

34 Examples of LLM Applications in Materials Science and Chemistry: Towards Automation, Assistants, Agents, and Accelerated Scientific Discovery

Yoel Zimmermann¹, Adib Bazgir², Alexander Al-Feghali³, Mehrad Ansari⁴, Joshua Bocarsly²⁷, L. Catherine Brinson⁵, Yuan Chiang^{6,7}, Defne Ciri⁵, Min-Hsueh Chiu⁸, Nathan Daelman⁹, Matthew L. Evans^{10,11}, Abhijeet S. Gangar²⁶, Janine George^{12,13}, Hassan Harb¹⁴, Ghazal Khalighinejad⁵, Sartaaj Takrim Khan¹⁵, Sascha Klawohn⁹, Magdalena Lederbauer^{1,20}, Soroush Mahjoub¹⁶, Bernadette Mohr^{9,17}, Seyed Mohamad Moosavi^{4,15}, Aakash Naik^{12,13}, Aleyna Beste Ozhan¹⁶, Dieter Plessers¹⁸, Aritra Roy¹⁹, Fabian Schöppach⁹, Philippe Schwaller²⁰, Carla Terboven²¹, Katharina Ueltzen^{12,13}, Yue Wu²⁸, Shang Zhu²², Jan Janssen²³, Calvin Li²⁴, Ian Foster^{14,25}, and Ben Blaiszik*^{14,25}

¹ETH Zurich

²University of Missouri-Columbia

³McGill University

⁴Acceleration Consortium

⁵Duke University

⁶University of California at Berkeley

⁷Lawrence Berkeley National Laboratory

⁸University of Southern California

⁹Humboldt University of Berlin

¹⁰Université catholique de Louvain

¹¹Matgenix SRL

¹²Friedrich-Schiller-Universität at Jena

¹³Federal Institute of Materials Research and Testing (BAM)

¹⁴Argonne National Laboratory

¹⁵University of Toronto

¹⁶Massachusetts Institute of Technology

¹⁷University of Amsterdam

¹⁸KU Leuven

¹⁹London South Bank University

²⁰EPFL

²¹Helmholtz-Zentrum Berlin für Materialien und Energie GmbH

²²University of Michigan-Ann Arbor

²³Max-Planck Institute for Sustainable Materials

²⁴Fum Technologies, Inc.

²⁵University of Chicago

²⁶University of California, Los Angeles

²⁷University of Houston

²⁸Independent Researcher

*Corresponding author: blaiszik@uchicago.edu

Abstract

Large Language Models (LLMs) are reshaping many aspects of materials science and chemistry research, enabling advances in molecular property prediction, materials design, scientific automation, knowledge extraction, and more. Recent developments demonstrate that the latest class of models are able to integrate structured and unstructured data, assist in hypothesis generation, and streamline research workflows. To explore the frontier of LLM capabilities across the research lifecycle, we review applications of LLMs through 34 total projects developed during the second annual Large Language Model Hackathon for Applications in Materials Science and Chemistry, a global hybrid event. These projects spanned seven key research areas: (1) molecular and material property prediction, (2) molecular and material design, (3) automation and novel interfaces, (4) scientific communication and education, (5) research data management and automation, (6) hypothesis generation and evaluation, and (7) knowledge extraction and reasoning from the scientific literature. Collectively, these applications illustrate how LLMs serve as versatile predictive models, platforms for rapid prototyping of domain-specific tools, and much more. In particular, improvements in both open source and proprietary LLM performance through the addition of reasoning, additional training data, and new techniques have expanded effectiveness, particularly in low-data environments and interdisciplinary research. As LLMs continue to improve, their integration into scientific workflows presents both new opportunities and new challenges, requiring ongoing exploration, continued refinement, and further research to address reliability, interpretability, and reproducibility.

Introduction

The integration of large language models (LLMs) into scientific workflows is reshaping how researchers approach data-driven discovery, automation, and even scientific reasoning and hypothesis generation [1, 2, 3, 4]. In chemistry and materials science, fields characterized by complex data modalities, heterogeneous data formats, sparse experimental datasets, and fragmented knowledge ecosystems, LLMs are emerging as versatile tools capable of bridging gaps between computational methods, experimental data, literature and text sources, and domain expertise [5, 6, 7, 8, 9, 10, 11, 12]. Early applications have already demonstrated potential applicability in tasks ranging from molecular property prediction [13, 14, 15] to automated laboratory workflows [16, 17] and development of novel user interfaces [18, 19]. As illustrated in Figure 1, we note that there is a significant opportunity for these broad new capabilities to be incorporated throughout the scientific research lifecycle; from initial ideation through experimental execution to communication, learning, and further iteration.

However, the rapidity of change and the nearly constant release of models with higher performance, lower cost, and wider application spaces, and release of other platform capabilities (e.g., agentic tools, deep research modalities) make it challenging to keep pace, necessitating a collaborative and interdisciplinary effort to identify high-impact use cases, address specific limitations, and prototype applications to catalyze deeper study [20, 21, 22, 23, 24, 25, 26]. Towards this goal, we believe that accessing the wisdom of the crowd via science hackathons provides a powerful, and dynamic framework for fostering collaboration building, knowledge exchange, innovation, and incentivizing the rapid problem-solving and exploration needed to realize the benefit of these new models for scientific discovery in materials science and chemistry [27, 28, 29, 30].

In this work, we describe and analyze select applications developed as part of the second Large Language Model Hackathon for Applications in Materials Science and Chemistry [30], detailing the broad classes of problems addressed by teams and highlighting trends in the approaches taken. We categorize the 34 submissions into seven key research areas and provide an overview of team contributions with highlights drawn from exemplar projects in each research area. We also present a summary table containing team details and code repository links for all submissions to offer a comprehensive view of the innovations demonstrated during the event.

Finally, we discuss the broader conclusions of the hackathon, emphasizing its role in fostering interdisciplinary collaboration, accelerating the adoption of artificial intelligence (AI) in scientific research [27, 28, 29], and identifying key challenges that require further investigation. By examining these contributions, we provide insight into how structured collaborative frameworks can drive the systematic integration of LLMs into

chemistry and materials science to accelerate research, improve researcher efficiency, and shape the future of AI-driven discovery.

Overview of Submissions

The hackathon resulted in 34 team submissions (with 32 submissions providing detailed descriptions), covering a broad spectrum of materials science and chemistry applications. The submissions and links to the respective source code repositories are listed in [Table 1](#). We categorized projects based on their primary objectives, clustering them into seven key areas, forming a constellation of new capabilities across the research lifecycle:

1. **Molecular and Material Property Prediction**: Forecasting chemical and physical properties of molecules and materials using LLMs, particularly excelling in low-data environments and combining structured and unstructured data.
2. **Molecular and Material Design**: Generating and optimizing novel molecules and materials using LLMs, including peptides, metal-organic frameworks, and sustainable construction materials.
3. **Automation and Novel Interfaces**: Developing natural language interfaces and LLM-powered automated workflows to simplify complex scientific tasks, making advanced tools and techniques more accessible to researchers.
4. **Scientific Communication and Education**: Enhancing academic communication, automating educational content creation, and supporting learning in materials science and chemistry.
5. **Research Data Management and Automation**: Streamlining the handling, organization, and processing of scientific data through LLM-powered tools and multimodal agents.
6. **Hypothesis Generation and Evaluation**: Using LLMs to generate, assess, and refine scientific hypotheses, leveraging multiple AI agents and statistical approaches.
7. **Knowledge Extraction and Reasoning**: Extracting structured information from scientific literature and performing sophisticated reasoning about chemical and materials science concepts through knowledge graphs and multimodal approaches.

Collectively, this constellation of capabilities, shown in [Figure 1](#), is applicable to long-standing challenges across the research lifecycle, creating a flywheel of improvements that promises to empower researchers with new capabilities and to speed the research process.

Table 1: Overview of the tools developed by the various tools, and links to source code repositories. Full descriptions of the projects can be found in Ref. [31].

Project	Authors	Links
Molecular and Material Property Prediction		
Leveraging Orbital-Based Bonding Analysis Information in LLMs	Katharina Ueltzen, Aakash Naik, Janine George	GitHub
Context-Enhanced Material Property Prediction (CEMPP)	Federico Ottomano, Elena Patyukova, Judith Clymo, Dmytro Antypov, Chi Zhang, Aritra Roy, Piyush Ranjan Maharana, Weijie Zhang, Xuefeng Liu, Erik Bitzek	GitHub

Project	Authors	Links
MolFoundation: Benchmarking Chemistry LLMs on Predictive Tasks	Hassan Harb, Xuefeng Liu, Anastasiia Tsymbal, Oleksandr Narykov, Dana O'Connor, Shagun Maheshwari, Stanley Lo, Archit Vasan, Zartashia Afzal, Kevin Shen	GitHub
3D Molecular Feature Vectors for Large Language Models	Jan Weinreich, Ankur K. Gupta, Amirhossein D. Naghdi, Alishba Imran	GitHub
LLMSpectrometry	Tyler Josephson, Fariha Agbere, Kevin Ishimwe, Colin Jones, Charishma Puli, Samiha Sharlin, Hao Liu	GitHub
Molecular and Material Design		
MC-Peptide: An Agentic Workflow for Data-Driven Design of Macro cyclic Peptides	Andres M. Bran, Anna Borisova, Marcel M. Calderon, Mark Tropin, Rob Mills, Philippe Schwaller	GitHub
Leveraging AI Agents for Designing Low Band Gap Metal-Organic Frameworks	Mehrad Ansari, Sartaaj Takrim Khan, Mahyar Rajabi, Seyed Mohamad Moosavi, Amro Aswad	GitHub
How Low Can You Go? Leveraging Small LLMs for Material Design	Alessandro Canalicchio, Alexander Moßhammer, Tehseen Rug, Christoph Völker	GitHub
Automation and Novel Interfaces		
LangSim	Yuan Chiang, Giuseppe Fisicaro, Greg Juhasz, Sarom Leang, Bernadette Mohr, Utkarsh Pratiush, Francesco Ricci, Leopold Talirz, Pablo A. Unzueta, Trung Vo, Gabriel Vogel, Sebastian Pagel, Jan Janssen	GitHub
LLMicroscopilot: assisting microscope operations through LLMs	Marcel Schloz, Jose C. Gonzalez	GitHub
T2DlIlama: Harnessing Language Model for Density Functional Theory (DFT) Parameter Suggestion	Chiku Parida, Martin H. Petersen	GitHub
Materials Agent: An LLM-Based Agent with Tool-Calling Capabilities for Cheminformatics	Archit Datar, Kedar Dabhadkar	GitHub
LLM with Molecular Augmented Token	Luis Pinto, Xuan Vu Nguyen, Tirtha Vinchurkar, Pradip Si, Suneel Kuman	GitHub
Scientific Communication and Education		
MaSTeA: Materials Science Teaching Assistant	Defne Ciri, Abhijeet S. Gangan, Mohd Zaki	GitHub

Project	Authors	Links
LLMy-Way	Ruijie Zhu, Faradawn Yang, Andrew Qin, Suraj Sudhakar, Jaehee Park, Victor Chen	GitHub
WaterLLM: Creating a Custom ChatGPT for Water Purification Using PromptEngineering Techniques	Viktoria Baibakova, Maryam G. Fard, Teslim Olaiwola, Olga Taran	GitHub
Research Data Management and Automation		
yeLLowhaMMer: A Multi-modal Tool-calling Agent for Accelerated Research Data Management	Matthew L. Evans, Benjamin Charmes, Vraj Patel, Joshua D. Bocarsly	GitHub
LLMads	Sarthak Kapoor, Jos'e M. Pizarro, Ahmed Ilyas, Alvin N. Ladines, Vikrant Chaudhary	GitHub
NOMAD Query Reporter: Automating Research Data Narratives	Nathan Daelman, Fabian Schöppach, Carla Terboven, Sascha Klawohn, Bernadette Mohr	GitHub
Speech-schema-filling:Creating Structured Data Directly from Speech	Hampus N"asstr"om, Julia Schumann, Michael G"otte, Jos'e A. M"rquez	GitHub
Hypothesis Generation and Evaluation		
Leveraging LLMs for Bayesian Temporal Evaluation of Scientific Hypotheses	Marcus Schwarting	GitHub
Multi-Agent Hypothesis Generation and Verification through Tree of Thoughts and Retrieval Augmented Generation	Aleyna Beste Ozhan, Soroush Mahjoubi	GitHub
ActiveScience	Min-Hsueh Chiu	GitHub
G-Peer-T: LLM Probabilities For Assessing Scientific Novelty and Nonsense	Alexander Al-Feghali, Sylvester Zhang	GitHub
Knowledge Extraction and Reasoning		
ChemQA	Ghazal Khalighinejad, Shang Zhu, Xuefeng Liu	GitHub
LithiumMind - Leveraging Language Models for Understanding Battery Performance	Xinyi Ni, Zizhang Chen, Rongda Kang, Sheng-Lun Liao, Pengyu Hong, Sandeep Madireddy	GitHub
KnowMat: Transforming Unstructured Material Science Literature into Structured Knowledge	Hasan M. Sayeed, Ramsey Issa, Trupti Mohanty, Taylor Sparks	GitHub
Ontosynthesis	Qianxiang Ai, Jiaru Bai, Kevin Shen, Jennifer D'Souza, Elliot Risch	GitHub
Knowledge Graph RAG for Polymer Simulation	Jiale Shi, Weijie Zhang, Dandan Tang, Chi Zhang	GitHub
Synthetic Data Generation and Insightful Machine Learning for High Entropy Alloy Hydrides	Tapashree Pradhan, Devi Dutta Biswajeet	GitHub

Project	Authors	Links
Chemsense: Are large language models aligned with human chemical preference?	Martíno Ríos-García, Nawaf Alampara, Mara Schilling-Wilhelmi, Abdelrahman Ibrahim, Kevin Maik Jablonka	GitHub
GlossaGen	Magdalena Lederbauer, Dieter Plessers, Philippe Schwaller	GitHub

We next discuss the constellation of capabilities in more detail and highlight exemplar projects across each key application area.

1 Molecular and Material Property Prediction

LLMs have rapidly advanced in molecular and material property prediction, employing both textual and numerical data to forecast a wide range of properties. Recent studies [1, 15, 13, 3] show LLMs performing comparably to, or even surpassing, conventional machine learning methods, particularly in low-data environments. The flexibility in processing both structured and unstructured data [32], as well as their general applicability to regression tasks [33], make LLMs a powerful tool for diverse predictive tasks in molecular and materials science.

1.1 Leveraging orbital-based bonding analysis information in LLMs for material property predictions

Previous studies have used different strategies to learn material properties using LLMs, such as enriching graph neural network (GNN) features with LLM embeddings [4], training domain-specific LLMs and customizing model architectures [5, 6, 7], or fine-tuning general-purpose LLMs [8, 14]. While exact strategies have differed, existing models predominantly operate on string representations of crystal structures primarily consisting of compositional and structural information commonly found in crystallographic information files (CIFs). Multiple studies have successfully utilized the text descriptions of structures [7, 14, 8] that can be generated using the Robocrystallographer package [9]. These descriptions consist of structural features like bond lengths, coordination polyhedra, lattice parameters, coordinates, structure type, and other descriptors. Other studies explored different string representations of compositional and structural information [4, 6, 14].

The team behind this submission emphasizes that, to their knowledge, no previous studies investigated including orbital-based bonding analysis information in LLMs for materials property prediction tasks. Thus, in this pilot study, the team tested including such descriptions in LLMs to predict the highest-frequency peak in their phonon density of states (DOS) [11, 10]. This target is relevant to the thermal properties of materials and it is a tracked component of the MatBench benchmark project. A key hypothesis is that the inclusion of the bonding analysis information for this vibrational property will improve the LLM's performance, as previous studies demonstrated the importance of such bonding information for the same target via a Random Forest model [34].

To test this hypothesis, the team fine-tuned multiple Llama 3 models on the textual description of 1264 crystal structures in the benchmark dataset. The text descriptions were generated using two packages: the Robocrystallographer and LobsterPy package [35]. The text descriptions from LobsterPy consist of orbital-based bonding analyses containing information on covalent bond strengths and antibonding states. The data used here is available on Zenodo [36] and was generated as part of an earlier dataset publication [34].

During the hackathon, one Llama model was fine-tuned with the Alpaca prompt format using both Robocrystallographer and LobsterPy text descriptions, and another one using solely Robocrystallographer input. Figure 2 depicts the prompt used to fine-tune an LLM to predict the last phonon DOS peak. The train/test/validation split was 0.64/0.2/0.16. The models were trained for 10 epochs with a validation step after each epoch. The textual output was converted back into numerical frequency values for the computation of MAEs and RMSEs. The results show that including bonding-based information improved the model's

The LLM-Powered Research Constellation

7 key application areas that integrate across the research lifecycle

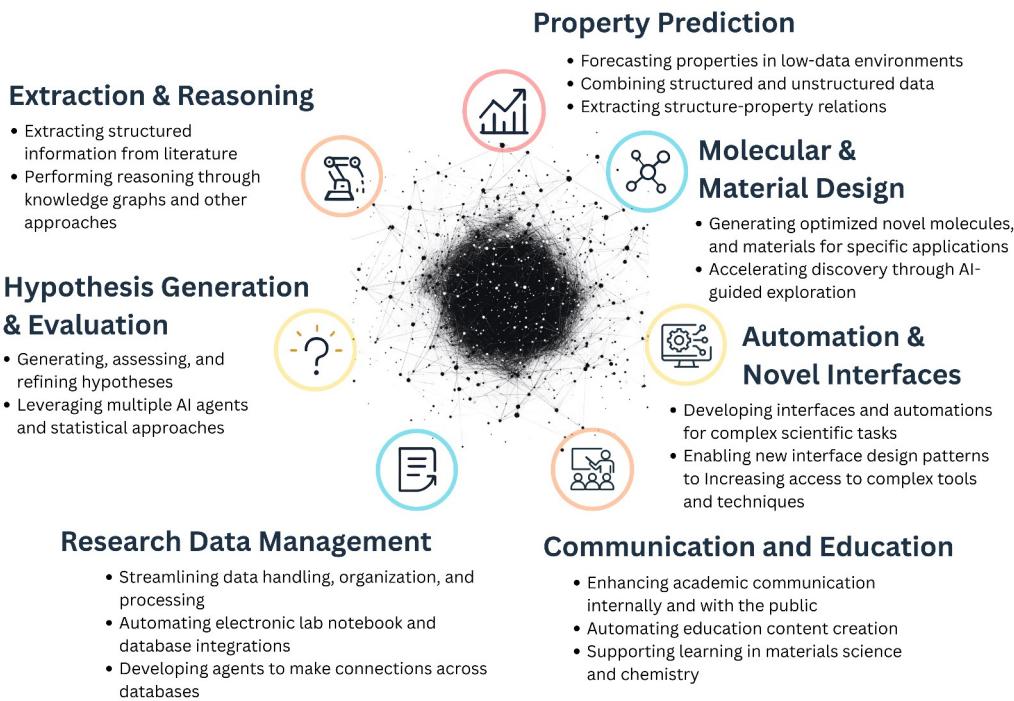


Figure 1: The LLM-Powered Research Constellation. At each stage of the research process, from initial ideation through experimental execution and communication of results, LLMs provide a constellation of capabilities spanning hypothesis generation, property prediction, novel interfaces, education, material design, automation, data management, scientific communication, and more. This constellation demonstrates the possibility of LLMs and multimodal models to drive a more efficient, rapid, and creative scientific discovery process through integrations across the research lifecycle.

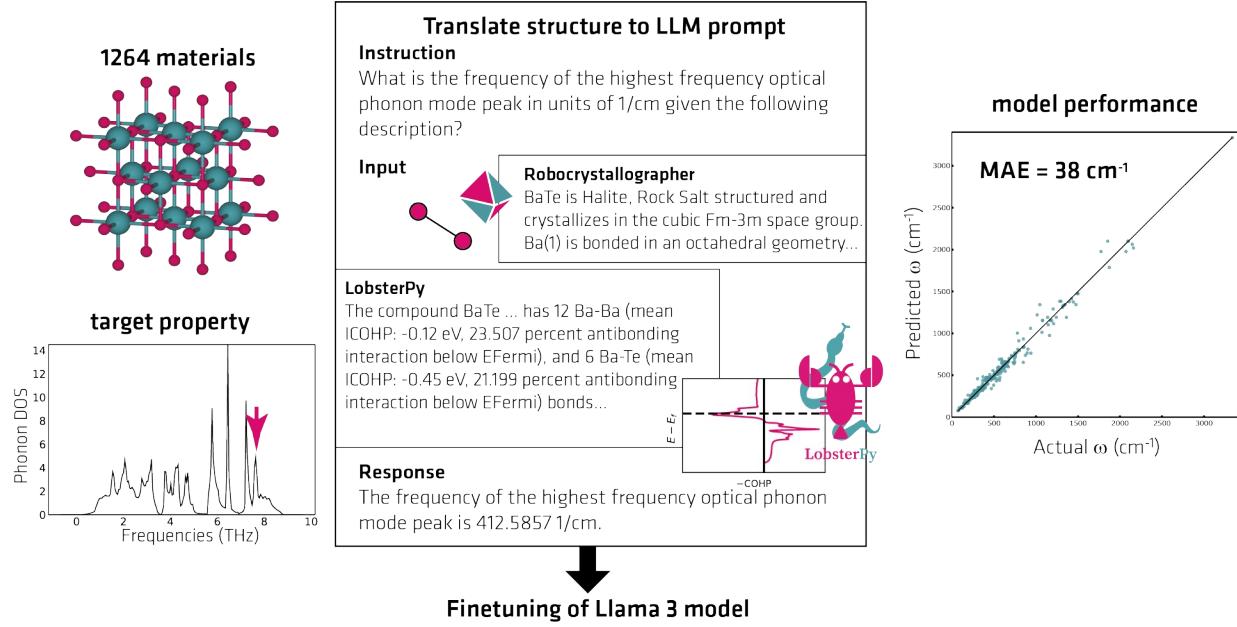


Figure 2: Schematic depicting the prompt for fine-tuning the LLM with Alpaca prompt format.

prediction. The results also corroborate the team’s previous finding that quantum-chemical bond strengths are relevant for this particular target property. Both model performances (Robocrystallographer: 44 cm^{-1} , Robocrystallographer+LobsterPy: 38 cm^{-1}) are comparable to other models of the MatBench test suite, with MAEs ranging from 29 cm^{-1} to 68 cm^{-1} as per the time of writing [37].

Although the preliminary results seem promising, the models have not yet been exhaustively analyzed, validated, or optimized yet. As the prediction of a numerical value and not its text embedding is of interest to the task, further model adaptation might be beneficial. For example, Rubungo et al. [7] modified T5, an encoder-decoder model, for regression tasks by removing its decoder and adding a linear layer on top of its encoder. Halving the number of model parameters allowed them to fine-tune on longer input sequences, improving model performance. A recently published benchmark for LLMs in materials property prediction also suggests that fine-tuning models with fewer parameters improves the prediction of materials properties [14].

With the available easy-to-use packages like Unslloth, [38] the team was able to integrate their materials data into fine-tuning an LLM for property prediction with very limited resources and time. Since these initial results, the work has been extended to a dataset of bonding-based text descriptions including 13,000 crystalline materials. In the future, the team aims to (1) test these text descriptions further to learn other thermal and elastic material properties like elastic constants and lattice thermal conductivity and (2) to extend further the text descriptions generated with the LobsterPy package to include, e.g., information on computed charges.

2 Molecular and Material Design

LLMs have also been applied to molecular and material design, proving capable in both settings [2, 39, 40, 41, 42], especially if pre-trained or fine-tuned with domain-specific data [43]. However, despite these advancements, LLMs still face limitations in practical applications [44].

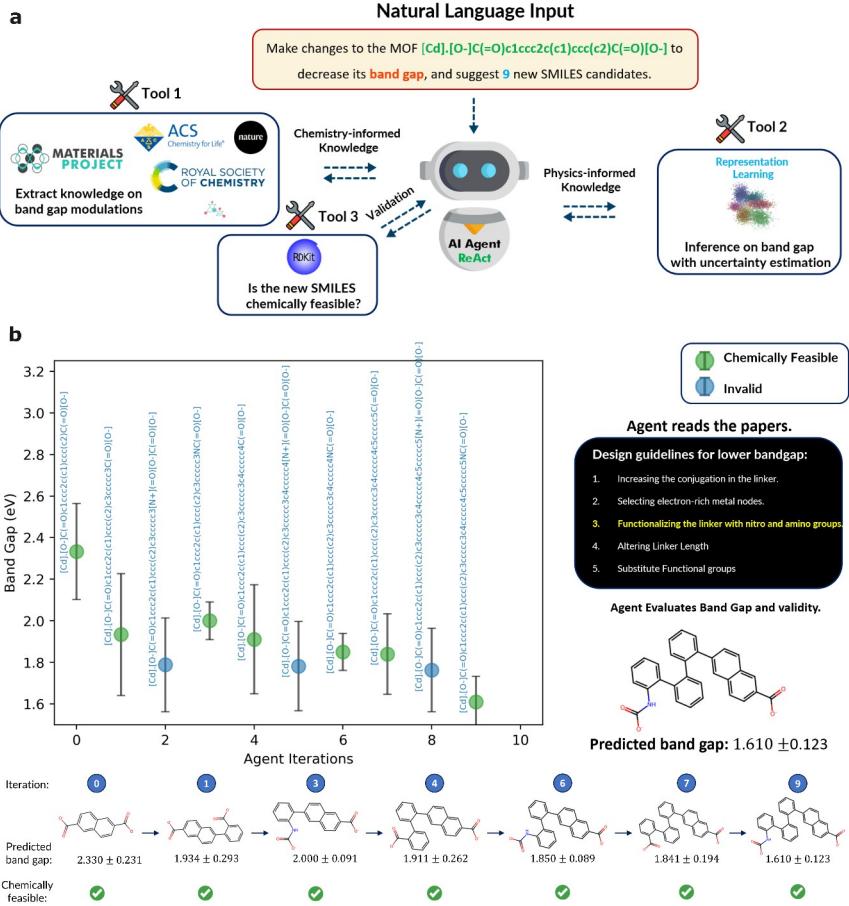


Figure 3: Workflow overview. The ReAct agent looks up guidelines for designing low band gap MOFs from research papers and suggests a new MOF (likely with a lower band gap). It then checks the validity of the new SMILES candidate and predicts the band gap with epistemic uncertainty estimation using an ensemble of surrogate fine-tuned MOFormers.b. Band gap predictions for new MOF candidates as a function of agent iterations

2.1 Leveraging AI Agents for Designing Low Band Gap Metal-Organic Frameworks

Metal-organic frameworks (MOFs) are known to be excellent candidates for electrocatalysis due to their large surface area, high adsorption capacity at low CO_2 concentrations, and the ability to fine-tune the spatial arrangement of active sites within their crystalline structure [45]. Low band gap MOFs are crucial as they efficiently absorb visible light and exhibit higher electrical conductivity, making them suitable for photocatalysis, solar energy conversion, sensors, and optoelectronics. This submission aims at using chemistry-informed ReAct [46] AI Agents to optimize the band gap property of MOFs. The overview of the workflow is presented 3a. The agent takes as inputs a textual representation of the initial MOF structure as a SMILES (Simplified Molecular Input Line-Entry System) string representation, and a short description of the property optimization task (i.e., reducing band gap), all in natural language. This is followed by an iterative closed-loop suggestion of new MOF candidates with a lower band gap with uncertainty quantification, by adjusting the initial MOF given a set of design guidelines automatically obtained from the scientific literature. A detailed analysis of this methodology, including its application to various classes of materials such as surfactants, ligands, and peptides can be found in reference [47], which supports both closed-loop and human-in-the-loop feedback cycles and thus enables real-time property inference for human-AI collaboration in molecular design.

The agent, powered by an LLM, is augmented with a set of tools allowing for chemistry-informed decision-making. These tools are as follows:

1. **Retrieval-Augmented Generation (RAG)**: This tool allows the agent to obtain design guidelines on how to adapt the MOF structure from unstructured text. Specifically, in this prototype, the agent has access to a fixed set of 7 MOF research papers (see Refs. [48, 49, 50, 51, 52, 53, 54]) as PDFs. This tool is designed to extract the most relevant sentences from papers in response to a given query. It works by embedding both the paper and the query into numerical vectors using OpenAI's text-ada-002 [55], then identifying the top k passages within the document that either explicitly mention or implicitly suggest the adaptations required for the specified band gap property for a MOF. Inspired by the team's earlier work [56], k is set to 9, but is dynamically adjusted based on the relevant context's length to avoid OpenAI's token limitation.
2. **Surrogate Band Gap Predictor** The surrogate model used is a transformer (MOFormer [57]) that takes as input the MOF as a SMILES string. This model is pre-trained using a self-supervised learning technique known as Barlow-Twin [58], where representation learning is done against structure-based embeddings from a crystal graph convolutional neural network (CGCNN) [59]. This was done against 16,000 BW20K entries [60]. The pre-trained weights are then transferred and fine-tuned to predict the band gap labels taken from 7,450 entries from the QMOF database [61]. From a 5-fold training, an ensemble of five transformers are trained to return the mean band gap and the standard deviation, which is used to assess uncertainty for predictions. For comparison, the team's transformer's mean absolute error (MAE) is approximately 0.467, whereas MOFormer, which was pre-trained on 400,000 entries, achieves an MAE of approximately 0.387.
3. **ChemicalFeasibility Evaluator** This tool primarily uses RDKit [62] to convert a SMILES string into an RDKit *Mol* object, and performs several validation steps to ensure chemical feasibility. First, it parses the SMILES string to confirm correct syntax. Next, it validates the atoms and bonds, ensuring they are chemically valid and recognized. It then checks atomic valences to ensure each atom forms a reasonable number of bonds. For ring structures, RDKit verifies the correct ring closure notation. Additionally, it adds implicit hydrogens to satisfy valence requirements and detects aromatic systems, marking relevant atoms and bonds as aromatic. These steps collectively ensure the molecule's basic chemical validity.

The team has used OpenAI's GPT-4 [63] with a temperature of 0.1 as the preferred LLM and LangChain [64] for the application framework development (nonetheless, the team confirms that the choice of LLM is only a hyperparameter and other LLMs can drive the agent).

The new MOF candidates and their corresponding inferred band gap are represented in Figure 3b. The agent starts by retrieving the following design guidelines for low band gap MOFs from research papers: 1. Increasing the conjugation in the linker. 2. Selecting electron-rich metal nodes. 3. Functionalizing the linker with nitro and amino groups. 4. Altering linker length. 5. Substitute functional groups (i.e., substituting hydrogen with electron-donating groups on the organic linker). Note that the metal node adaptations are restrained by simply changing the system input prompt. The agent iteratively implements the above strategies, makes changes to the initial MOF, and suggests a new SMILES. The new SMILES is validated using the Chemical Feasibility Evaluator tool, and if found invalid, the agent uses a self-correction feedback loop to suggest new candidates, accounting for the extracted design guideline. After each valid modification, the band gap of the new MOF is then assessed using the fine-tuned ensemble of surrogate MOFormers to ensure a lower band gap. The self-correction feedback loop also handles new MOFs with undesired higher band gaps with respect to the initial MOF, by reverting to the most recent valid MOF candidate with the lowest band gap identified throughout the iterations.

3 Automation and Novel Interfaces

LLMs are increasingly important to the modern scientific workflow, enabling the development of more intuitive interfaces for users dealing with complex digital tools. For example, platforms such as ChemCrow [19], RestGPT [65], and HoneyComb [18] allow researchers to input commands in natural language to interact

with and analyze complex software and databases. With LLMs, democratized access and dramatically simpler interfaces are possible for programs like specialized computational techniques or command-line interfaces that may previously have required deep expertise. LLMs excel at autonomous planning and task execution in multistep scenarios [16] by breaking complex processes into smaller actions, making experimental or computational workflows controllable by models with less need for direct oversight. Such behavior may include but is not limited to: simple interaction with laboratory robotic systems [66, 17], where difficult scientific objectives can be converted into precise, callable commands the basis of precision and consistency. The integration of LLMs and robotics promises to improve operational efficiency and enable new designs of experimental workflows with increased flexibility.

3.1 LangSim – Large Language Model Interface for Atomistic Simulation

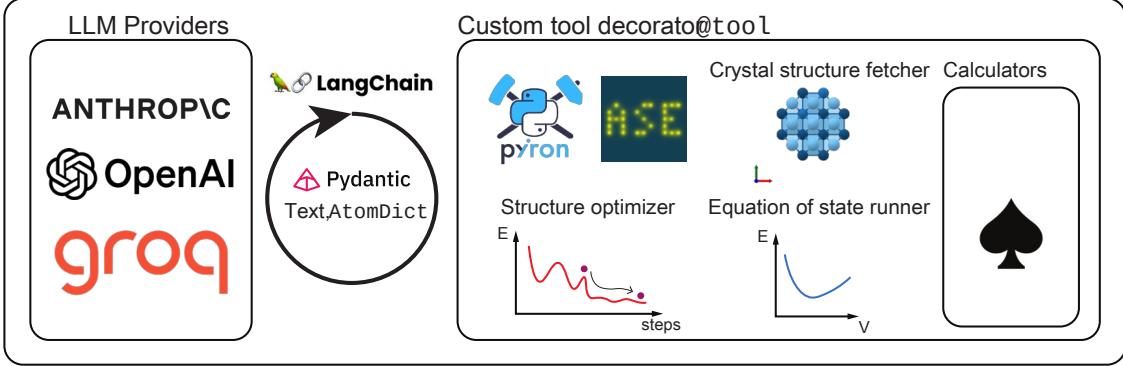
LLMs can augment scientists with their common workflows, dramatically simplifying the interactions across systems using natural language input to understand and implement the intent of the user. The LangSim project [67] prototyped an interface to showcase the ability of LLMs to autonomously start atomistic simulations to study material properties on an atomistic scale. This provides the LLM with a way to request and then use novel scientific data and insights that were previously not available in published databases one might imagine, e.g., the on-the-fly calculation of defect properties, e.g., grain boundary segregation energies. In addition, by integrating the LLM in the active learning cycle of an autonomous materials discovery loop, with the option to calculate different material properties and access existing databases the LLM becomes an AI scientist on a quest to discover novel materials. In this project, straightforward atomistic simulation and agentic scientific reasoning were explored as a natural language interface to users without programming skills.

The LangSim project implements atomistic simulation agents based on both pyiron [68] and LangChain [69]. LangChain enables the LLM to call any kind of Python function and include the output in the thought process of the next iteration. In the case of LangSim, these Python functions represent simulation workflows implemented in the pyiron [68] workflow framework to calculate material properties with atomistic simulations. By restricting the LLM to pre-defined simulation workflows, the risk of hallucination is reduced compared to generative approaches, which request the LLM to define and generate the simulation workflow. Based on the MACE [70] foundation model for atomistic simulation, LangSim was used to predict the binary concentration of solid solution alloy required to match a user-defined bulk modulus, demonstrating an inverse materials design approach to enable application-specific alloy design.

3.2 LLMicroscopilotassisting microscope operations through LLMs

While the state-of-the-art microscopes in materials science are crucial for high-resolution imaging and analysis, they are still rather addressed by expert operators due to their complex and steep-cost ownership. Their manipulation involves delicate tasks, mostly involving precision alignment, guaranteed optimal performances, and shifting between different operational modes to address various research questions that require extensive training and experience. This unobtainable quality has not only slowed down routine experimental procedures but has also formed a serious roadblock to opportunities for broadening access and allowing an acceleration of scientific discovery. With progress in natural language processing, LLMs opened the way for a Copernican revolution in this landscape. Integration of LLMs to the microscope interface will allow complex operations to be done through natural language commands. Similar to modern chatbots, which allow even those with no programming knowledge to generate complex computer programs [71], LLMs can become intuitive intermediaries assisting users in traversing the manifold control procedures of advanced microscopes. Early studies of scanning probe microscopy have shown that LLMs can facilitate remote access [72] and even direct control [73] of these instruments, lessening the workload for expert operators. A promising approach is to use an LLM agent that accesses and operates some concrete external tools. These agents also interpret user commands and use observations in realtime to make decisions, reducing the hallucinations, or wrong outputs, that sometimes appear with a standalone LLM. This would streamline the user experience, further relieving researchers from having to navigate through complex, tool-specific APIs, thus broadening the reach of advanced microscopes, especially to non-experts.

LangSim Atomistic Simulation Agents



Use case: Inverse alloy design

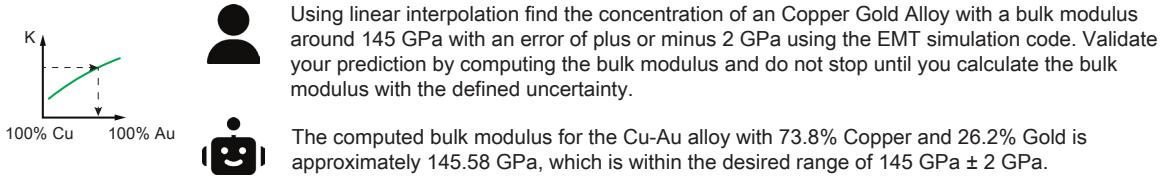


Figure 4: LangSim framework for atomistic simulation and inverse design. Custom atomistic modeling tools (such as pyiron, ASE Python package functions with underlying EMT and MACE-MP-0 forcefields) are integrated using LangChain @ tooldecorator. Pydantic model is used to exchange atomic information in a structured format between LLM and tools. The emerging agentic capability for inverse alloy design is demonstrated. LLM agent is able to find the target composition of Cu-Au alloy with the desired bulk modulus.

An illustration of such is the work performed by the LLMicroscopilot team, an LLM-based agent partially automating the operation of a scanning transmission electron microscope. LLMicroscopilot, its prototype, combines a generally trained foundation model, which is then tailored to specific people and domains through dedicated control tools. This agent operates quite well at first, utilizing the API for a microscope experiment simulation tool [74], by performing such important tasks as estimating experimental parameters and executing the actual experiments. Therefore, this automation reduced dependence on personnel highly trained in operating such systems, thus increasing the opportunities for wider engagement in materials science due to the impact on usability. In the future, though, developments in the field are expected with LLMicroscopilot. In the future, they would involve integrating open-source microscope hardware control tools [75] and include capabilities for database access. Consequently, the system will be able to utilize Retrieval-Augmented Generation techniques to further inform parameter estimation and aid in the data analysis. Effectively, this will allow researchers to integrate LLMs in user interfaces at high-end microscopes and, instead of working on tedious, routine operational tasks, invest their energy in high-level scientific research and innovation, democratizing access to advanced experimental techniques.

4 Scientific Communication and Education

LLMs are transforming how scientific and educational content is created and shared, enhancing accessibility and personalized learning [76, 77, 78, 79]. By automating tasks like question generation, feedback, and grading, LLMs streamline educational processes, freeing educators to focus on individual learning needs. Additionally, LLMs assist in translating complex scientific findings into accessible formats, broadening public engagement [79]. However, technological readiness, transparency, and ethical concerns around data privacy and bias remain critical challenges to address [78, 76].

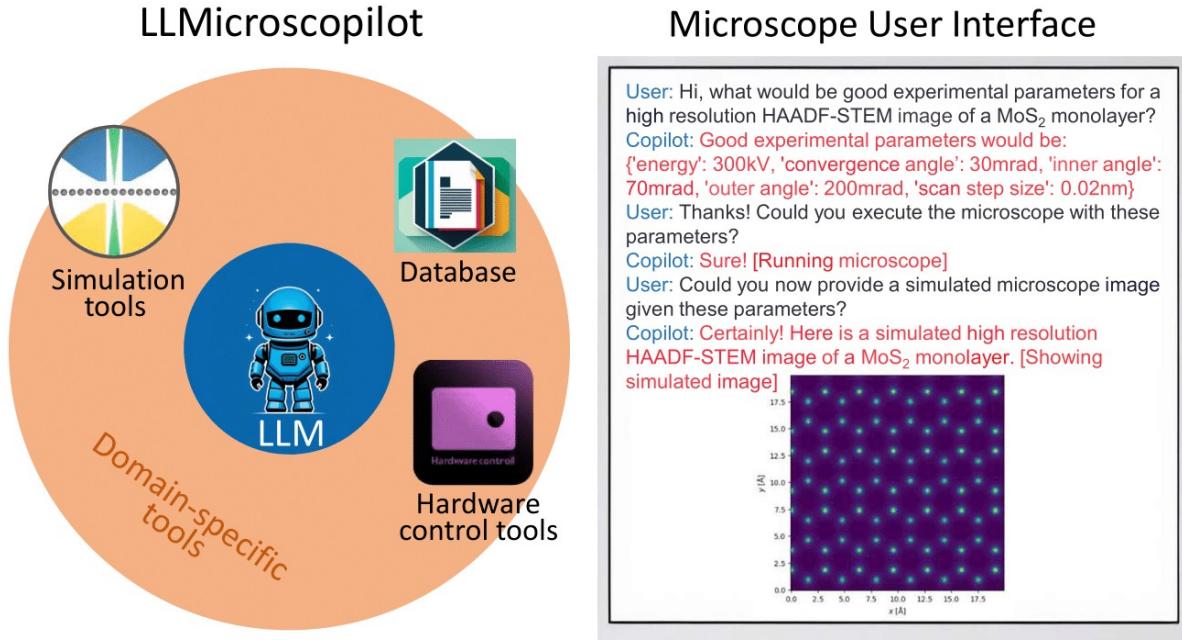


Figure 5: Schematic overview of the LLMicroscopilot assistant. The microscope user interface allows the user to input queries, which are then processed by the LLM. The LLM executes appropriate tools to provide domain-specific knowledge, support data analysis, or operate the microscope.

4.1 MaSTeA: Materials Science Teaching Assistant

This team selected 650 questions from the materials science question answering dataset (MaScQA) [80], requiring undergraduate-level understanding to solve. These questions are classified into four types based on their structure: Multiple Choice Questions (MCQs), Match the Following (MATCH), Numerical Questions with Given Options (MCQN), and Numerical Questions (NUM). MCQs are generally conceptual, with four options, where mostly one is correct, though occasionally multiple answers are valid. MATCH questions involve two lists of entities that need to be correctly paired, with four answer choices provided, one of which contains the correct set of matches. MCQN questions present a numerical problem with four answer choices, requiring a solution to identify the correct option, while NUM questions have numerical answers rounded to the nearest integer or floating-point number as specified. The team aimed to automate the evaluation of open-source and proprietary LLMs on MaScQA and develop an interactive interface for students to engage with these questions. Various models, including LLAMA3-8B, HAIKU, SONNET, GPT-4, and OPUS, were evaluated across 14 subject categories such as characterization, applications, properties, and behavior. The evaluation results, summarized in Table 2, show that the OPUS variant of Claude consistently outperformed other models, achieving the highest accuracy in most categories. GPT-4 also demonstrated strong performance, particularly in material processing and fluid mechanics. As expected from prior studies, larger models such as OPUS and GPT-4 outperformed the smaller LLAMA3-8B, reinforcing the significance of model size in performance [81]. The results suggest that there is significant room for improvement to enhance the accuracy of language models in answering scientific questions.

The evaluation involved:

- **Extracting corresponding values:** For MCQs, correct choices were identified using regular expressions and compared to model predictions.
- **Prediction verification:** Numerical predictions were validated against exact or acceptable ranges, while MCQ responses were matched to correct answer choices.

- **Calculating accuracy:** Accuracy was computed per question type and topic, followed by an overall assessment across all questions.

The evaluation results, summarized in [Table 2](#), show that the OPUS variant of Claude consistently outperformed other models, achieving the highest accuracy in most categories. GPT-4 also demonstrated strong performance, particularly in material processing and fluid mechanics. As expected from prior studies, larger models such as OPUS and GPT-4 outperformed the smaller LLAMA3-8B, reinforcing the significance of model size in performance [81]. The results suggest that there is significant room for improvement to enhance the accuracy of language models in answering scientific questions. The interactive web app, MaSTeA (Materials Science Teaching Assistant), developed using Streamlit, allows easy model testing to identify LLMs' strengths and weaknesses in different materials science subfields. The interface can be seen in [Figure 6](#).

Table 2: Accuracy of LLMs for each topic

Topic	# Questions	LLaMA-3-8b	Haiku	Sonnet	OPUS	GPT4
Thermodynamics	114	37.72	47.37	55.26	73.68	57.02
Atomic structure	100	32	40	49	64	59
Mechanical behavior	96	22.92	41.67	52.08	71.88	43.75
Material manufacturing	91	43.96	57.14	56.04	80.22	68.13
Material applications	53	52.83	64.15	77.36	92.45	86.79
Phase transition	41	31.71	46.34	65.85	70.73	63.41
Electrical properties	36	33.33	25	55.56	72.22	44.44
Material processing	35	48.57	54.29	74.29	88.57	88.57
Transport phenomena	24	37.5	70.83	58.33	87.5	62.5
Magnetic properties	15	26.67	46.67	46.67	66.67	60
Material characterization	14	78.57	57.14	85.71	92.86	71.43
Fluid mechanics	14	21.43	50	57.14	78.57	85.71
Material testing	9	77.78	66.67	100	100	100
Miscellaneous	8	62.5	62.5	62.5	75	62.5

With MaSTeA, the team demonstrated the potential of interactive tools to help students practice answering questions and learn the steps to reach the correct solution. By evaluating LLM performance, the goal was to guide future model development and identify areas for improvement. The results suggest that LLMs can benefit from strategies such as self-consistency [82] and retrieval-augmented generation (RAG) [83], which have been shown to reduce hallucinations and increase accuracy. Additionally, integrating advanced reasoning models could further improve performance. Recent advancements in domain-specific LLMs, such as LLaMat [84], highlight the potential of specialized training to enhance scientific reasoning.

5 Research Data Management and Automation

Various submissions were received that attempt to enhance the management, accessibility, and automation of scientific data workflows using LLMs. These efforts, often leveraging multimodal agents, aim to simplify complex data handling, improve reproducibility, and accelerate insights across diverse scientific disciplines. We highlight two exemplar projects: “yeLLowhaMmer” a multimodal LLM-based data management agent that automates data handling within electronic lab notebooks (ELNs) and laboratory information manage-

Materials Science Teaching Assistant

Settings Select your topic to study

Select a model

Topics

GPT4

Fluid

List of question

MCQS-NUM-1

MCQS - Multiple choice questions, MCQS-NUMS - Numerical MCQS, MATCH - Matching questions, NUM - Numerical questions

A fluid is flowing with a velocity of 0.5 m/s on a plate moving with a velocity of 0.01 m/s in the same direction. The velocity at the interface of the fluid and plate is (A) 0.0 m/s (B) 0.01 m/s (C) 0.255 m/s (D) 0.50 m/s

Get answer

We are given the velocity of the fluid and the velocity of the plate. We need to find the velocity at the interface of the fluid and the plate. "Let's analyze the situation step by step." 1. Velocity of fluid (V_f) = 0.5 m/s 2. Velocity of plate (V_p) = 0.01 m/s 3. Since both the fluid and plate are moving in the same direction the relative velocity between them can be calculated as: Relative velocity (V_r) = $V_f - V_p$ 4. Now substitute the given values in the formula: $V_r = 0.5 \text{ m/s} - 0.01 \text{ m/s} = 0.49 \text{ m/s}$ 5. The velocity at the interface will be the same as the velocity of the plate because the fluid and the plate are in contact with each other without any slippage. Therefore if there is no relative velocity between the fluid and plate at the interface it indicates that their velocities are the same. Hence the velocity at the interface is the same as the velocity of the plate which is: Answer: 0.01 m/s (B)

Correct answer should be B

Figure 6: MaSTeA interface demonstrating a numerical question task. The model arrives at the correct answer by reasoning through the problem, providing students with a step-by-step solution if they struggle to solve it independently.

ment systems (LIMS), and “NOMAD Query Reporter”, an LLM-based agent that uses RAG to generate context-aware summaries from large materials science repositories like NOMAD [85]

5.1 yeLLowhaMMer: A Multi-modal Tool-calling Agent for Accelerated Research Data Management

As scientific data continues to grow in volume and complexity, there is a need for tools that can simplify the job of managing this data to draw insights, increase reproducibility, and accelerate discovery. Digital systems of record, such as electronic lab notebooks (ELNs) or laboratory information management systems (LIMS), have been a great advancement in this area. However, capturing data using, e.g., electronic lab notebooks (ELNs) or laboratory information management systems (LIMS) is laborious, or simply impossible, to accomplish using graphical user interfaces alone. Recent advances in AI present an opportunity to augment how researchers interact with their data, improving scientific data management and allowing scientists to ask scientific questions of these data sources in new ways.

YeLLowhaMmer explored how large language models can be used to simplify and accelerate data handling tasks in order to generate new insights, improve reproducibility, and save time for researchers using the open-source *datalab* [86] ELN/LIMS. Previously, the team had made progress toward this goal by developing a conversational assistant, Whinchat [30], that allows users to ask questions about their data. However, this assistant was unable to take action with a user’s data or seek additional information as is often needed for scientific tasks. Thus, the team developed yeLLowhaMmer as a multimodal large language model (MLLM)-based data management agent capable of taking free-form text and image instructions from users and executing a variety of complex scientific data management tasks.

The agent is powered by commercial MLLMs used within an agentic framework capable of iteratively writing and executing Python code that interacts with *datalab* instances via the *datalab-api* package. In typical usage, a yeLLowhaMmer user might instruct the agent: “Pull up my 10 most recent sample entries and summarize the synthetic approaches used.” In this case, the agent will attempt to write *datalab* python API code to query for the user’s samples in the *datalab* instance and write a human-readable summary based on the result. If the code it generates gives an error (or does not give sufficient information), the agent can iteratively rewrite the program until the task is accomplished successfully. Importantly, this paradigm is enabled by the presence of a structured API for diverse forms of scientific data; which is provided by *datalab* in its open-source schemas and API documentation.

In developing yeLLowhaMmer, the team found that simply copying documentation for the new *datalab-api* package into the system prompt produced poor code. Creating a simplified version with concrete examples and abridged JSON Schema formats proved more effective. The 12,000-character prompt (ca. 3,200 tokens) works well with modern large context models like Claude 3 Haiku. Future scientific libraries might benefit from maintaining both standard documentation and condensed “agents.txt” files optimized for ML agents.

This work shows the opportunity to integrate more tightly into scientific data management workflows to allow researchers to quickly handle complex tasks and efficiently ask questions of all collected data. An important challenge is to find ways to ensure that data curated or modified by such agents will be appropriately ‘credited’ by, for example, visually demarcating AI-generated content, and providing UI pathways for human users to verify or relabel such data in an efficient manner. Finally, recent progress in MLLM’s ability to handle audio and video content in addition to text and images will allow agents to use audiovisual data in real time to provide even more comprehensive user interfaces.

5.2 NOMAD Query Reporter: Automating Research Data Narratives

Research data management (RDM) in materials science includes a wide variety of schemas and data structures. Databases such as NOMAD [85, 87] support extensible context-aware schemas. Hence, the results of a single query may in fact contain various schemas, complicating the data analysis process. NOMAD Query Reporter is a proof-of-concept application built to produce a written summary of the common methodological parameters and standout results in a scientific style. These may serve as the first step in an analysis workflow, or as progenitors of a journal article’s “methods” section.

Given the large size –over 19 million entries– and dynamic nature –open public uploads– of the NOMAD database, retraining or fine-tuning strategies are challenging. Instead, this prototype implements a retrieval-

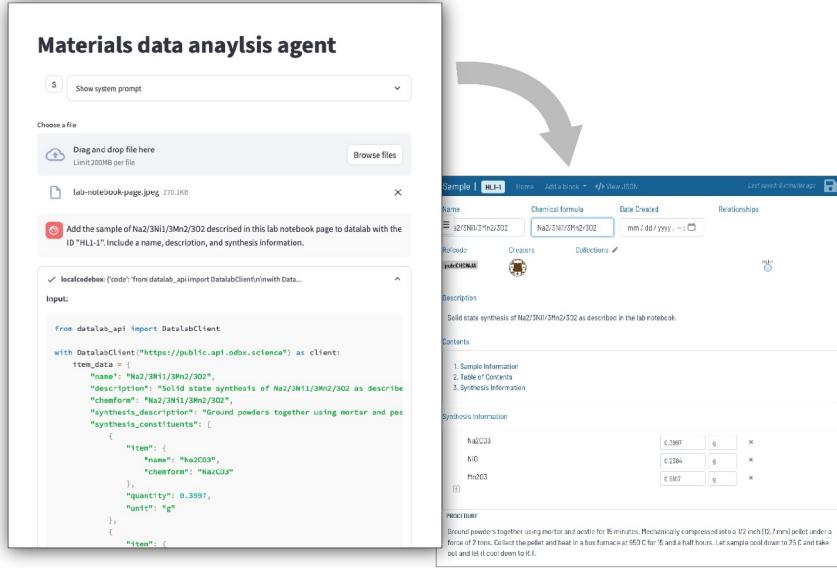


Figure 7: The yeLLowhaMmer multimodal agent can be used for a variety of data management tasks. Here, it is shown automatically adding an entry into the *databar* lab data management system based on an image of a handwritten lab notebook page.

augmented generation (RAG) approach, as defined by Gao et al. [88], to enrich Llama3 (70B version) model’s [89] knowledge base. The team progressively fed data by field into the LLM’s chat-completion API as context. Subsequently, the construction of the summary was completed by topic (i.e., properties, techniques, material composition) in a multi-turn conversation style with the “roles” feature clearly distinguishing the LLM’s tasks from the data provided. Alignment with earlier versions of the chat history is enforced both via low-temperature settings as well as prompt engineering. For a step-by-step overview, see Figure 8.

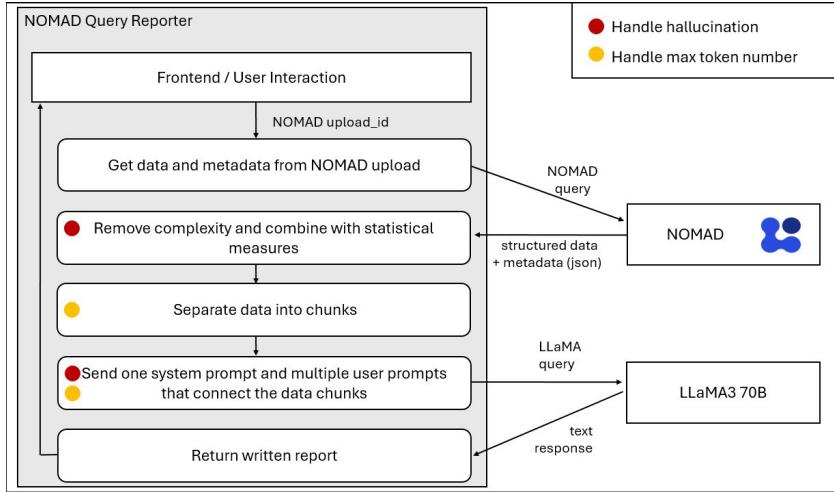


Figure 8: Flowchart of the Query Reporter usage, including the back-end interaction with external resources, i.e., NOMAD and Llama. Intermediate steps managing hallucinations or token limits are marked in red and orange, respectively.

This work highlights the ability of LLMs to augment research data management systems via returning information in formats that are easily understandable by users. While the prototype NOMAD Query Reporter is able to manage homogenized hits well, attempts at extending to manually annotated, heterogeneous data

from ELNs proved challenging. Thus, follow-up work should consider more performant models and advanced RAG and other strategies to improve model context.

6 Hypothesis Generation and Evaluation

LLMs can be leveraged to streamline scientific inquiry, hypothesis generation, and verification. Recent work across psychology, astronomy, and biomedical research demonstrates their capacity to generate novel, validated hypotheses by integrating domain-specific data structures like causal graphs [90, 91, 92, 93, 94]. Although still largely untapped in chemistry and materials science, this approach holds substantial promise for accelerating discovery and innovation in these fields [95, 96, 97].

6.1 Multi-Agent Hypothesis Generation and Verification through Tree of Thoughts and Retrieval Augmented Generation

Scientific discovery thrives on the ability to generate and evaluate new hypotheses efficiently. However, the process of forming meaningful and testable hypotheses often requires extensive background research, domain knowledge, and iterative refinement. Advances in large language models offer an opportunity to assist researchers in streamlining this process, particularly through structured, multi-agent frameworks that systematically generate, evaluate, and refine ideas.

The Thoughtful Beavers team (Soroush Mahjoubi, Aleyna B. Ozhan) designed a multi-agent system to enhance scientific inquiry in materials science. Similar systems have proven useful in social sciences [98], and the system was adapted specifically for hypothesis generation in the domain of cement and concrete. The system consists of specialized agents that work in tandem retrieving background knowledge, generating inspirations, formulating hypotheses, and evaluating their feasibility, utility, and novelty. By leveraging a combination of retrieval-augmented generation, tree-of-thoughts reasoning [99], and LLM-as-a-judge frameworks, this pipeline, which is illustrated in Figure 9, ensures that only the most promising hypotheses emerge from the process.

To test this pipeline, the authors focused on sustainability challenges in concrete design. By processing 66,000 abstracts related to the field, an embedding-based retrieval system was built to extract relevant insights and generate research questions. From this dataset, the approach produced 1,000 structured hypotheses, which were then subjected to rigorous evaluation. The results showed that 243 hypotheses were deemed feasible based on current scientific knowledge, 175 demonstrated practical utility, and 12 stood out as highly novel.

Looking ahead, this framework can be adapted to other material systems or even cross-disciplinary applications. By adjusting the background retrieval process, researchers could apply this method to areas such as ceramics, composites, or biomedical materials. Additionally, cross-pollination of ideas between domains could inspire new lines of research. As LLM capabilities continue to evolve, integrating AI-assisted hypothesis generation with expert validation could significantly accelerate scientific progress while maintaining the critical role of human creativity in innovation.

7 Knowledge Extraction and Reasoning

Extraction of structured scientific knowledge from unstructured text using LLMs to assisting researchers in navigating complex academic content is of wide interest [100, 101, 102, 103]. These systems streamline tasks like named entity recognition and relation extraction, offering flexible solutions tailored to materials science and chemistry [101]. Tool-augmented frameworks help LLMs address complex reasoning by leveraging scientific tools and resources, expanding their utility as assistants in scientific research [104].

7.1 ActiveScience

Extracting and refining knowledge in hard sciences is crucial. While large language models excel in summarization and dialogue generation, they are also prone to generating false information, a phenomenon known

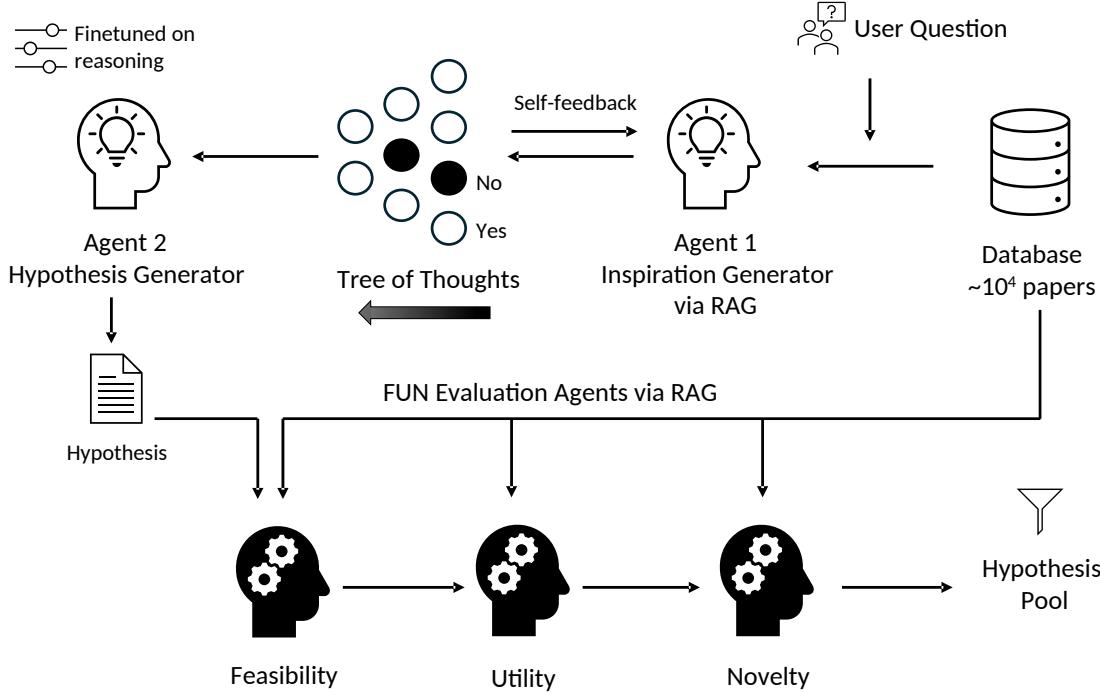


Figure 9: Multi-Agent Hypothesis Generation and Verification Framework. The system uses Retrieval-Augmented Generation, Tree of Thoughts, and Feasibility, Utility, and Novelty evaluation agents to generate and refine hypotheses for sustainable concrete design.

as hallucination. This presents a significant challenge for researchers leveraging LLMs in scientific fields. Various strategies exist to mitigate hallucinations. One approach involves fine-tuning models or constructing additional lightweight models after pretraining, but these methods require substantial computational resources, making them impractical in many cases. A more accessible alternative is retrieval-augmented generation (RAG), which enhances LLMs by incorporating external information. Conceptually, if a fine-tuned model resembles a domain expert with deep knowledge, a pre-trained model is akin to a generalist with broad understanding. By supplying additional context, pre-trained models can generate more accurate and reliable outputs. To address this challenge, Min-Hsueh Chiu introduced an automated framework *Active-Science* that leverages large language models to ingest scientific articles into a knowledge graph and enable natural language queries for domain knowledge extraction. The framework integrates three key components: a data source API, a large language model, and a graph database. While these components can be replaced with equivalent technologies, this work specifically utilizes the ArXiv API [105], GPT-3.5 Turbo [25], and Neo4j [106].

For structured representation of knowledge and relationships, *ActiveScience* employs an ontology that defines key entities such as Application, Property, Material, Element, and Metadata. The ontology design is adaptable and scalable to specific use cases. *ActiveScience* constructs its knowledge graph by extracting relevant triples from scientific articles. Specifically, prompts are generated using the predefined ontology and the introduction sections of articles to produce Cypher import statements containing structured triples, such as (Material: "Nanowire") - [HAS ELEMENT] - (Element: "Aluminum") and (Material: "Nanowire") - [HAS.FORMULA] - (Formula: "Al-Si alloy"). These triples are then imported into a Neo4j graph database. To facilitate RAG, the GraphCypherQACChain module from LangChain is employed. For instance, given the query, "Retrieve the top three reference URLs where the Property contains 'opti' GraphCypherQACChain dynamically generates a Cypher query based on the predefined ontology schema, executes it within Neo4j, and returns the relevant results. The processes of query generation and natural language processing are handled by LLMs. The pipeline and output are illustrated in Figure 10.

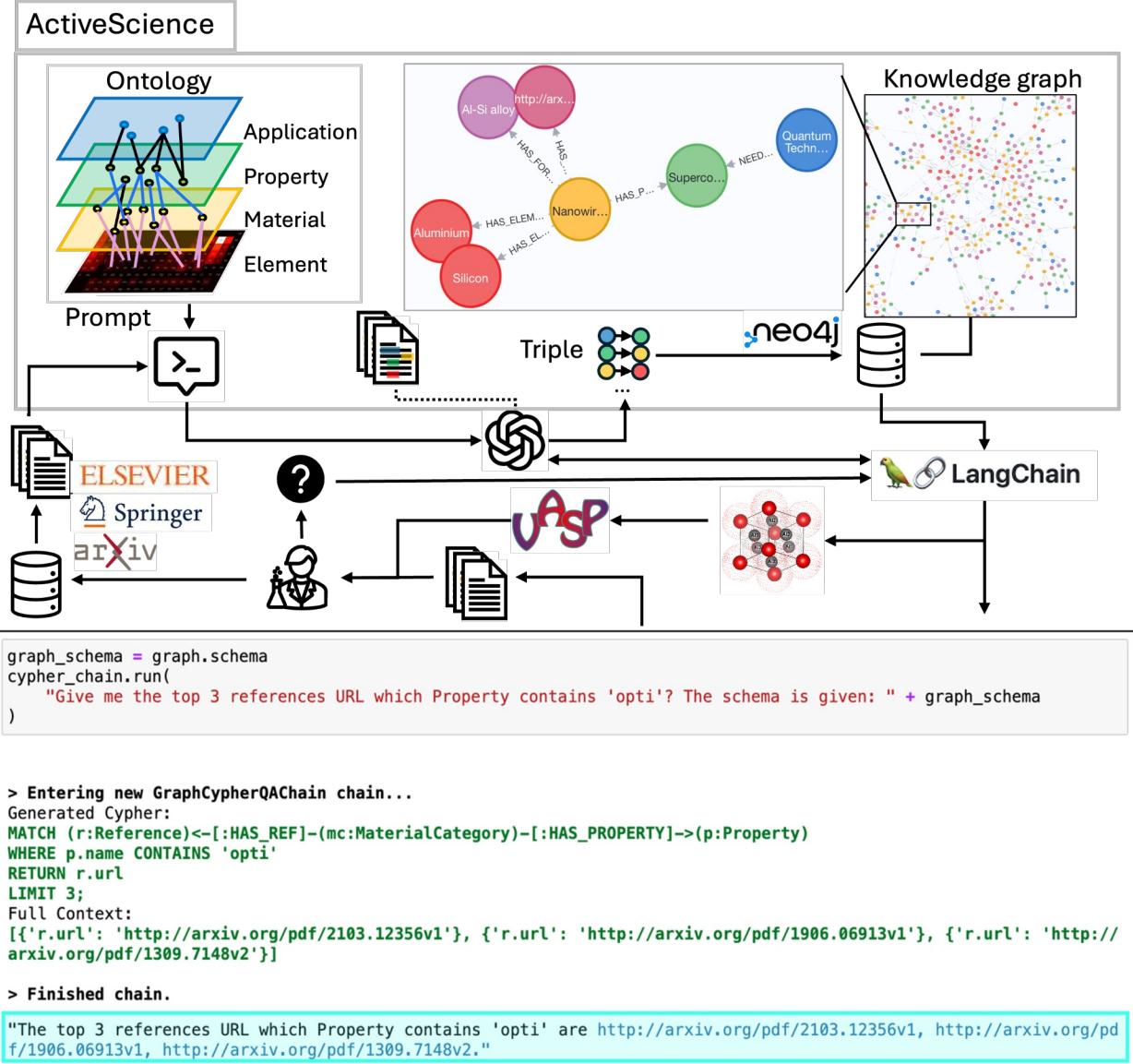


Figure 10: ActiveScience Framework for Knowledge Extraction. The system combines ontology-driven prompts, large language models, and a Neo4j knowledge graph to enable natural language queries and retrieval-augmented generation (RAG) for scientific research insights. Additionally, a code snippet demonstrating the use of LangChain is shown.

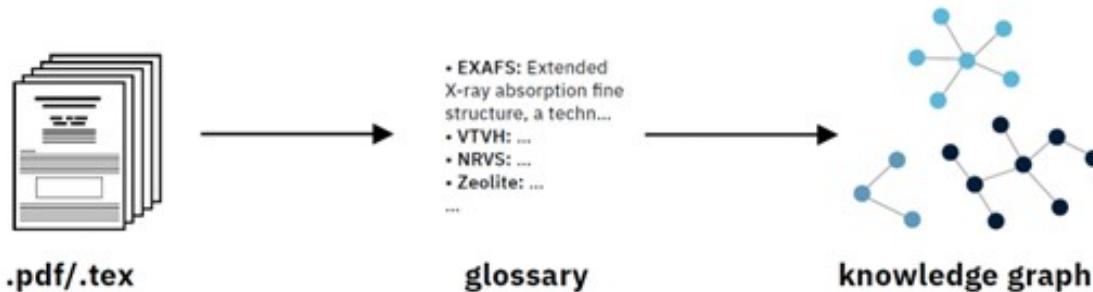


Figure 11: Schematic overview of the *GlossaGen* project. Textual information is extracted from PDF and LaTeX files and a glossary is generated with terms and their definition. From this, a knowledge graph is created, showing entities and relationships between terms.

7.2 **GlossaGen**

Academic literature, particularly review articles and grant applications, would substantially benefit from the inclusion of comprehensive glossaries elucidating complex terminology and discipline-specific nomenclature. However, the manual generation of such reference materials is a labor-intensive and redundant process. To address this limitation, Lederbauer et al. developed *GlossaGen*, which leverages large language models to automate the creation of glossaries for academic articles and grant proposals, eliminating the need for time-consuming manual compilation. To efficiently process PDF or TeX articles, a pre-processing step automatically extracts the title and DOI, and chunks the text into smaller, context-preserving sections for LLM analysis. LLMs such as GPT-3.5-Turbo [25] and GPT-4-Turbo [24] then identify and define scientific terms with the help of Typed Predictors [107] and Chain-of-Thought [108] prompting, ensuring well-structured, contextually relevant, and accurate outputs. The generated glossary is not merely presented as a list of terms but also as an ontology-based knowledge graph using Neo4J [106] and Graph Maker [109], visualizing the intricate relationships between various technical concepts (Figure 11). A user-friendly interface prototype, developed with Gradio [110], enables seamless interaction and customization, making the system accessible to researchers.

Future enhancements could focus on improving glossary output through LLM fine-tuning, integrating retrieval-augmented generation, and enabling article image parsing. Additionally, the system can better support users by allowing them to input specific terms for glossary explanations, ensuring comprehensive coverage even when LLMs omit key concepts. Overall, *GlossaGen*'s rapid development and promising capabilities highlight the potential of LLMs to assist researchers in their scientific outreach.

7.3 **ChemQA**

Foundation models exhibit strong capabilities in chemistry reasoning, yet their performance across different input modalities — text, images, and their combination, remains underexplored. Building upon prior benchmarks such as IsoBench [111] and ChemLLMBench [112] the **VizChem** team (Khalighinejad et al.) introduced *ChemQA* [113], a multimodal question-answering dataset designed to assess chemistry reasoning in language models.

ChemQA comprises five distinct QA tasks: atom counting, molecular weight calculation, name conversion, molecule captioning, and retrosynthesis planning. Each task is formulated with both molecular images and textual SMILES representations, enabling a systematic study of multimodal reasoning in chemistry.

The evaluation results, shown in Figure 12, reveal that the models achieve higher accuracy when provided with both text and images, while the performance drops significantly with image-only inputs. Notably, Claude 3 Opus demonstrates superior performance in text-based tasks, whereas Gemini Pro and GPT-4 Turbo excel in multimodal settings [114, 115, 116]. These findings highlight the limitations of current models in processing visual chemistry data independently.

By introducing *ChemQA*, the **VizChem** team underscored the need for enhanced multimodal reasoning in chemistry. Future work should focus on improving the integration of textual and visual representations

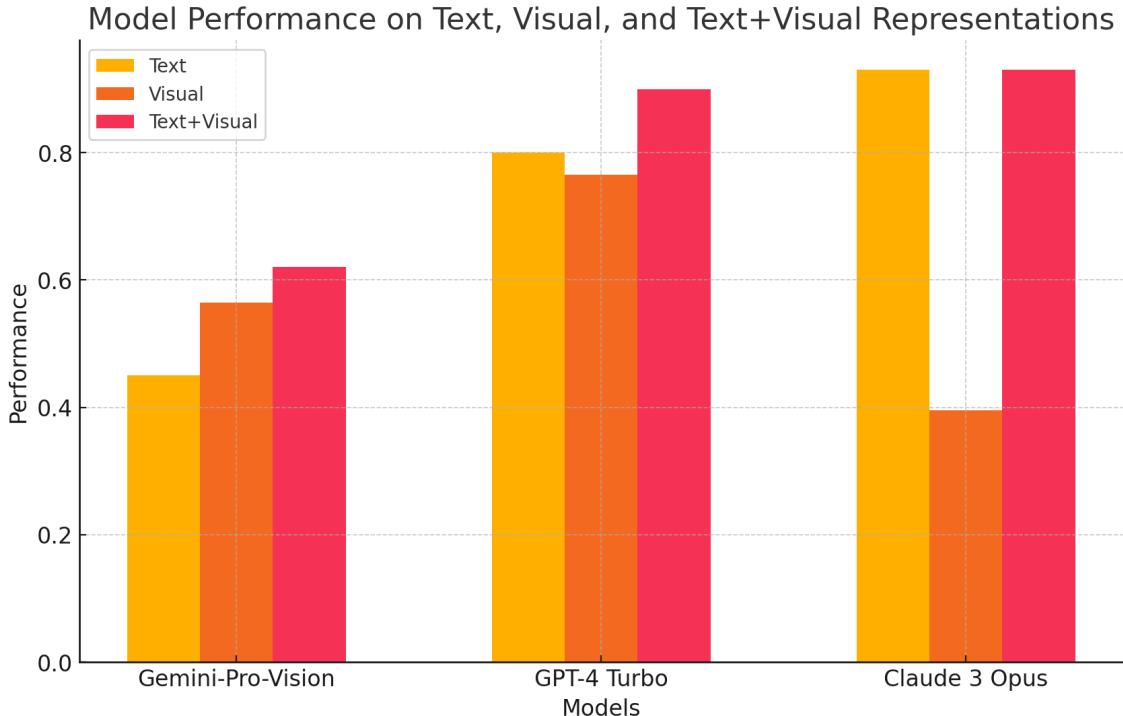


Figure 12: Performance of Gemini Pro, GPT-4 Turbo, and Claude3 Opus on text, visual, and text+visual representations. The plot shows that models achieve higher accuracy with combined text and visual inputs compared to visual-only inputs.

to advance AI-driven scientific analysis.

Hackathon Event Overview

The second annual Large Language Model (LLM) Hackathon for Applications in Materials Science and Chemistry was held on May 9, 2024, bringing together a global network of researchers, students, and industry professionals. With 556 registered participants and over 120 active contributors forming 34 teams, the event spanned multiple time zones and research domains, underscoring the broad interest in applying LLMs to scientific discovery (Figure 13). This hackathon built on the success of the previous year's event, described in detail in [30]. The hybrid format included physical hubs in Toronto, Montreal, San Francisco, Berlin, Lausanne, and Tokyo, fostering interdisciplinary collaboration across institutions and time zones. The event began with a kickoff panel featuring experts Elsa Olivetti (MIT), Jon Reifsneider (Duke), Michael Craig (Valence Laboratories), and Marwin Segler (Microsoft), who discussed the evolving role of LLMs in scientific research.

The charge of the hackathon was intentionally open-ended: to explore the vast potential application space and create tangible demonstrations of the most innovative, impactful, and scalable solutions within a constrained timeframe. Participants leveraged open-source and best-in-class multimodal models to tackle challenges in materials science and chemistry. These teams submitted projects covering molecular property prediction, materials design, automation, hypothesis generation, and knowledge extraction, demonstrating the versatility of LLMs in scientific research. Many incorporated retrieval-augmented generation (RAG), multi-agent reasoning, and natural language interfaces, showcasing AI's expanding role in scientific discovery.

Beyond technical contributions, the hackathon fostered a global research community, with 483 researchers continuing collaborations via Slack. The event demonstrated the value of structured collaboration in accelerating AI-driven discovery and bridging computational scientists, experimentalists, and AI researchers.

LLM HACKATHON FOR APPLICATIONS IN MATERIALS AND CHEMISTRY

- 1 ONLINE HUB FOR WORLDWIDE ACCESS
- 6 IN-PERSON LOCATIONS

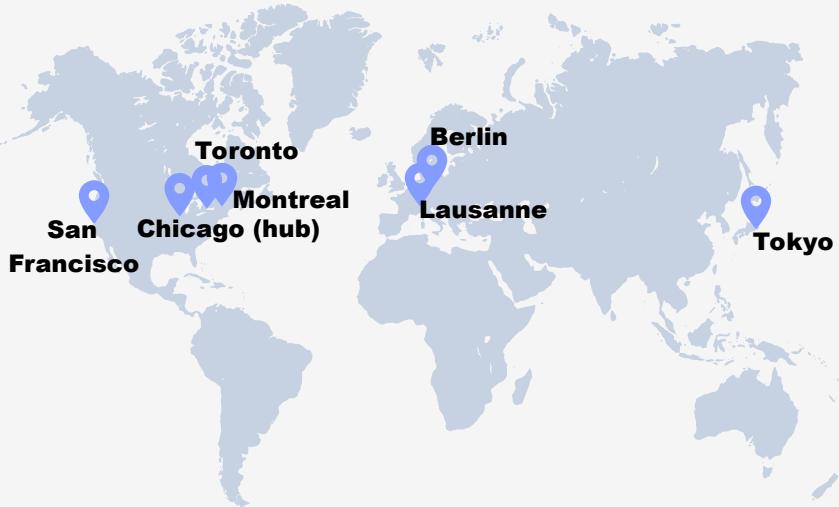


Figure 13: LLM Hackathon for Applications in Materials and Chemistry hybrid hackathon. Researchers were able to participate from both remote and in-person locations (purple pins).

Conclusion

The LLM Hackathon for Applications in Materials Science and Chemistry has demonstrated the dual utility and immense promise of LLMs to impact materials science and chemistry research across the entire lifecycle. Together, the projects 1) demonstrate the promise of a new set of tools that together form a cohesive patchwork to perform tasks ranging from hypothesis generation to data extraction, novel interface design, analysis of results, and more; and 2) showcase the ability of LLMs to enable rapid prototyping and exploration of the application space. Participants effectively utilized LLMs to explore solutions to specific challenges while rapidly evaluating their ideas over just a short 24-hour period, highlighting compelling abilities to enhance the efficiency and creativity of research processes across many applications. It's important to note that many projects benefited from significant advancements in LLM performance since the previous year's hackathon. That is, the performance across the application space was improved simply via the release of more powerful versions of Gemini, ChatGPT, Claude, Llama, and other models and more easily accessible APIs and examples. If this trend continues, we expect to see even broader applications in subsequent hackathons and in materials science and chemistry more generally. We note that reliance on proprietary APIs raises reproducibility concerns as models evolve or are deprecated, while infrastructure demands for training, fine-tuning, or running inference on models with parameters reaching hundreds of billions require yet more computational resources, leading to significant infrastructure roadblocks to further academic work.

Importantly, the hybrid hackathon format itself proved to be an effective mechanism to foster interdisciplinary collaboration, accelerate the prototyping of AI-driven tools, and create a global community of researchers engaged in exploring LLM applications. The hybrid format, combining physical hubs with virtual participation, facilitated knowledge exchange across continents, highlighting the importance of accessible, multimodal, and scalable approaches to scientific innovation.

Acknowledgments

Planning for this event was supported by NSF Awards #2226419 and #220989. We would like to thank event sponsors who provided platform credits and prizes for teams, including RadicalAI, Iteratec, Reincarnate, Acceleration Consortium, and Neo4j. Site coordinators include: Brandon Lines, Philippe Schwaller, Pepe Marquez, Mehrad Ansari and Seyed Mohamad Moosavi. Mohamad Moosavi acknowledges support from the Data Science Institute at the University of Toronto for organizing events related to LLMs. Mehrad Ansari acknowledges Mahyar Rajabi, Seyed Mohamad Moosavi, and Amro Aswad for their feedback on the project. Aakash Naik, Katharina Uelzen, and Janine George would like to acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding their work on this project by providing generous computing time on the GCS Supercomputer SuperMUC-NG at Leibniz Super computing Centre (www.lrz.de) (project pn73da).

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] K. M. Jablonka, P. Schwaller, A. Ortega-Guerrero, and B. Smit, "Leveraging large language models for predictive chemistry," *Nature Machine Intelligence*, vol. 6, p. 161–169, Feb. 2024.
- [2] D. Bhattacharya, H. J. Cassady, M. A. Hickner, and W. F. Reinhart, "Large language models as molecular design engines," *Journal of Chemical Information and Modeling*, vol. 64, pp. 7086–7096, 2024.
- [3] K. Choudhary, "AtomGPT: Atomistic Generative Pretrained Transformer for Forward and Inverse Materials Design," *The Journal of Physical Chemistry Letters*, vol. 15, pp. 6909–6917, 2024.

- [4] J. Yin, A. Bose, G. Cong, I. Lyngaa, and Q. Anthony, “Comparative study of large language model architectures on frontier,” in *2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 556–569, IEEE, 2024.
- [5] T. Gupta, M. Zaki, N. A. Krishnan, and Mausam, “Matscibert: A materials domain language model for text mining and information extraction,” *npj Computational Materials*, vol. 8, no. 1, p. 102, 2022.
- [6] X. Liu, Y. Wang, T. Yang, X. Liu, and X. Wen, “Alchembert: Exploring lightweight language models for materials informatics,” *ChemRxiv*, 2025.
- [7] A. N. Rubungo, C. Arnold, B. P. Rand, and A. B. Dieng, “Llm-prop: Predicting physical and electronic properties of crystalline solids from their text descriptions,” *arXiv preprint arXiv:2310.14029*, 2023.
- [8] S. Kim, J. Schrier, and Y. Jung, “Explainable synthesizability prediction of inorganic crystal polymorphs using large language models,” *Angewandte Chemie International Edition*, p. e202423950, 2024.
- [9] A. M. Ganose and A. Jain, “Robocrystallographer: automated crystal structure text descriptions and analysis,” *MRS Communications*, vol. 9, no. 3, pp. 874–881, 2019.
- [10] A. Dunn, Q. Wang, A. Ganose, D. Dopp, and A. Jain, “Benchmarking materials property prediction methods: the matbench test set and automatminer reference algorithm,” *npj Computational Materials*, vol. 6, no. 1, p. 138, 2020.
- [11] G. Petretto, S. Dwaraknath, H. PC Miranda, D. Winston, M. Giantomassi, M. J. Van Setten, X. Gonze, K. A. Persson, G. Hautier, and G.-M. Rignanese, “High-throughput density-functional perturbation theory phonons for inorganic materials,” *Scientific data*, vol. 5, no. 1, pp. 1–12, 2018.
- [12] A. Bazgir, R. chandra Praneeth Madugula, and Y. Zhang, “Matagent: A human-in-the-loop multi-agent lilm framework for accelerating the material science discovery cycle,” *AI for Accelerated Materials Design-ICLR 2025*, 2025.
- [13] R. Jacobs, M. P. Polak, L. E. Schultz, H. Mahdavi, V. Honavar, and D. Morgan, “Regression with large language models for materials and molecular property prediction,” 2024.
- [14] A. N. Rubungo, K. Li, J. Hattrick-Simpers, and A. B. Dieng, “Llm4mat-bench: benchmarking large language models for materials property prediction,” *arXiv preprint arXiv:2411.00177*, 2024.
- [15] C. Qian, H. Tang, Z. Yang, H. Liang, and Y. Liu, “Can large language models empower molecular property prediction?,” 2023.
- [16] D. A. Boiko, R. MacKnight, B. Kline, and G. Gomes, “Autonomous chemical research with large language models,” *Nature*, vol. 624, pp. 570–578, 12 2023.
- [17] G. Tom, S. P. Schmid, S. G. Baird, Y. Cao, K. Darvish, H. Hao, S. Lo, S. Pablo-García, E. M. Rajaonson, M. Skreta, N. Yoshikawa, S. Corapi, G. D. Akkoc, F. Strieth-Kalthoff, M. Seifrid, and A. Aspuru-Guzik, “Self-driving laboratories for chemistry and materials science,” *Chemical Reviews*, vol. 124, pp. 9633–9732, 8 2024.
- [18] H. Zhang, Y. Song, Z. Hou, S. Miret, and B. Liu, “Honeycomb: A flexible lilm-based agent system for materials science,” 2024.
- [19] A. M. Bran, S. Cox, O. Schilter, C. Baldassari, A. D. White, and P. Schwaller, “Augmenting large language models with chemistry tools,” *Nature Machine Intelligence*, vol. 6, no. 5, pp. 525–535, 2024.
- [20] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Srivankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron, B. Tang, B. Chern, C. Caucheteux, C. Nayak, C. Bi, C. Marra, C. McConnell, C. Keller, C. Touret, C. Wu, C. Wong, C. C. Ferrer, C. Nikolaidis, D. Allonsius, D. Song, D. Pintz, D. Livshits, D. Wyatt, D. Esiobu, D. Choudhary, D. Mahajan, D. Garcia-Olano, D. Perino, D. Hupkes,

E. Lakomkin, E. AlBadawy, E. Lobanova, E. Dinan, E. M. Smith, F. Radenovic, F. Guzmán, F. Zhang, G. Synnaeve, G. Lee, G. L. Anderson, G. Thattai, G. Nail, G. Mialon, G. Pang, G. Cucurell, H. Nguyen, H. Korevaar, H. Xu, H. Touvron, I. Zarov, I. A. Ibarra, I. Kloumann, I. Misra, I. Evtimov, J. Zhang, J. Copet, J. Lee, J. Geffert, J. Vranes, J. Park, J. Mahadeokar, J. Shah, J. van der Linde, J. Billock, J. Hong, J. Lee, J. Fu, J. Chi, J. Huang, J. Liu, J. Wang, J. Yu, J. Bitton, J. Spisak, J. Park, J. Rocca, J. Johnstun, J. Saxe, J. Jia, K. V. Alwala, K. Prasad, K. Upasani, K. Plawiak, K. Li, K. Heafield, K. Stone, K. El-Arini, K. Iyer, K. Malik, K. Chiu, K. Bhalla, K. Lakhotia, L. Rantala-Yeary, L. van der Maaten, L. Chen, L. Tan, L. Jenkins, L. Martin, L. Madaan, L. Malo, L. Blecher, L. Landzaat, L. de Oliveira, M. Muzzi, M. Pasupuleti, M. Singh, M. Paluri, M. Kardas, M. Tsim-poukelli, M. Oldham, M. Rita, M. Pavlova, M. Kambadur, M. Lewis, M. Si, M. K. Singh, M. Hassan, N. Goyal, N. Torabi, N. Bashlykov, N. Bogoychev, N. Chatterji, N. Zhang, O. Duchenne, O. C. elebi, P. Alrassy, P. Zhang, P. Li, P. Vasic, P. Weng, P. Bhargava, P. Dubal, P. Krishnan, P. S. Koura, P. Xu, Q. He, Q. Dong, R. Srinivasan, R. Ganapathy, R. Calderer, R. S. Cabral, R. Stojnic, R. Raileanu, R. Maheswari, R. Girdhar, R. Patel, R. Sauvestre, R. Polidoro, R. Sumbaly, R. Taylor, R. Silva, R. Hou, R. Wang, S. Hosseini, S. Chennabasappa, S. Singh, S. Bell, S. S. Kim, S. Edunov, S. Nie, S. Narang, S. Raparthy, S. Shen, S. Wan, S. Bhosale, S. Zhang, S. Vandenhende, S. Batra, S. Whitman, S. Sootla, S. Collot, S. Gururangan, S. Borodinsky, T. Herman, T. Fowler, T. Sheasha, T. Georgiou, T. Scialom, T. Speckbacher, T. Mihaylov, T. Xiao, U. Karn, V. Goswami, V. Gupta, V. Ramanathan, V. Kerkez, V. Gonguet, V. Do, V. Vogeti, V. Albiero, V. Petrovic, W. Chu, W. Xiong, W. Fu, W. Meers, X. Martinet, X. Wang, X. Wang, X. E. Tan, X. Xia, X. Xie, X. Jia, X. Wang, Y. Goldschlag, Y. Gaur, Y. Babaei, Y. Wen, Y. Song, Y. Zhang, Y. Li, Y. Mao, Z. D. Coudert, Z. Yan, Z. Chen, Z. Papakipos, A. Singh, A. Srivastava, A. Jain, A. Kelsey, A. Shajnfeld, A. Gangidi, A. Victoria, A. Goldstand, A. Menon, A. Sharma, A. Boesenberg, A. Baevski, A. Feinstein, A. Kallet, A. Sangani, A. Teo, A. Yunus, A. Lupu, A. Alvarado, A. Caples, A. Gu, A. Ho, A. Poulton, A. Ryan, A. Ramchandani, A. Dong, A. Franco, A. Goyal, A. Saraf, A. Chowdhury, A. Gabriel, A. Bharambe, A. Eisenman, A. Yazdan, B. James, B. Maurer, B. Leonhardi, B. Huang, B. Loyd, B. D. Paola, B. Paranjape, B. Liu, B. Wu, B. Ni, B. Hancock, B. Wasti, B. Spence, B. Stojkovic, B. Gamido, B. Montalvo, C. Parker, C. Burton, C. Mejia, C. Liu, C. Wang, C. Kim, C. Zhou, C. Hu, C.-H. Chu, C. Cai, C. Tindal, C. Feichtenhofer, C. Gao, D. Civin, D. Beaty, D. Kreymer, D. Li, D. Adkins, D. Xu, D. Testuggine, D. David, D. Parikh, D. Liskovich, D. Foss, D. Wang, D. Le, D. Holland, E. Dowling, E. Jamil, E. Montgomery, E. Presani, E. Hahn, E. Wood, E.-T. Le, E. Brinkman, E. Arcaute, E. Dunbar, E. Smothers, F. Sun, F. Kreuk, F. Tian, F. Kokkinos, F. Ozgenel, F. Caggioni, F. Kanayet, F. Seide, G. M. Florez, G. Schwarz, G. Badeer, G. Swee, G. Halpern, G. Herman, G. Sizov, Guangyi, Zhang, G. Lakshminarayanan, H. Inan, H. Shojanazeri, H. Zou, H. Wang, H. Zha, H. Habeeb, H. Rudolph, H. Suk, H. Aspegren, H. Goldman, H. Zhan, I. Damlaj, I. Molybog, I. Tufanov, I. Leontiadis, I.-E. Veliche, I. Gat, J. Weissman, J. Geboski, J. Kohli, J. Lam, J. Asher, J.-B. Gaya, J. Marcus, J. Tang, J. Chan, J. Zhen, J. Reizenstein, J. Teboul, J. Zhong, J. Jin, J. Yang, J. Cummings, J. Carvill, J. Shepard, J. McPhie, J. Torres, J. Ginsburg, J. Wang, K. Wu, K. H. U. K. Saxena, K. Khandelwal, K. Zand, K. Matosich, K. Veeraraghavan, K. Michelena, K. Li, K. Jagadeesh, K. Huang, K. Chawla, K. Huang, L. Chen, L. Garg, L. A. L. Silva, L. Bell, L. Zhang, L. Guo, L. Yu, L. Moshkovich, L. Wehrstedt, M. Khabsa, M. Avalani, M. Bhatt, M. Mankus, M. Hasson, M. Lennie, M. Reso, M. Groshev, M. Naumov, M. Lathi, M. Keneally, M. Liu, M. L. Seltzer, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, M. Clark, M. Macey, M. Wang, M. J. Hermoso, M. Metanat, M. Rastegari, M. Bansal, N. Santhanam, N. Parks, N. White, N. Bawa, N. Singhal, N. Egebo, N. Usunier, N. Mehta, N. P. Laptev, N. Dong, N. Cheng, O. Chernoguz, O. Hart, O. Salpekar, O. Kalinli, P. Kent, P. Parekh, P. Saab, P. Balaji, P. Rittner, P. Bontrager, P. Roux, P. Dollar, P. Zvyagina, P. Ratanchandani, P. Yuvraj, Q. Liang, R. Alao, R. Rodriguez, R. Ayub, R. Murthy, R. Nayani, R. Mitra, R. Parthasarathy, R. Li, R. Hogan, R. Battey, R. Wang, R. Howes, R. Rinott, S. Mehta, S. Siby, S. J. Bondu, S. Datta, S. Chugh, S. Hunt, S. Dhillon, S. Sidorov, S. Pan, S. Mahajan, S. Verma, S. Yamamoto, S. Ramaswamy, S. Lindsay, S. Lindsay, S. Feng, S. Lin, S. C. Zha, S. Patil, S. Shankar, S. Zhang, S. Zhang, S. Wang, S. Agarwal, S. Sajuyigbe, S. Chintala, S. Max, S. Chen, S. Kehoe, S. Satterfield, S. Govindaprasad, S. Gupta, S. Deng, S. Cho, S. Virk, S. Subramanian, S. Choudhury, S. Goldman, T. Remez, T. Glaser, T. Best, T. Koehler, T. Robinson, T. Li, T. Zhang, T. Matthews, T. Chou, T. Shaked, V. Vontimitta, V. Ajayi, V. Montanez, V. Mohan, V. S. Kumar, V. Mangla, V. Ionescu, V. Poenaru, V. T. Mihaleescu,

V. Ivanov, W. Li, W. Wang, W. Jiang, W. Bouaziz, W. Constable, X. Tang, X. Wu, X. Wang, X. Wu, X. Gao, Y. Kleinman, Y. Chen, Y. Hu, Y. Jia, Y. Qi, Y. Li, Y. Zhang, Y. Zhang, Y. Adi, Y. Nam, Yu, Wang, Y. Zhao, Y. Hao, Y. Qian, Y. Li, Y. He, Z. Rait, Z. DeVito, Z. Rosnbrick, Z. Wen, Z. Yang, Z. Zhao, and Z. Ma, "The llama 3 herd of models," 2024.

- [21] M. Abdin, J. Aneja, H. Awadalla, A. Awadallah, A. A. Awan, N. Bach, A. Bahree, A. Bakhtiari, J. Bao, H. Behl, A. Benhaim, M. Bilenko, J. BJORCK, S. Bubeck, M. Cai, Q. Cai, V. Chaudhary, D. Chen, D. Chen, W. Chen, Y.-C. Chen, Y.-L. Chen, H. Cheng, P. Chopra, X. Dai, M. Dixon, R. Eldan, V. Fragoso, J. Gao, M. Gao, M. Gao, A. Garg, A. D. Giorno, A. Goswami, S. Gunasekar, E. Haider, J. Hao, R. J. Hewett, W. Hu, J. Huynh, D. Iter, S. A. Jacobs, M. Javaheripi, X. Jin, N. Karampatziakis, P. Kauffmann, M. Khademi, D. Kim, Y. J. Kim, L. Kurilenko, J. R. Lee, Y. T. Lee, Y. Li, Y. Li, C. Liang, L. Liden, X. Lin, Z. Lin, C. Liu, L. Liu, M. Liu, W. Liu, X. Liu, C. Luo, P. Madan, A. Mahmoudzadeh, D. Majercak, M. Mazzola, C. C. T. Mendes, A. Mitra, H. Modi, A. Nguyen, B. Norick, B. Patra, D. Perez-Becker, T. Portet, R. Pryzant, H. Qin, M. Radmilac, L. Ren, G. de Rosa, C. Rosset, S. Roy, O. Ruwase, O. Saarikivi, A. Saied, A. Salim, M. Santacroce, S. Shah, N. Shang, H. Sharma, Y. Shen, S. Shukla, X. Song, M. Tanaka, A. Tupini, P. Vaddamanu, C. Wang, G. Wang, L. Wang, S. Wang, X. Wang, Y. Wang, R. Ward, W. Wen, P. Witte, H. Wu, X. Wu, M. Wyatt, B. Xiao, C. Xu, J. Xu, W. Xu, J. Xue, S. Yadav, F. Yang, J. Yang, Y. Yang, Z. Yang, D. Yu, L. Yuan, C. Zhang, C. Zhang, J. Zhang, L. L. Zhang, Y. Zhang, Y. Zhang, Y. Zhang, and X. Zhou, "Phi-3 technical report: A highly capable language model locally on your phone," 2024.
- [22] Anthropic, "The claude 3 model family: Opus, sonnet, haiku," 2024.
- [23] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. de las Casas, E. B. Hanna, F. Bressand, G. Lengyel, G. Bour, G. Lample, L. R. Lavaud, L. Saulnier, M.-A. Lachaux, P. Stock, S. Subramanian, S. Yang, S. Antoniak, T. L. Scao, T. Gervet, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mixtral of experts," 2024.
- [24] OpenAI, "Gpt-4-turbo and gpt-4." Available at <https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-2> 2023. Accessed:February 24, 2025.
- [25] OpenAI, "Gpt-3.5-turbo." Available at <https://platform.openai.com/docs/models/gpt-3-5-turbo> 2023. Accessed:February 24, 2025.
- [26] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenstschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fullford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kaftan, Lukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Lukasz Kondraciuk, A. Kondrich, A. Konstantinidis, K. Koscic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. M'ely, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto,

- Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph, "Gpt-4 technical report," 2024.
- [27] A. Nolte, L. B. Hayden, and J. D. Herbsleb, "How to support newcomers in scientific hackathons - an action research study on expert mentoring," *Proceedings of the ACM on Human-Computer Interaction*, vol. 4, pp. 1–23, 5 2020.
- [28] E. P. P. Pe-Than and J. D. Herbsleb, "Understanding hackathons for science: Collaboration, affordances, and outcomes," in *Lecture Notes in Computer Science*, pp. 27–37, Springer International Publishing, 2019.
- [29] B. Heller, A. Amir, R. Waxman, and Y. Maaravi, "Hack your organizational innovation: literature review and integrative model for running hackathons," *Journal of Innovation and Entrepreneurship*, vol. 12, 3 2023.
- [30] K. M. Jablonka, Q. Ai, A. Al-Feghali, S. Badhwar, J. D. Bocarsly, A. M. Bran, S. Bringuier, L. C. Brinson, K. Choudhary, D. Circi, S. Cox, W. A. de Jong, M. L. Evans, N. Gastellu, J. Genzling, M. V. Gil, A. K. Gupta, Z. Hong, A. Imran, S. Kruschwitz, A. Labarre, J. L'ala, T. Liu, S. Ma, S. Majumdar, G. W. Merz, N. Moitessier, E. Moubarak, B. Mourño, B. Pelkie, M. Pieler, M. C. Ramos, B. Ranković, S. G. Rodrigues, J. N. Sanders, P. Schwaller, M. Schwarting, J. Shi, B. Smit, B. E. Smith, J. Van Herck, C. Völker, L. Ward, S. Warren, B. Weiser, S. Zhang, X. Zhang, G. A. Zia, A. Scourtas, K. J. Schmidt, I. Foster, A. D. White, and B. Blaiszik, "14 examples of how llms can transform materials science and chemistry: a reflection on a large language model hackathon," *Digital Discovery*, vol. 2, no. 5, pp. 1233–1250, 2023.
- [31] Y. Zimmermann, A. Bazgir, Z. Afzal, F. Agbere, Q. Ai, N. Alampara, A. Al-Feghali, M. Ansari, D. Antypov, A. Aswad, J. Bai, V. Baibakova, D. D. Biswajeet, E. Bitzek, J. D. Bocarsly, A. Borisova, A. M. Bran, L. C. Brinson, M. M. Calderon, A. Canalicchio, V. Chen, Y. Chiang, D. Circi, B. Charmes, V. Chaudhary, Z. Chen, M.-H. Chiu, J. Clymo, K. Dabhadkar, N. Daelman, A. Datar, W. A. de Jong, M. L. Evans, M. G. Fard, G. Fisicaro, A. S. Gangan, J. George, J. D. C. Gonzalez, M. Götte, A. K. Gupta, H. Harb, P. Hong, A. Ibrahim, A. Ilyas, A. Imran, K. Ishimwe, R. Issa, K. M. Jablonka, C. Jones, T. R. Josephson, G. Juhasz, S. Kapoor, R. Kang, G. Khalighinejad, S. Khan, S. Klawohn, S. Kuman, A. N. Ladines, S. Leang, M. Lederbauer, Sheng-Lun, Liao, H. Liu, X. Liu, S. Lo, S. Madireddy, P. R. Maharana, S. Maheshwari, S. Mahjoubi, J. A. Márquez, R. Mills, T. Mohanty, B. Mohr, S. M. Moosavi, A. Moßhammer, A. D. Naghdi, A. Naik, O. Narykov, H. Näsström, V. Nguyen, X. Ni, D. O'Connor, T. Olayiwola, F. Ottomano, A. B. Ozhan, S. Pagel, C. Parida, J. Park, V. Patel, E. Patyukova, M. H. Petersen, L. Pinto, J. M. Pizarro, D. Plessers, T. Pradhan, U. Pratiush, C. Puli, A. Qin, M. Rajabi, F. Ricci, E. Risch, M. Ríos-García, A. Roy, T. Rug, H. M. Sayeed, M. Scheidgen, M. Schilling-Wilhelmi, M. Schloz, F. Schöppach, J. Schumann, P. Schwaller, M. Schwarting, S. Sharlin, K. Shen, J. Shi, P. Si, J. D'Souza, T. Sparks, S. Sudhakar, L. Talirz, D. Tang, O. Taran, C. Terboven, M. Tropin, A. Tsymbal, K. Ueltzen, P. A. Unzueta, A. Vasan, T. Vinchurkar, T. Vo, G. Vogel, C. Völker, J. Weinreich, F. Yang, M. Zaki, C. Zhang, S. Zhang, W. Zhang, R. Zhu, S. Zhu, J. Janssen, C. Li, I. Foster, and B. Blaiszik, "Reflections from the 2024 large language model (llm) hackathon for applications in materials science and chemistry," 2025.
- [32] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh,

- D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.
- [33] R. Vacareanu, V. A. Negru, V. Suciu, and M. Surdeanu, "From words to numbers: Your large language model is secretly a capable regressor when given in-context examples," in *First Conference on Language Modeling*, 2024.
- [34] A. A. Naik, C. Ertural, N. Dhamrait, P. Benner, and J. George, "A quantum-chemical bonding database for solid-state materials," *Scientific Data*, vol. 10, no. 1, p. 610, 2023.
- [35] A. A. Naik, K. Ueltzen, C. Ertural, A. J. Jackson, and J. George, "Lobsterpy: A package to automatically analyze lobster runs," *Journal of Open Source Software*, vol. 9, no. 94, p. 6286, 2024.
- [36] A. A. Naik, C. Ertural, N. Dhamrait, P. Benner, and J. George, "A Quantum-Chemical Bonding Database for Solid-State Materials (JSONs: Part 1)." <https://zenodo.org/records/80912023>.
- [37] Matbench, "The matbench test suite, phonon dataset." https://web.archive.org/web/20240712132705/https://matbench.materialsproject.org/Leaderboards%20Per-Task/matbench_v0.1_matbench_phono2024. Accessed:2024-07-12.
- [38] M. H. Daniel Han and U. team, "The unsloth package." <https://github.com/unslothai/unsloth> 2024. Accessed:2024-03-14.
- [39] G. Liu, M. Sun, W. Matusik, M. Jiang, and J. Chen, "Multimodal large language models for inverse molecular design with retrosynthetic planning," 2024.
- [40] S. Jia, C. Zhang, and V. Fung, "Llmatdesign: Autonomous materials discovery with large language models," 2024.
- [41] H. Jang, Y. Jang, J. Kim, and S. Ahn, "Can llms generate diverse molecules? towards alignment with structural diversity," 2025.
- [42] J. Lu, Z. Song, Q. Zhao, Y. Du, Y. Cao, H. Jia, and C. Duan, "Generative design of functional metal complexes utilizing the internal knowledge of large language models," 2024.
- [43] A. Kristiadi, F. Strieth-Kalthoff, M. Skreta, P. Poupart, A. Aspuru-Guzik, and G. Pleiss, "A sober look at LLMs for material discovery: Are they actually good for Bayesian optimization over molecules?," in *Proceedings of the 41st International Conference on Machine Learning* (R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, and F. Berkenkamp, eds.), vol. 235 of *Proceedings of Machine Learning Research*, pp. 25603–25622, PMLR, 21–27 Jul 2024.
- [44] S. Miret and N. M. A. Krishnan, "Are llms ready for real-world materials discovery?," 2024.
- [45] L. Li, H. S. Jung, J. W. Lee, and Y. T. Kang, "Review on applications of metal–organic frameworks for co2 capture and the performance enhancement mechanisms," *Renewable and Sustainable Energy Reviews*, vol. 162, p. 112441, 2022.
- [46] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," *arXiv preprint arXiv:2210.03629*, 2022.
- [47] M. Ansari, J. Watchorn, C. E. Brown, and J. S. Brown, "dziner: Rational inverse design of materials with ai agents," *arXiv preprint arXiv:2410.03963*, 2024.
- [48] M. Usman, S. Mendiratta, and K.-L. Lu, "Semiconductor metal–organic frameworks: future low-bandgap materials," *Advanced Materials*, vol. 29, no. 6, p. 1605071, 2017.
- [49] E. Flage-Larsen, A. Røyset, J. H. Cavka, and K. Thorshaug, "Band gap modulations in uiö metal–organic frameworks," *The Journal of Physical Chemistry C*, vol. 117, no. 40, pp. 20610–20616, 2013.

- [50] L.-M. Yang, G.-Y. Fang, J. Ma, E. Ganz, and S. S. Han, “Band gap engineering of paradigm mof-5,” *Crystal growth & design*, vol. 14, no. 5, pp. 2532–2541, 2014.
- [51] L.-M. Yang, P. Vajeeston, P. Ravindran, H. Fjellvag, and M. Tilset, “Theoretical investigations on the chemical bonding, electronic structure, and optical properties of the metal- organic framework mof-5,” *Inorganic chemistry*, vol. 49, no. 22, pp. 10283–10290, 2010.
- [52] M. Ali, E. Pervaiz, T. Noor, O. Rabi, R. Zahra, and M. Yang, “Recent advancements in mof-based catalysts for applications in electrochemical and photoelectrochemical water splitting: A review,” *International Journal of Energy Research*, vol. 45, no. 2, pp. 1190–1226, 2021.
- [53] Y. Yan, C. Wang, Z. Cai, X. Wang, and F. Xuan, “Tuning electrical and mechanical properties of metal-organic frameworks by metal substitution,” *ACS Applied Materials & Interfaces*, vol. 15, no. 36, pp. 42845–42853, 2023.
- [54] C.-K. Lin, D. Zhao, W.-Y. Gao, Z. Yang, J. Ye, T. Xu, Q. Ge, S. Ma, and D.-J. Liu, “Tunability of band gaps in metal-organic frameworks,” *Inorganic chemistry*, vol. 51, no. 16, pp. 9039–9044, 2012.
- [55] R. Greene, T. Sanders, L. Weng, and A. Neelakantan, “New and improved embedding model,” 2022.
- [56] M. Ansari and S. M. Moosavi, “Agent-based learning of materials datasets from the scientific literature,” *Digital Discovery*, vol. 3, no. 12, pp. 2607–2617, 2024.
- [57] Z. Cao, R. Magar, Y. Wang, and A. Barati Farimani, “Moformer: self-supervised transformer model for metal-organic framework property prediction,” *Journal of the American Chemical Society*, vol. 145, no. 5, pp. 2958–2967, 2023.
- [58] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, “Barlow twins: Self-supervised learning via redundancy reduction,” in *International conference on machine learning*, pp. 12310–12320, PMLR, 2021.
- [59] T. Xie and J. C. Grossman, “Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties,” *Physical review letters*, vol. 120, no. 14, p. 145301, 2018.
- [60] S. M. Moosavi, A. Nandy, K. M. Jablonka, D. Ongari, J. P. Janet, P. G. Boyd, Y. Lee, B. Smit, and H. J. Kulik, “Understanding the diversity of the metal-organic framework ecosystem,” *Nature communications*, vol. 11, no. 1, pp. 1–10, 2020.
- [61] A. S. Rosen, S. M. Iyer, D. Ray, Z. Yao, A. Aspuru-Guzik, L. Gagliardi, J. M. Notestein, and R. Q. Snurr, “Machine learning the quantum-chemical properties of metal-organic frameworks for accelerated materials discovery,” *Matter*, vol. 4, no. 5, pp. 1578–1597, 2021.
- [62] G. Landrum, “Rdkit documentation,” *Release*, vol. 1, no. 1-79, p. 4, 2013.
- [63] OpenAI, “Gpt-4 technical report,” 2023.
- [64] H. Chase, “Langchain,” 10 2022.
- [65] Y. Song, W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang, R. Yao, Y. Tian, and S. Li, “Restgpt: Connecting large language models with real-world restful apis,” 2023.
- [66] K. Darvish, M. Skreta, Y. Zhao, N. Yoshikawa, S. Som, M. Bogdanovic, Y. Cao, H. Hao, H. Xu, A. Aspuru-Guzik, A. Garg, and F. Shkurti, “Organa: A robotic assistant for automated chemistry experimentation and characterization,” 2025.
- [67] LangSim Project, “Langsim.” Available at <https://jan-janssen.com/Lang2024>.
- [68] J. Janssen, S. Surendralal, Y. Lysogorskiy, M. Todorova, T. Hickel, R. Drautz, and J. Neugebauer, “pyiron: An integrated development environment for computational materials science,” *Computational Materials Science*, vol. 163, pp. 24 – 36, 2019.

- [69] H. Chase, “Langchain.”
- [70] I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovacs, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, *et al.*, “A foundation model for atomistic materials chemistry,” *arXiv preprint arXiv:2401.00096*, 2023.
- [71] S. Bauer, P. Benner, T. Bereau, V. Blum, M. Boley, C. Carbogno, C. R. A. Catlow, G. Dehm, S. Eibl, R. Ernstorfer, *et al.*, “Roadmap on data-centric materials science,” *Modelling and Simulation in Materials Science and Engineering*, vol. 32, no. 6, p. 063301, 2024.
- [72] Z. Diao, H. Yamashita, and M. Abe, “Leveraging large language models and social media for automation in scanning probe microscopy,” *arXiv preprint arXiv:2405.15490*, 2024.
- [73] M. C. Yongtao Liu and R. K. Vasudevan, “Synergizing human expertise and ai efficiency with language model for microscopy operation and automated experiment design,” *Machine Learning Science and Technology*, vol. 5, no. 2, 2024.
- [74] J. Madsen and T. Susi, “The abtem code: transmission electron microscopy from first principles,” *Open Research Europe*, 2021.
- [75] C. Meyer, N. Dellby, J. A. Hachtel, T. Lovejoy, A. Mittelberger, and O. Krivanek, “Nion swift: Open source image processing software for instrument control, data acquisition, organization, visualization, and analysis using python,” *Microscopy and Microanalysis*, vol. 25, no. S2, pp. 122–123, 2019.
- [76] L. Yan, L. Sha, L. Zhao, Y. Li, R. Martinez-Maldonado, G. Chen, X. Li, Y. Jin, and D. Gašević, “Practical and ethical challenges of large language models in educationA systematic scoping review,” *British Journal of Educational Technology*, vol. 55, pp. 90–112, 8 2023.
- [77] S. Wang, T. Xu, H. Li, C. Zhang, J. Liang, J. Tang, P. S. Yu, and Q. Wen, “Large language models for education: A survey and outlook,” 2024.
- [78] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günemann, E. Hüllermeier, S. Krusche, G. Kutyniok, T. Michaeli, C. Nerdel, J. Pfeffer, O. Poquet, M. Sailer, A. Schmidt, T. Seidel, M. Stadler, J. Weller, J. Kuhn, and G. Kasneci, “Chatgpt for good? on opportunities and challenges of large language models for education,” *Learning and Individual Differences*, vol. 103, p. 102274, 4 2023.
- [79] M. S. Schäfer, “The notorious gpt:science communication in the age of artificial intelligence,” *Journal of Science Communication*, vol. 22, 5 2023.
- [80] M. Zaki, N. A. Krishnan, *et al.*, “Mascqa: investigating materials science knowledge of large language models,” *Digital Discovery*, vol. 3, no. 2, pp. 313–327, 2024.
- [81] K. Seßler, Y. Rong, E. Gürzüklü, and E. Kasneci, “Benchmarking large language models for math reasoning tasks,” *arXiv preprint arXiv:2408.10839*, 2024.
- [82] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, and D. Zhou, “Self-consistency improves chain of thought reasoning in language models,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [83] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Kettler, M. Lewis, W. tau Yih, T. Rocktäschel, and S. Riedel, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.
- [84] V. Mishra, S. Singh, D. Ahlawat, M. Zaki, V. Bihani, H. S. Grover, B. Mishra, S. Miret, N. Krishnan, *et al.*, “Foundational large language models for materials research,” *arXiv preprint arXiv:2412.09560*, 2024.
- [85] C. Draxl and M. Scheffler, “The nomad laboratory: from data sharing to artificial intelligence,” *Journal of Physics: Materials*, vol. 2, p. 036001, 5 2019.

- [86] M. L. Evans and J. D. Bocarsly, "databar," 2024.
- [87] M. Scheidgen, L. Himanen, A. N. Ladines, D. Sikter, M. Nakhaee, Á. Fekete, T. Chang, A. Golparvar, J. A. Márquez, S. Brockhauser, *et al.*, "Nomad: A distributed web-based platform for managing materials science research data," *Journal of Open Source Software*, vol. 8, no. 90, p. 5388, 2023.
- [88] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, H. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, vol. 2, 2023.
- [89] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Roziere, N. Goyal, E. Hambro, F. Azhar, *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.
- [90] Y. Zhou, H. Liu, T. Srivastava, H. Mei, and C. Tan, "Hypothesis generation with large language models," in *Proceedings of the 1st Workshop on NLP for Science (NLP4Science)*, p. 117–139, Association for Computational Linguistics, 2024.
- [91] A. Abdel-Rehim, H. Zenil, O. Orhobor, M. Fisher, R. J. Collins, E. Bourne, G. W. Fearnley, E. Tate, H. X. Smith, L. N. Soldatova, and R. D. King, "Scientific hypothesis generation by a large language model: Laboratory validation in breast cancer treatment," 2024.
- [92] S. Tong, K. Mao, Z. Huang, Y. Zhao, and K. Peng, "Automating psychological hypothesis generation with ai: when large language models meet causal graph," *Humanities and Social Sciences Communications*, vol. 11, 7 2024.
- [93] I. Ciucă, Y.-S. Ting, S. Kruk, and K. Iyer, "Harnessing the power of adversarial prompting and large language models for robust hypothesis generation in astronomy," 2023.
- [94] A. Bazgir, R. chandra Praneeth Madugula, and Y. Zhang, "Protein hypothesis: A physics-aware chain of multi-agent rag lilm for hypothesis generation in protein science," *Towards Agentic AI for Science: Hypothesis Generation, Comprehension, Quantification, and Validation*, 2025.
- [95] Q. Liu, M. P. Polak, S. Y. Kim, M. A. A. Shuvo, H. S. Deodhar, J. Han, D. Morgan, and H. Oh, "Beyond designer's knowledge: Generating materials design hypotheses via large language models," 2024.
- [96] O. Shir, "Towards ai research agents in the chemical sciences," *ChemRxiv*, 1 2024.
- [97] A. Bazgir, R. chandra Praneeth Madugula, and Y. Zhang, "Agent hypothesis: A survey on hypothesis generation using lilm systems," *Towards Agentic AI for Science: Hypothesis Generation, Comprehension, Quantification, and Validation*, 2025.
- [98] Z. Yang, X. Du, J. Li, J. Zheng, S. Poria, and E. Cambria, "Large language models for automated open-domain scientific hypotheses discovery," 2024.
- [99] S. Yao, D. Yu, J. Zhao, I. Shafran, T. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," in *Advances in Neural Information Processing Systems* (A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, eds.), vol. 36, pp. 11809–11822, Curran Associates, Inc., 2023.
- [100] M. Shamsabadi, J. D'Souza, and S. Auer, "Large language models for scientific information extraction: An empirical study for virology," 2024.
- [101] J. Daggelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson, and A. Jain, "Structured information extraction from scientific text with large language models," *Nature Communications*, vol. 15, 2 2024.
- [102] D. Xu, W. Chen, W. Peng, C. Zhang, T. Xu, X. Zhao, X. Wu, Y. Zheng, Y. Wang, and E. Chen, "Large language models for generative information extraction: A survey," 2024.

- [103] J. Li, M. Zhang, N. Li, D. Weyns, Z. Jin, and K. Tei, “Generative ai for self-adaptive systems: State of the art and research roadmap,” *ACM Transactions on Autonomous and Adaptive Systems*, vol. 19, pp. 1–60, 9 2024.
- [104] Y. Ma, Z. Gou, J. Hao, R. Xu, S. Wang, L. Pan, Y. Yang, Y. Cao, A. Sun, H. Awadalla, and W. Chen, “Sciagent: Tool-augmented language models for scientific reasoning,” 2024.
- [105] “arxiv api.” Accessed: March 10, 2025.
- [106] Neo4J, “Neo4j.” Documentation at <https://neo4j.com> 2024. Accessed:February 24, 2025.
- [107] O. Khattab, A. Singhvi, P. Maheshwari, Z. Zhang, K. Santhanam, S. Vardhamanan, S. Haq, A. Sharma, T. T. Joshi, H. Moazam, H. Miller, M. Zaharia, and C. Potts, “Dspy: Compiling declarative language model calls into self-improving pipelines,” 2023.
- [108] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, “Chain-of-thought prompting elicits reasoning in large language models,” 2023.
- [109] Graph Maker, “Graph maker.” Available at https://github.com/rahulnyk/graph_m 2024. Accessed:February 24, 2025.
- [110] Gradio, “Gradio.” Available at <https://github.com/gradio-app/gradio> 2024. Accessed:February 24, 2025.
- [111] D. Fu, G. Khalighinejad, O. Liu, B. Dhingra, D. Yogatama, R. Jia, and W. Neiswanger, “Isobench: Benchmarking multimodal foundation models on isomorphic representations,” 2024.
- [112] T. Guo, K. Guo, B. Nan, Z. Liang, Z. Guo, N. V. Chawla, O. Wiest, and X. Zhang, “What can large language models do in chemistry? a comprehensive benchmark on eight tasks,” 2023.
- [113] S. Zhu, X. Liu, and G. Khalighinejad, “Chemqa: a multimodal question-and-answering dataset on chemistry reasoning.” <https://huggingface.co/datasets/shangzhu/ChemQA> 2024
- [114] OpenAI, “Gpt-4 technical report,” 2024.
- [115] G. Team, “Gemini: A family of highly capable multimodal models,” 2024.
- [116] “The claude 3 model family: Opus, sonnet, haiku.”