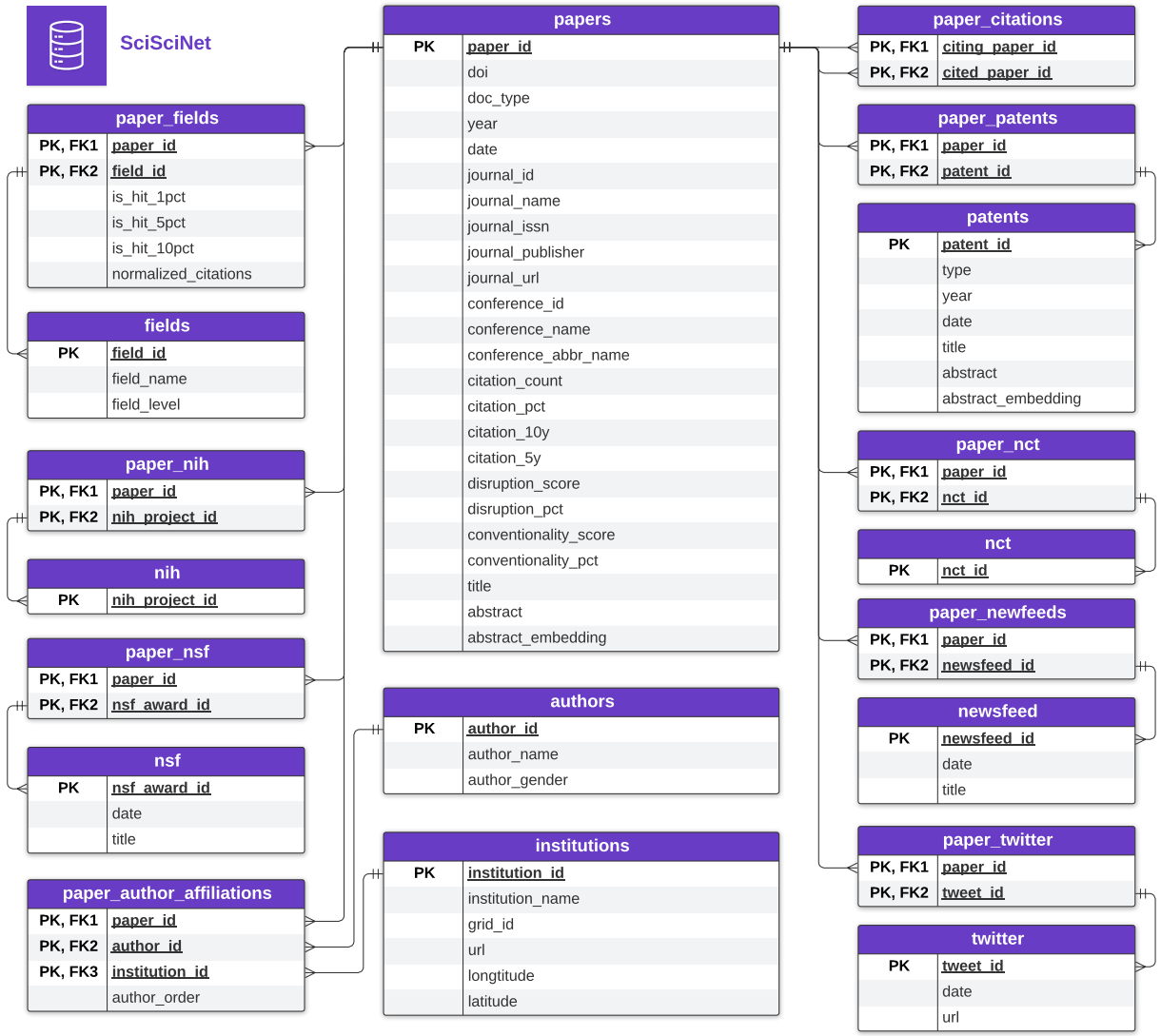


Figure 4: Schema diagram of the variant of SciSciNet used in *SciSciGPT*. *SciSciGPT* connects to this data lake, which serves as its primary repository of scholarly data. This version of SciSciNet features a refined schema and enhanced paper and patent data.

4.3 - Multi-agent AI system

SciSciGPT includes a *ResearchManager* and four specialist agents.

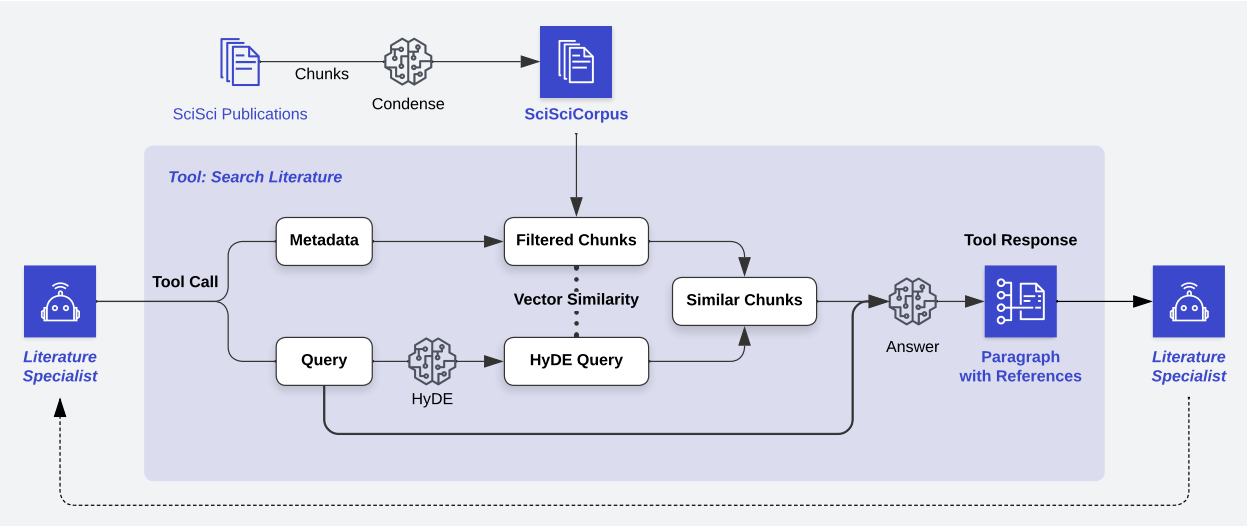


Figure 5: Architecture of the *LiteratureSpecialist* agent for retrieval-augmented generation (RAG) in SciSci research. The tool processes SciSci publications in multiple stages. When called, the tool accepts two inputs: optional metadata filters (year, author, section type, paper title) and a search query. The query is processed through HyDE (Hypothetical Document Embedding) to generate an enhanced search query. Both the HyDE query and metadata filters are used to retrieve filtered chunks from the corpus. Vector similarity retrieval identifies the top K most relevant chunks, which are then processed by a language model to generate a comprehensive response paragraph with appropriate references.

LiteratureSpecialist. Understanding and contextualizing research questions within the SciSci domain is critical for determining the novelty of the research question and ensuring efficient use of existing knowledge, including previous approaches to similar scientific questions, conclusions from prior studies, and other researchers' assessments of the implications of their findings.

We designed the *LiteratureSpecialist* to facilitate literature understanding and contextualize *SciSciGPT*'s workflow within the SciSci research domain using the `literature_search` tool for RAG. Given a search query, the tool first filters papers using potential meta-data parameters identified by the LLM from the query (e.g., section=Abstract). It then retrieves chunks from SciSciCorpus by text embedding similarity between the query and the corpus; identifies the most relevant papers; and summarizes the retrieved chunks into paragraphs with references in response to the search query. As a tool designed to support a multi-step RAG workflow, the LLM typically dynamically and iteratively invokes it to focus on different levels of paper information. For example, it may first analyze abstracts and then progressively delve into other key sections (e.g., methodology, results, discussion) to deepen its understanding of the literature. Through this step-by-step process, the tool gradually generates a summary paragraph that synthesizes the current SciSci research relevant to the query, which is output to the user and stored in context memory to guide subsequent activities. Through RAG, *LiteratureSpecialist* not only provides *SciSciGPT* with domain knowledge but also ensures accurate and verifiable scholarly references (see SN 5.5. for illustrative case studies).

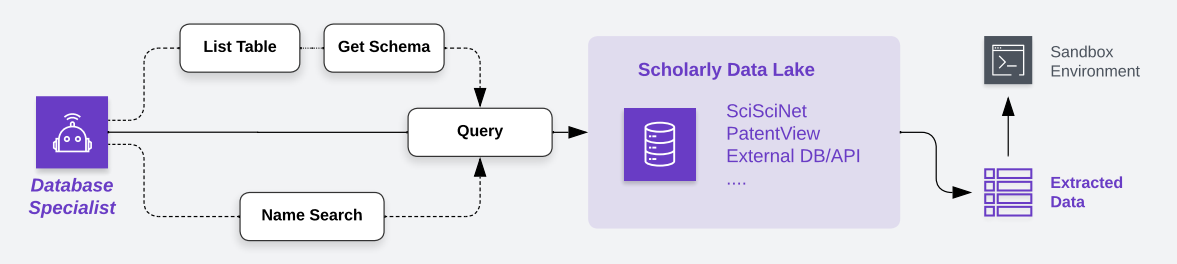


Figure 6: An example of the *DatabaseSpecialist*'s workflow for data extraction.

DatabaseSpecialist. Understanding the complex data structure in the SciSci domain is essential for bridging abstract concepts and relationships in the research question to specific data. We designed the *DatabaseSpecialist* to comprehend the intricate SciSci data lake, extract relevant data, and preprocess it through data cleaning and transformation. This agent incorporates a suite of specialized tools: (1) `sql_list_table` retrieves all available table-level descriptions, helping with database navigation, (2) `sql_get_schema` provides detailed structural information for specified tables, including column specifications, data formats, and formatted sample rows, (3) `sql_query` executes the SQL queries generated by the agent, returning a preview of the fetched data frame (top k rows and column names) and a temporary file path for further use, and (4) `name_search` performs embedding-based similarity matching within a vector database to identify the most semantically relevant entities based on the user's query. This tool is necessary because key entities in the SciSci field–such as scientific fields and research institutions–are often referred to by multiple names, making standardization crucial for accurate analysis. With these tools, the *DatabaseSpecialist* can comprehend both the delegated tasks and the SciSci data structure to extract relevant data segments for further analysis. Overall, this design enables *SciSciGPT* to navigate the scholarly data lake commonly used in SciSci research.

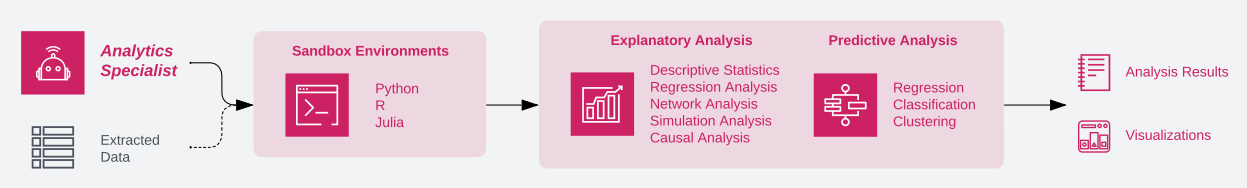


Figure 7: An example of the AnalyticsSpecialist's workflow for analysis and visualization.

AnalyticsSpecialist. Once *SciSciGPT* has established an understanding of the relevant SciSci literature and the data lake, it needs to conduct the analysis and derive insights. As SciSci is an inherently multidisciplinary field, research in this area requires familiarity with a diverse range of computational methods, from basic statistical techniques (e.g., descriptive and regression analysis) to advanced modeling approaches (e.g., machine learning). Thus, we designed the *AnalyticsSpecialist* to implement appropriate methodologies, write and execute code to conduct the analysis, and generate insights through text and visualizations that are tailored to the user's query. The agent integrates three open-source tools within isolated, stateful sandboxes to enable efficient code execution, debugging, and refinement: (1) `python` offers extensive machine

learning frameworks and general-purpose programming capabilities, (2) `r` provides robust statistical computing and visualization libraries, and (3) `Julia` provides high-performance scientific computing capabilities with concise syntax. Together, these tools equip the agent with comprehensive analytics toolkits, allowing it to write and execute code and create new analyses.

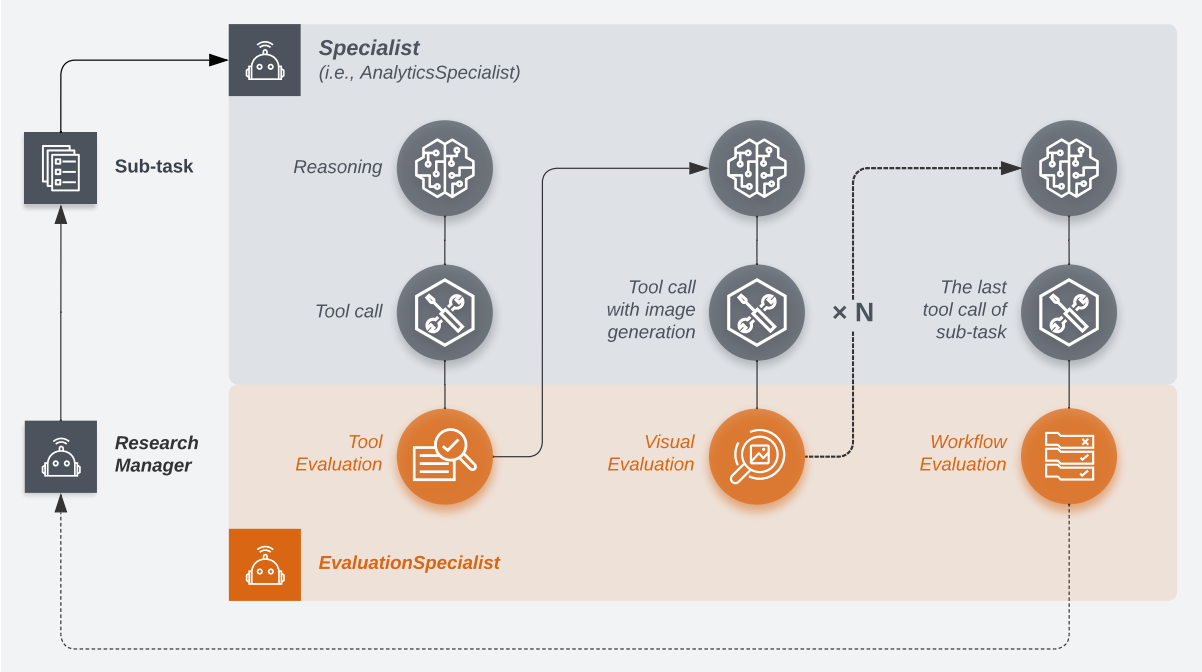


Figure 8: An example of the *EvaluationSpecialist*'s workflow for multi-level self-evaluation. After the *ResearchManager* assigns a task to a specialist agent (e.g., *AnalyticsSpecialist*), and the specialist begins the task, the *EvaluationSpecialist* systematically evaluates the specialist's tool calls. The evaluation process occurs in three stages: (1) the tool evaluation assesses the success of individual tool executions; (2) a visual evaluation systematically assesses any visualization that is generated; and (3) a final workflow evaluation examines the complete execution chain after the specialist finishes the task. The *EvaluationSpecialist* provides the specialist agent with feedback at each stage, including a score and specific suggestions for improvement when needed. It also provides the *ResearchManage*r with a task report.

***EvaluationSpecialist*.** To ensure the quality and reliability of these AI-generated analyses, processes, and findings, we designed the *EvaluationSpecialist* to conduct multi-level self-evaluations, which include tool evaluations, visual evaluations, and task evaluations. The tool evaluation assesses each specialist's tool usage by analyzing the task context, including the task assigned by the *ResearchManager*, the workflow history, the tool parameters, and the tool response. The visual evaluation assesses any visualization that is generated, typically by the *AnalyticsSpecialist*. The *EvaluationSpecialist* examines the figure comprehensively, considering its alignment with the task, the data it uses, and its adherence to visual design principles. The visual evaluation results in a list of suggestions for potential improvements that the specialist agent can use to refine the visualization. And the task evaluation analyzes the entire workflow after a specialist agent completes its task, and no more tool calls are created. The *EvaluationSpecialist* then provides a comprehensive execution report to the *ResearchManager*.

For each of these assessments, the *EvaluationSpecialist* assigns a reward score to guide the other agents' next steps. Depending on the score, *SciSciGPT* either continues with its current approach, makes minor adjustments, or backtracks for major revisions. This multi-level self-evaluation mechanism ensures that *SciSciGPT* maintains quality control throughout complex research tasks.

4.4 - Implementation

Meta-Prompting for Reasoning: *SciSciGPT* uses meta-prompting to facilitate the reasoning chain in slow-thinking LLMs[69–71], enhancing their ability to engage in deeper, more structured analytical processes. This approach incorporates two key functionalities: (1) *structured reasoning*, which guides logical step-by-step analysis, and (2) *verbal reinforcement learning*, which refines responses through iterative feedback and adaptation. Structured reasoning requires *SciSciGPT* to use a comprehensive tag taxonomy with predefined XML-style tags, such as `<thinking>`, `<step>`, `<reflection>`, `<answer>`, `<count>`, and `<reward>`, which represent distinct cognitive stages in the LLM's reasoning process. These labels ensure that responses are organized into clear, logical, and maintainable steps. By contrast, verbal reinforcement learning enables *SciSciGPT* to adjust its progress based on the reward score it receives from the *EvaluationSpecialist*. We provide detailed meta-prompts for all agents in SN 2.

Contextual Memory Management: Given *SciSciGPT*'s extensive workflows, the multimodal input and output, and the iterative feature for progressive research workflow, it must maintain focus and efficiency during iterative and resource-intensive multi-turn literature retrieval or data-driven insight exploration. As long-context conversations pose significant challenges to LLMs, *SciSciGPT* employs several mechanisms to compress the context, optimize prompt quality, minimize redundancy, and improve computational efficiency[72,73] by pruning content less relevant to ongoing reasoning: (1) The *ResearchManager* acts as the system's coordinator and maintains a complete record of all interactions with users and specialists. This comprehensive view allows it to track methodological choices, determine relevant prior context, and provide specialists with precisely the right excerpts to facilitate coherent long-range reasoning. When delegating tasks, the *ResearchManager* controls the visibility of history to the specialist, selecting the relevant portions of the specialist's earlier dialogue so it can build seamlessly on past computations. (2) Each specialist in *SciSciGPT* operates independently. A specialist's workflow begins with receiving tasks from the *ResearchManager*, proceeds through iterative reasoning and tool calling, and concludes by returning results to the *ResearchManager*. Specialists cannot access another specialist's workflow or the interaction between the user and *ResearchManager*. (3) The <thinking> tag functions as a scratchpad for inner monologue, allowing agents to engage in detailed reasoning. This monologue remains invisible to other agents. Since the *ResearchManager* is designed to maintain a comprehensive awareness of all exchanges between users and specialists, this inner monologue is pruned for the *ResearchManager*. Using a compact context helps manage token limitations and shields the *ResearchManager* from unnecessary reasoning details. (4) Rather than presenting raw images of all generated figures in the context, *SciSciGPT* transforms the modality of all generated

figures into a textual representation using the capability of *EvaluationSpecialist* to output structured textual summaries of generated figures (i.e., retaining the `<evaluation>` and `<caption>` outputs).

Web Interface for Collaborative Research: As an AI collaborator, *SciSciGPT* behaves like a chatbot, building context from the sequential accumulation of messages generated by the user, agents, and tools.-SciSciGPT's conversational web interface is a standard chat interface, like ChatGPT, with account management, persistent history, and multi-modal support for text, code, visualizations. This design enables users to iteratively refine or expand their queries and explore insights through the back-and-forth interaction for scientific workflows. To further illustrate these human-AI interactions, we include a few examples and analyses in SN 5.4, showcasing iterative clarifications and corrections, and demonstrating *SciSciGPT*'s human-AI collaborative process.

5 – LLM Agent Capability Maturity Model

While *SciSciGPT* focuses on the science of science as a testbed, its architecture design suggests broader applicability across data-intensive domains, especially those in computational social science. To better understand its generalizability, we propose an LLM agent capability maturity model, building on key concepts from system development[74–76], which allows us to formalize essential progression stages for AI research collaborators through a four-tiered developmental roadmap. This roadmap not only guides the current designs of *SciSciGPT*, which is a proof-of-concept for this capability maturity model, but also provides further pathways for enhancement.

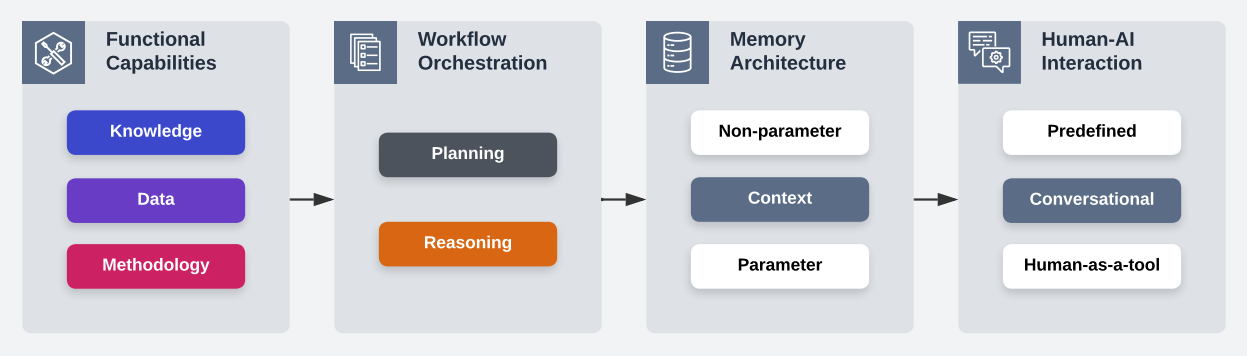


Figure 9: LLM Agent Capability Maturity Model: A four-level progression framework showing 1) Functional Capabilities - extending LLMs through specialized tools for knowledge access, data processing, and methodology implementation; 2) Workflow Orchestration - implementing planning and reasoning mechanisms for complicated research task; 3) Memory Architecture - maintaining information persistence, adaptation, and customization throughout multiple interactions; and 4) Human-AI Interaction - defining different modes of system engagement. Colored blocks highlight components implemented in SciSciGPT, balancing technical complexity with research effectiveness.

We envision four progressive maturity levels that define increasingly sophisticated AI capabilities (See SN 3 for details). At the first level, functional capabilities extend LLMs beyond text generation through specialized tools for domain knowledge access, data processing, and

statistical methods implementation. In *SciSciGPT*, these capabilities are fundamental elements of the specialized agents. At the second level, workflow orchestration introduces planning and reasoning mechanisms. In *SciSciGPT*, planning is exemplified by our *ResearchManager*-specialists architecture, which decomposes tasks along modular research functions, analogous to different categories of domain research tasks. Reflective reasoning is enabled by meta-prompting and the *EvaluationSpecialist*. At the third maturity level, memory architecture maintains the overall information environment throughout the research processes, enabling agents to use previous interactions and histories to facilitate adaptation and customization based on their specific needs. *SciSciGPT* implements selectively controlled prompt and context management to maintain focus and efficiency across progressive explorations. Finally, at the fourth level, human-AI interaction is made possible by the interactive components of the systems, facilitating progressive conversational research workflows.

As a proof of concept of this capability maturity model, *SciSciGPT* selectively implements core components at each level (highlighted as colored blocks in Fig. 9), while balancing implementation complexity against practical utility and prioritizing research effectiveness over maximum technical sophistication. As AI agents increase their capabilities and reach, the model presented in Fig. 9 may serve as a useful roadmap to facilitate more comprehensive human-AI collaborations.

6 - Expert Review

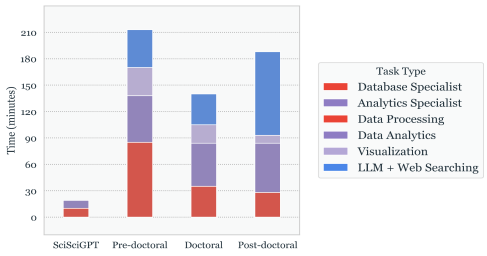
We conduct a preliminary assessment of *SciSciGPT*'s effectiveness, efficiency, and usability as an AI research collaborator through (1) an exploratory pilot study that compares its response time and accuracy to those of human researchers answering the same research questions and (2) semi-structured interviews with SciSci experts after introducing them to the system.

6.1 – Exploratory pilot study

We compared the performance of *SciSciGPT* with that of three domain researchers with different levels of expertise (pre-doctoral, doctoral, and post-doctoral) to develop an initial assessment of the system's effectiveness and efficiency. The participants reported an average of 3.7 years of data science experience and 1.7 years of experience in SciSci research. We provided the participants with identical environments (datasets and Python/R coding platforms) and communicated the task by giving them the same inputs we gave to *SciSciGPT*. They were permitted to use all their standard research tools, including web resources, existing codebases, LLMs for coding, and IDE plugins, but were not permitted to use *SciSciGPT*. Because participants completed the same research tasks in their preferred existing data science and coding environments (e.g., Claude 3.5, GPT-4o, ChatGPT-o1), this comparison is best understood as human researchers using general-purpose AI tools versus *SciSciGPT*, rather than simply human researchers versus *SciSciGPT*.

After the participants completed the tasks, we invited three postdoctoral researchers to review the participants' results and *SciSciGPT*'s output, assessing each with a five-point scale (with

higher scores indicating greater effectiveness) across five dimensions: effectiveness, technical soundness, analytical depth, visualization quality, and documentation clarity.



	SciSciGPT	Pre-doctoral participant	Doctoral participant	Post-doctoral participant
Overall effectiveness	4.3	3.8	3.5	3.8
Technical soundness	4.3	3.5	3.3	3.5
Depth of analysis	4.5	3.3	3.3	3.8
Visualization quality	4.2	3.3	3.3	3.7
Documentation clarity	4.5	2.7	2.7	3.0

Figure 10: Exploratory research efficiency and effectiveness comparison. a) Time allocation across workflow components (data processing, analytics, visualization, LLM + web searching). b) Average postdoctoral evaluator workflow effectiveness rating on a five-point scale for each participant and SciSciGPT.

Figure 10 presents the time-to-completion across all tasks for *SciSciGPT* and the human participants, as well as the postdoctoral reviewers' average ratings for each dimension of our research quality assessment. In this exploratory study, *SciSciGPT* accelerated the research process, completing the same tasks in about 10% of the average time required by experienced researchers in the field. Notably, all participants utilized LLMs to assist with coding tasks during the study, making this comparison particularly relevant for understanding *SciSciGPT*'s contributions to modern research workflows. Results suggest that *SciSciGPT* completed tasks more efficiently, probably due to its integration of multiple analytical steps, from data processing to analysis to visualization and iterative refinement.

More importantly, when evaluating the quality of work, the three expert evaluators found that *SciSciGPT*'s output was stronger than the human researchers' work across various dimensions we examined. We acknowledge, however, that participants may have been operating under task-completion constraints, and that their results may not reflect their full capabilities, especially in unconstrained research settings with unlimited time for refinement. Moreover, we note that given the limited sample size, these results should be interpreted as exploratory rather than offering generalizable insights. Nevertheless, these exploratory results suggest that *SciSciGPT* may outperform experienced researchers for tasks requiring a 2–3-hour completion time, delivering superior outcomes across multiple quality dimensions and in much less time.

As part of their review, evaluators noted that *SciSciGPT* produced excessively detailed documentation. On the one hand, this extended the evaluators' reading time and increased their cognitive load, potentially leading to a sub-optimal user experience. On the other hand, the detailed documentation highlights a key advantage of human-AI collaborations enabled by systems like *SciSciGPT*, where each step of the analysis is meticulously documented and can be revisited later or by other researchers as needed, substantially facilitating the reproducibility of research. Overall, the lengthy documentation underscores a trade-off between comprehensiveness and brevity and highlights the need for further refinement.

6.2 - Semi-structured interviews

In addition to the exploratory pilot study, we gathered qualitative insights about *SciSciGPT* from three expert SciSci researchers (E_A, E_B, and E_C). We first conducted a 10-minute walkthrough of the system's architecture and core functionalities, and we then gave the experts 30 minutes for system exploration in which they experimented with *SciSciGPT* and asked clarifying questions. Next, we conducted 60-minute semi-structured interviews using a standardized questionnaire (see SN 4). The interviews probed the experts' research practices, as well as their thoughts on *SciSciGPT*'s database repository, the AI capabilities, and the human-AI collaboration workflow it enables. All sessions were recorded and transcribed for analysis, with key findings summarized below.

We began by exploring the experts' current research processes to identify potential *SciSciGPT* integration points. All three reported using standard computational tools: Jupyter Notebook/RStudio with Pandas, literature search tools like Google Scholar and Elicit, and web-based coding assistance tools like ChatGPT and Claude.

When discussing research challenges, experts consistently highlighted data management as a primary pain point, with E_C noting, *"Loading large datasets is annoying. Also dealing with messy data."* E_A expressed frustration with traditional data processing workflows, describing tasks like *"loading large CSV, TSV into memory"* and data cleaning as time-consuming bottlenecks. Our experts estimated that the integrated SciSciNet dataset can be used to address the vast majority of SciSci research questions, highlighting the comprehensiveness of the data coverage. At the same time, they also suggested ways to further improve the data coverage, including expanding SciSciNet with additional public databases and enabling seamless integration of external and user-uploaded data.

All experts found *SciSciGPT* valuable for early-stage data exploration and prototyping. Experts agreed that the *ResearchManager*, as the central controller of the multi-agent framework, effectively decomposed user questions into manageable tasks. E_A noted that *DatabaseSpecialist* operates "*faster than humans*" with generally reliable results. The *EvaluationSpecialist* received particularly strong positive feedback for its visualization assessment capabilities, with E_B noting that it "*spots problems*" and "*generates helpful suggestions about visualization clarity*." The experts also commended the *LiteratureSpecialist*'s ability to generate logical iterative workflows.

After engaging with the system, our experts also identified occasional failure cases. E_B found instances of database downsampling through unnecessary `LIMIT` clauses in BigQuery. They also observed coordination issues. For example, if the *DatabaseSpecialist* failed to collect necessary data, the *AnalyticsSpecialist* could produce unreliable outputs. E_C found that the *AnalyticsSpecialist*'s analytical choices occasionally deviated from their personal preferences and field conventions. They also noted *SciSciGPT*'s inability to implement advanced statistical models, like exponential random graph models (ERGM). These specific instances highlighted areas for future improvements.

All experts considered *SciSciGPT*'s interactive features important, reporting that they particularly value the ability to ask follow-up questions, clarify intentions, explore topics in-depth, and request more explanations of previous responses. However, the presentation of the system's research workflow documentation received mixed feedback. While all experts agreed on the necessity of complete workflow transparency, they diverged in the appropriate level of information granularity. E_A and E_C expressed concern that the system response could be overwhelming. For example, E_B explained, "*Details are good, but maybe it's a little too much. But it's generally good. It would be better if it were collapsible and expandable."* Overall, the experts recommended clearer differentiation between the types of information (e.g., content from specific agents or tools) and the levels of information. They suggested, for example, that the system could default to collapsing the detailed reasoning chain and code snippets for a more streamlined presentation.

Guided by this feedback, we design a new function that allows the system to streamline the information display while preserving the underlying fine-grained logs necessary for reproducibility and transparency. The main idea is that because *SciSciGPT* emits its internal state in XML with semantic tags, we can use this machine-readable structure to demonstrate information to the user selectively, automatically folding or hiding low-level details by default to reduce cognitive load while preserving transparency. This way, the newly designed interface allows collapsible toggles but does not discard any provenance; motivated readers can still access all the workflow information as needed. This design allows us to decouple audit sufficiency (which remains complete) from the information density shown to users (now adjustable).

Our experts also raised important concerns. "*I feel uncomfortable trusting something not generated by myself. As a researcher, I'm responsible for all mistakes. Ultimately, it will be my name on the paper."* They compared working with the AI collaborator to pre-doctoral assistants; both require explicit guidance. While they appreciated *SciSciGPT*'s greater transparency compared to human collaborators, they emphasized the substantial effort required to validate the system's results. Ultimately, trust appears to be an important factor in collaboration—whether it is with a human or AI.

Finally, we conducted a preliminary qualitative assessment by these three domain experts to review every intermediate output generated in the two case studies. After a brief walkthrough of the *EvaluationSpecialist*'s logic, each expert rated the corresponding *ToolEval*, *VisualEval*, and *TaskEval* on a five-point scale (1 = poor; 5 = excellent) and annotated points of disagreement or confusion. Across all assessments for the *EvaluationSpecialist*, the mean scores are 4.98 for *ToolEval*, 4.17 for *VisualEval*, and 5.00 for *TaskEval*. According to the Experts, *ToolEval* and *TaskEval* appear to consistently identify reasonable areas for improvement and assign credible scores. By contrast, *VisualEval*'s relatively lower average seems to at least in part reflect the multimodality challenges commonly faced by LLMs. Overall, these results suggest that *EvaluationSpecialist* seems to align with expert judgment on textual capabilities; its *VisualEval* component shows comparatively less alignment, partly due to multimodal LLM capabilities,

which are currently less mature than textual capabilities, though have been improving over time. SI 4.3 provides more details about experts' feedback about the *EvaluationSpecialist*.

While a broader evaluation is necessary to strengthen these findings, our preliminary assessments highlight *SciSciGPT's* ability to leverage multiple LLM functionalities to streamline SciSci research processes. At the same time, our expert reviews and evaluations also suggest several ways that *SciSciGPT* can be further enhanced, including 1) the adaptation of ongoing LLM advancements, such as large reasoning models and reinforcement learning-based post-training on related tasks; 2) architectural improvements that integrate enhanced RAG techniques and improve the documentation of methodological choices; 3) database module improvements that incorporate broader data sources and support user data imports; and 4) interface refinements, including options to adjust the information granularity of implementation details and more flexible visualization options. Overall, our evaluations in this paper are exploratory by nature. While systematic evaluations of alignments are beyond the scope of this work, they represent an important area of future work, potentially as part of a broader exploration into automated research evaluation frameworks.

7 - Discussion

Taken together, by automating technical workflows, *SciSciGPT* reduces research task completion time from hours to minutes, allowing researchers to focus on the creative and interpretive aspects of their work. This seems particularly beneficial in early-stage research, idea generation, and verification processes. Beyond time savings, *SciSciGPT* lowers technical barriers to entry, broadening participation in the field by enabling those with basic domain knowledge but limited technical skills to explore data more effectively. The acceleration of research and broadening of participation has the potential to shift how researchers work and collaborate.

SciSciGPT is intentionally constructed upon the API of existing commercial LLMs (e.g., currently Anthropic Claude). As a wrapper, *SciSciGPT* directly inherits the powerful baseline capabilities and continual improvements of its underlying backbone model. Thus, *SciSciGPT* naturally performs what general-purpose chatbots like ChatGPT can perform, including general question-answering, summarization, coding, and reasoning. Additionally, our modular architecture, as well as the open-source nature, affords users the flexibility to incorporate new, improved models in the future, ensuring that *SciSciGPT* benefits directly from advances in the broader LLM ecosystem.

One key advantage of *SciSciGPT* is to empower human researchers at early, exploratory stages of the research process, rather than replacing their expertise or independently producing publication-ready outputs. Consequently, our case studies were deliberately chosen to represent typical early-stage explorations that researchers in the SciSci community frequently undertake. While these initial analyses may appear relatively simple, they illustrate how *SciSciGPT* supports rapid prototyping, iterative idea refinement, and exploratory analyses—tasks that traditionally involve significant manual effort and time. These iterative interactions also underscore a broader design challenge: effectively managing conversational context in extended,

multi-turn research workflows. Maintaining the right balance between retaining relevant history and adapting to evolving user goals is an active research frontier for LLM agents, and one we see as an important future direction, especially as models improve. Given the interdisciplinary nature of the science of science community, researchers and practitioners vary widely in their technical backgrounds. By lowering technical barriers, *SciSciGPT* can further facilitate the participation of new entrants, broadening the range of ideas and expertise in the field.

While this paper focuses on the field of the science of science, the framework offered by *SciSciGPT* may extend to other computational disciplines. Indeed, the integration of data, research methods, and literature is not unique to SciSci, but rather, with appropriate adjustments, such AI-powered research assistants may find wide applicability in other domains, especially those that are data-intensive or span multiple disciplines. Such systems could democratize access, enable more sophisticated analyses, and empower researchers to address complex questions with greater efficiency and effectiveness. Indeed, adapting the designs and thinkings behind *SciSciGPT* to other fields represents an important area for future work, which may require domain-specific adjustments to its specialist agents and underlying datasets, and the modular nature of the key components may further facilitate adaptation. Given its open-source nature*, SciSciGPT* also invites broader community-driven efforts to extend and customize the framework to suit various research domains and questions. By offering a framework for human-AI collaboration, *SciSciGPT* thus represents an initial step toward a broader vision of AI-assisted discovery, which may impact a range of computationally supported fields.

The continued relevance of *SciSciGPT* depends in part on the breadth and timeliness of its underlying data, suggesting that continuously incorporating new data as they become available represents a fruitful area of future research. For example, the newly-released SciSciNet-v2 incorporates the latest 2025 OpenAlex snapshot with expanded data tables, additional fields, and other refinements. While integrating the full-scale version would require further optimization to meet runtime constraints, the system's modular architecture is designed to facilitate such expansions. It is important to note, however, even without continuous updates, *SciSciGPT* remains valuable, as much science-of-science research relies on historical data, and static datasets can retain substantial utility over extended periods. Moreover, *SciSciGPT*'s open-source nature enables the research community to contribute updates, incorporate new data types, and extend coverage, ensuring the framework can evolve through collective effort.

An important consideration for systems like *SciSciGPT* is the stability of their outputs when faced with identical prompts. Current mainstream commercial LLMs do not guarantee identify outputs, even at zero temperature, and *SciSciGPT* naturally inherits this non-determinism from its backbone models. Additional variability can arise from the external tools it invokes, such as data processing pipelines, analytical routines, and visualization algorithms. Thus, one might expect the multi-step workflows to exhibit some inherent degree of run-to-run variation.

To this end, we performed some initial experiments assessing the run-to-run variability of *SciSciGPT* (SN 2.6). Our experiments highlight the role of prompt specificity in shaping workflow consistency and provide practical guidance for both users and future system design.

These findings suggest that careful prompt engineering, explicit methodological specifications, and deliberate workflow management can help minimize variability where precision is essential. At the same time, variability itself can have value: the differences we observe across repeated runs mirror the variability seen among human researchers given the same data and instructions[77]. Such variation can expose alternative analytical pathways and surface degrees of freedom in methodological choices that might otherwise remain hidden. This raises an intriguing possibility: Whether research agents might be intentionally designed to first explore a broader solution space before iteratively converging toward a preferred approach. This perspective suggests that variability in this case is not simply a limitation to be eliminated, but sometimes can be a feature to be strategically harnessed in the early, exploratory phases of research.

It is important to reckon with ethical considerations as AI plays a greater role in research and discovery. Automating traditional research tasks like data analysis increasingly blurs the distinction between human contributions and machine-generated work, which may challenge established norms around authorship and intellectual ownership. Widespread adoption of systems like *SciSciGPT* could also have implications for early-career researchers and newcomers to the field and may hinder their ability to develop essential analytical skills, potentially leading to a research workforce less equipped to verify, challenge, or refine AI-generated insights. Moreover, research[78,79] reveals disparities in AI tool adoption across groups and fields, suggesting unequal access and adoption in the research community. Lastly, as AI systems continue to grow in relevance for researchers, it raises the question of whether such human-AI collaborations could shape the trajectory of the field, by influencing the questions researchers prioritize and the methodologies considered valid. For example, if researchers tend to prioritize problems that align with the strengths of *SciSciGPT*, other crucial areas of inquiry that are less compatible with the use of such tools may be marginalized, potentially narrowing the scope and diversity of the field over time.

Given these considerations, the development and adoption of promising AI systems like *SciSciGPT* demand careful and thoughtful approaches that preserve the human element in scientific discovery while leveraging AI to augment researchers' productivity. The human-machine partnership envisioned in *SciSciGPT* emphasizes the importance of complementing AI-driven analyses with human oversight and expertise. With time, the research community may develop guidelines and best practices to ensure accountability and maintain research integrity. By fostering a culture of transparency and collaboration, the research community can harness the potential of human-AI collaboration while mitigating its risks.

Data availability

Data necessary for reproducing the main results will be made publicly available. SciSciNet is publicly available at: https://springernature.figshare.com/collections/SciSciNet_A_large-scale_open_data_lake_for_the_science_of_science_research/6076908/1.

Code availability

Code necessary for all frontend and backend implementation is publicly available at https://github.com/erzhuoshao/SciSciGPT.

References

1. Wang, D. & Barabási, A.-L. *The Science of Science*. (Cambridge University Press, Cambridge, 2021). doi:10.1017/9781108610834.

2. Stephan, P. *How Economics Shapes Science*. (Harvard University Press, 2012). doi:10.4159/harvard.9780674062757.

3. Bush, V. Science the Endless Frontier, A Report to the President. *U.S. Government Printing Office, Washington, DC* (1945).

4. Ahmadpoor, M. & Jones, B. F. The dual frontier: Patented inventions and prior scientific advance. *Science* **357**, 583–587 (2017).

5. Yin, Y., Gao, J., Jones, B. F. & Wang, D. Coevolution of policy and science during the pandemic. *Science* **371**, 128–130 (2021).

6. Yin, Y., Dong, Y., Wang, K., Wang, D. & Jones, B. F. Public use and public funding of science. *Nat Hum Behav* **6**, 1344–1350 (2022).

7. Fortunato, S. *et al.* Science of science. *Science* **359**, eaao0185 (2018).

8. Liu, L., Jones, B. F., Uzzi, B. & Wang, D. Data, measurement and empirical methods in the science of science. *Nat Hum Behav* **7**, 1046–1058 (2023).

9. Lin, Z., Yin, Y., Liu, L. & Wang, D. SciSciNet: A large-scale open data lake for the science of science research. *Sci Data* **10**, 315 (2023).

10. Priem, J., Piwowar, H. & Orr, R. OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts. Preprint at https://doi.org/10.48550/arXiv.2205.01833 (2022).

11. Herzog, C., Hook, D. & Konkiel, S. Dimensions: Bringing down barriers between scientometricians and data. *Quantitative Science Studies* **1**, 387–395 (2020).

12. Hendricks, G., Tkaczyk, D., Lin, J. & Feeney, P. Crossref: The sustainable source of community-owned scholarly metadata. *Quantitative Science Studies* **1**, 414–427 (2020).

13. Wang, K. *et al.* Microsoft Academic Graph: When experts are not enough. *Quantitative Science Studies* **1**, 396–413 (2020).

14. Baas, J., Schotten, M., Plume, A., Côté, G. & Karimi, R. Scopus as a curated, high-quality bibliometric data source for academic research in quantitative science studies. *Quantitative Science Studies* **1**, 377–386 (2020).

15. Birkle, C., Pendlebury, D. A., Schnell, J. & Adams, J. Web of Science as a data source for research on scientific and scholarly activity. *Quantitative Science Studies* **1**, 363–376 (2020).

16. Szomszor, M. & Adie, E. Overton: A bibliometric database of policy document citations. *Quantitative Science Studies* **3**, 624–650 (2022).

17. Marx, M. & Fuegi, A. Reliance on science by inventors: Hybrid extraction of in-text patent-to-article citations. *Journal of Economics & Management Strategy* **31**, 369–392 (2022).

18. Salganik, M. J. *Bit by Bit: Social Research in the Digital Age*. (Princeton University Press, 2019).

19. Jones, B. F. The burden of knowledge and the 'death of the renaissance man': Is innovation getting harder? *Review of Economic Studies* **76**, 283–317 (2009).

20. Wuchty, S., Jones, B. F. & Uzzi, B. The Increasing Dominance of Teams in Production of Knowledge. *Science* **316**, 1036–1039 (2007).

21. Hill, R. *et al.* Adaptability and the Pivot Penalty in Science and Technology. Preprint at https://doi.org/10.48550/arXiv.2107.06476 (2024).

22. Wang, Y., Qian, Y., Qi, X., Cao, N. & Wang, D. InnovationInsights: A Visual Analytics Approach for Understanding the Dual Frontiers of Science and Technology. *IEEE Transactions on Visualization and Computer Graphics* **30**, 518–528 (2024).

23. Vaccaro, M., Almaatouq, A. & Malone, T. When combinations of humans and AI are useful: A systematic review and meta-analysis. *Nat Hum Behav* **8**, 2293–2303 (2024).

24. Sharma, A., Lin, I. W., Miner, A. S., Atkins, D. C. & Althoff, T. Human–AI collaboration enables more empathic conversations in text-based peer-to-peer mental health support. *Nat Mach Intell* **5**, 46–57 (2023).

25. Can Generative AI improve social science? | PNAS. https://www.pnas.org/doi/abs/10.1073/pnas.2314021121.

26. Brown, T. B. *et al.* Language Models are Few-Shot Learners. Preprint at https://doi.org/10.48550/arXiv.2005.14165 (2020).

27. Wei, J. *et al.* Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. Preprint at https://doi.org/10.48550/arXiv.2201.11903 (2023).

28. Yao, S. *et al.* Tree of Thoughts: Deliberate Problem Solving with Large Language Models. Preprint at https://doi.org/10.48550/arXiv.2305.10601 (2023).

29. Schick, T. *et al.* Toolformer: Language Models Can Teach Themselves to Use Tools. Preprint at https://doi.org/10.48550/arXiv.2302.04761 (2023).

30. Yao, S. *et al.* ReAct: Synergizing Reasoning and Acting in Language Models. Preprint at https://doi.org/10.48550/arXiv.2210.03629 (2023).

31. Wang, Y., Wang, W., Joty, S. & Hoi, S. C. H. CodeT5: Identifier-aware Unified Pre-trained Encoder-Decoder Models for Code Understanding and Generation. Preprint at https://doi.org/10.48550/arXiv.2109.00859 (2021).

32. Fried, D. *et al.* InCoder: A Generative Model for Code Infilling and Synthesis. Preprint at https://doi.org/10.48550/arXiv.2204.05999 (2023).

33. Li, Y. *et al.* Competition-Level Code Generation with AlphaCode. *Science* **378**, 1092–1097 (2022).

34. Chen, M. *et al.* Evaluating Large Language Models Trained on Code. Preprint at https://doi.org/10.48550/arXiv.2107.03374 (2021).

35. Skarlinski, M. D. *et al.* Language agents achieve superhuman synthesis of scientific knowledge. Preprint at https://doi.org/10.48550/arXiv.2409.13740 (2024).

36. Lála, J. *et al.* PaperQA: Retrieval-Augmented Generative Agent for Scientific Research. Preprint at https://doi.org/10.48550/arXiv.2312.07559 (2023).

37. Hu, X. *et al.* InfiAgent-DABench: Evaluating Agents on Data Analysis Tasks. Preprint at https://doi.org/10.48550/arXiv.2401.05507 (2024).

38. Guo, S. *et al.* DS-Agent: Automated Data Science by Empowering Large Language Models with Case-Based Reasoning. Preprint at https://doi.org/10.48550/arXiv.2402.17453 (2024).

39. Hong, S. *et al.* Data Interpreter: An LLM Agent For Data Science. Preprint at https://doi.org/10.48550/arXiv.2402.18679 (2024).

40. Sun, M. *et al.* LAMBDA: A Large Model Based Data Agent. Preprint at https://doi.org/10.48550/arXiv.2407.17535 (2024).

41. Besta, M. *et al.* Graph of Thoughts: Solving Elaborate Problems with Large Language Models. *AAAI* **38**, 17682–17690 (2024).

42. Fan, W. *et al.* A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. Preprint at https://doi.org/10.48550/arXiv.2405.06211 (2024).

43. Gao, Y. *et al.* Retrieval-Augmented Generation for Large Language Models: A Survey. Preprint at https://doi.org/10.48550/arXiv.2312.10997 (2024).

44. Alkaissi, H. & McFarlane, S. I. Artificial Hallucinations in ChatGPT: Implications in Scientific Writing. *Cureus* **15**, e35179.

45. Dahl, M., Magesh, V., Suzgun, M. & Ho, D. E. Large Legal Fictions: Profiling Legal Hallucinations in Large Language Models. *Journal of Legal Analysis* **16**, 64–93 (2024).

46. Evans, O. *et al.* Truthful AI: Developing and governing AI that does not lie. Preprint at https://doi.org/10.48550/arXiv.2110.06674 (2021).

47. Ji, Z. *et al.* Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.* **55**, 1–38 (2023).

48. Ji, Z. *et al.* Towards Mitigating Hallucination in Large Language Models via Self-Reflection. Preprint at https://doi.org/10.48550/arXiv.2310.06271 (2023).

49. Cheng, J. *et al.* Dated Data: Tracing Knowledge Cutoffs in Large Language Models. Preprint at https://doi.org/10.48550/arXiv.2403.12958 (2024).

50. Lewis, P. *et al.* Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Preprint at https://doi.org/10.48550/arXiv.2005.11401 (2021).

51. Karpas, E. *et al.* MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. Preprint at https://doi.org/10.48550/arXiv.2205.00445 (2022).

52. Press, O. *et al.* Measuring and Narrowing the Compositionality Gap in Language Models. Preprint at https://doi.org/10.48550/arXiv.2210.03350 (2023).

53. Asai, A., Wu, Z., Wang, Y., Sil, A. & Hajishirzi, H. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. Preprint at https://doi.org/10.48550/arXiv.2310.11511 (2023).

54. Gao, L., Ma, X., Lin, J. & Callan, J. Precise Zero-Shot Dense Retrieval without Relevance Labels. Preprint at https://doi.org/10.48550/arXiv.2212.10496 (2022).

55. Yu, W. *et al.* Chain-of-Note: Enhancing Robustness in Retrieval-Augmented Language Models. Preprint at https://doi.org/10.48550/arXiv.2311.09210 (2024).

56. Sun, M. *et al.* A Survey on Large Language Model-based Agents for Statistics and Data Science. Preprint at https://doi.org/10.48550/arXiv.2412.14222 (2024).

57. Qiao, B. *et al.* TaskWeaver: A Code-First Agent Framework. Preprint at https://doi.org/10.48550/arXiv.2311.17541 (2024).

58. Zhang, W., Shen, Y., Lu, W. & Zhuang, Y. Data-Copilot: Bridging Billions of Data and Humans with Autonomous Workflow. Preprint at https://doi.org/10.48550/arXiv.2306.07209 (2024).

59. Lu, C. *et al.* The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery. Preprint at https://doi.org/10.48550/arXiv.2408.06292 (2024).

60. Schmidgall, S. *et al.* Agent Laboratory: Using LLM Agents as Research Assistants. Preprint at https://doi.org/10.48550/arXiv.2501.04227 (2025).

61. *Enhancing the Effectiveness of Team Science*. (National Academies Press, Washington, D.C., 2015). doi:10.17226/19007.

62. Barabási, A.-L. Network science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **371**, 20120375 (2013).

63. Zhang, Y., Yuan, Y. & Yao, A. C.-C. Meta Prompting for AI Systems. Preprint at https://doi.org/10.48550/arXiv.2311.11482 (2024).

64. Open Science Collaboration. Estimating the reproducibility of psychological science. *Science* **349**, aac4716 (2015).

65. Nosek, B. A. *et al.* Promoting an open research culture. *Science* **348**, 1422–1425 (2015).

66. Wu, L., Wang, D. & Evans, J. A. Large teams develop and small teams disrupt science and technology. *Nature* **566**, 378–382 (2019).

67. Wan, H., Zhang, Y., Zhang, J. & Tang, J. AMiner: Search and Mining of Academic Social Networks. *Data Intelligence* **1**, 58–76 (2019).

68. Lopez, P. kermitt2/grobid. (2025).

69. Harish. harishsg993010/LLM-Research-Scripts. (2025).

70. OpenAI *et al.* OpenAI o1 System Card. Preprint at https://doi.org/10.48550/arXiv.2412.16720 (2024).

71. DeepSeek-AI *et al.* DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. Preprint at https://doi.org/10.48550/arXiv.2501.12948 (2025).

72. Liu, N. F. *et al.* Lost in the Middle: How Language Models Use Long Contexts. Preprint at https://doi.org/10.48550/arXiv.2307.03172 (2023).

73. Zhao, J. *et al.* LongAgent: Scaling Language Models to 128k Context through Multi-Agent Collaboration. Preprint at https://doi.org/10.48550/arXiv.2402.11550 (2024).

74. Humphrey, W. S. Characterizing the software process: a maturity framework. *IEEE Software* **5**, 73–79 (1988).

75. Carnegie Mellon University, C., Paulk, M. C., Weber, C. V., Curtis, B. & Chrissis, M. B. *The Capability Maturity Model: Guidelines for Improving the Software Process*. (Addison-Wesley Longman Publishing Co., Inc., USA, 1995).

76. Paulk, M. C., Curtis, B., Chrissis, M. B. & Weber, C. V. Capability maturity model, version 1.1. *IEEE Software* **10**, 18–27 (1993).

77. Breznau, N. *et al.* Observing many researchers using the same data and hypothesis reveals a hidden universe of uncertainty. *Proceedings of the National Academy of Sciences* **119**, e2203150119 (2022).

78. Otis, N. G., Cranney, K., Delecourt, S. & Koning, R. Global Evidence on Gender Gaps and Generative AI. Preprint at https://doi.org/10.31219/osf.io/h6a7c (2024).

79. Gao, J. & Wang, D. Quantifying the use and potential benefits of artificial intelligence in scientific research. *Nat Hum Behav* **8**, 2281–2292 (2024).