nature reviews materials

https://doi.org/10.1038/s41578-025-00772-8

Perspective

Check for updates

Large language models for reticular chemistry

Zhiling Zheng[1,2,3], Nakul Rampal[1,2,3], Theo Jaffrelot Inizan[1,2,3], Christian Borgs[2,3], Jennifer T. Chayes[2,3,4,5,6] & Omar M. Yaghi[1,2,7]

Abstract

Reticular chemistry is the science of connecting molecular building units into crystalline extended structures such as metal–organic frameworks and covalent organic frameworks. Large language models (LLMs), a type of generative artificial intelligence system, can augment laboratory research in reticular chemistry by helping scientists to extract knowledge from literature, design materials and collect and interpret experimental data — ultimately accelerating scientific discovery. In this Perspective, we explore the concepts and methods used to apply LLMs in research, including prompt engineering, knowledge and tool augmentation and fine-tuning. We discuss how 'chemistry-aware' models can be tailored to specific tasks and integrated into existing practices of reticular chemistry, transforming the traditional 'make, characterize, use' protocol driven by empirical knowledge into a discovery cycle based on finding synthesis–structure–property–performance relationships. Furthermore, we explore how modular LLM agents can be integrated into multi-agent laboratory systems, such as self-driving robotic laboratories, to streamline labour-intensive tasks and collaborate with chemists and how LLMs can lower the barriers to applying generative artificial intelligence and data-driven workflows to such challenging research questions as crystallization. This contribution equips both computational and experimental chemists with the insights necessary to harness LLMs for materials discovery in reticular chemistry and, more broadly, materials science.

Sections

[1]Department of Chemistry, University of California — Berkeley, Berkeley, CA, USA. [2]Bakar Institute of Digital Materials for the Planet, Berkeley, CA, USA. [3]Department of Electrical Engineering and Computer Sciences, University of California — Berkeley, Berkeley, CA, USA. [4]Department of Mathematics, University of California — Berkeley, Berkeley, CA, USA. [5]Department of Statistics, University of California — Berkeley, Berkeley, CA, USA. [6]School of Information, University of California — Berkeley, Berkeley, CA, USA. [7]KACST–UC Berkeley Center of Excellence for Nanomaterials for Clean Energy Applications, King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia. ✉e-mail: yaghi@berkeley.edu

Introduction

Reticular chemistry, the science of linking molecular building units with strong bonds to make crystalline extended structures, offers a vast and versatile playground for material design and discovery[1–4]. The flexibility with which these building blocks can be modified has led to the synthesis and study of thousands of reticular compounds each year (Fig. 1), including metal–organic frameworks (MOFs), covalent organic frameworks (COFs), zeolitic imidazolate frameworks (ZIFs) and molecular weaving. Still, this progress has barely scratched the surface of this nearly infinite design space[2,5]. To accelerate discovery in this field, the limitations of the traditionally used trial-and-error approach, which relies heavily on domain-specific knowledge and lacks scalability of operations, must be overcome.

The appeal and promise of generative artificial intelligence (AI) systems have been growing over the past few years[6–11]. Among them, large language models (LLMs) have captured the attention of the chemistry and material community owing to their unique capabilities in natural language processing, chemical knowledge integration and tool utilization[9,12,13]. Such features have the potential of enabling, for instance, customizable extraction of chemical data from literature[14,15] and adaptable automation of synthesis on robotic platforms[16], which may empower chemists to address societal problems faced today in climate change, clean energy, clean air, clean water and health more efficiently by streamlining routine laboratory tasks[2]. Yet, as a newly emerging field, the practice of identifying and configuring LLM agents for specific downstream tasks as well as optimizing their performance can be vexing for a chemist without prior experience or knowledge about how to 'plug in' LLMs to help improve their existing workflows.

This Perspective aims to address how LLMs can transform the practice of reticular chemistry and, more specifically, how they can improve understanding of the synthesis–structure–property–performance relationships in this field and thereby accelerate the discovery of reticular materials. We begin by introducing a set of basic techniques for leveraging LLMs and then present their potential applications in the laboratory. Our goal is to inspire new methods of using LLMs and to break down any barriers to their adoption in reticular chemistry. Ultimately, we aim to provide concepts, opportunities, challenges and key insights from recent work for those 'LLM-curious' chemists — whether computational or experimental — who are eager to explore the potential of LLMs in their research.

Fundamentals of large language models

LLMs, exemplified by the GPT series[17,18], Claude series, Gemini series[19] and LLaMA series[20,21], are a type of AI system designed to recognize and generate human-like language patterns. These models are deep neural networks with billions of parameters, typically based on the transformer architecture[22] (Fig. 2a). They are trained on a vast corpora of text data, enabling them to learn intricate linguistic patterns, grammar, context and semantics[9,20,23]. Additionally, LLMs can be designed to be multimodal[24,25] (Fig. 2b), meaning they can process various types of data, in particular images and videos, through encoding[26,27], expanding the range of problems to which they can be applied[19,28,29]. Several comprehensive reviews detailing the development of LLMs have been published[30–34]. Here, we focus on methodologies that chemists can use to 'teach' or optimize the behaviour of a base LLM model (Fig. 2c–e and Table 1), aligning it with specific tasks in reticular chemistry.

Prompt engineering

Prompt engineering refers to optimizing the user prompt (input instructions) given to an LLM to consistently produce high-quality responses aligned with the user's goals. This approach is widely used[35–37] and particularly appealing to chemists because it enables 'teaching' LLMs using natural language[13,15,16,32,38,39], which lowers the barrier and the need for extensive coding expertise. In other words, the text-based instructions themselves can serve as a form of programming the LLM. For instance, a prompt such as 'Please summarize the synthesis conditions from the paragraph below into a table' can guide the LLM to identify chemical entities within the provided (con)text and produce structured output, such as a table in this case, without requiring the user to implement code to define chemical terms explicitly (Fig. 2c, top). Moreover, the prompt can be modified in natural language to satisfy the needs of different domains for various synthesis parameters[15].

An important consideration with LLMs is that their primary aim is to generate natural language, which does not necessarily guarantee the production of accurate information[9,28,30]. This can lead to instances in which an LLM produces convincing yet factually incorrect content, a phenomenon known as hallucination[9,40]. For instance, it can generate non-existent citations[14], fabricated synthesis conditions[15] or wrong chemical structures and properties[41], all of which are undesirable outputs. In zero-shot prompting scenarios, in which the model has not been provided task-specific examples in the prompt, our experience indicates that drafting an effective prompt for chemistry-related tasks involves three key principles[15].

Minimize hallucinations. Include a sentence such as 'Please use the provided text to answer the question. If you do not know, answer 'N/A' to ensure the LLM bases its response on the given context rather than generating speculative answers with wrong information.

Provide detailed instructions. Reduce ambiguity by specifying precise parameters, such as 'metal–linker ratio, reaction temperature, reaction time', rather than vague terms such as 'different reaction conditions'. This clarity helps the LLM to interpret the prompt consistently. Additionally, defining the model's role, such as 'You are an expert in organic synthesis', can further align its output with domain-specific tasks.

Request structured output. Specify the desired output format in the prompt to guide the LLM's response structure to ensure more consistent results and facilitate easier post-processing. In addition, an output template can be provided to make sure the LLM can generate answers aligning with human preference.

As demonstrated in the next section and through the prompt templates provided in some of our group's work[15,38,42], the use of these principles when designing simple prompts leads to an improvement in the LLM's performance towards the desired goal. It has also been observed that minor variations in wording, sentence order or even typos in the prompt do not substantially affect the outcome, provided that the three key principles are adhered to.

In few-shot learning scenarios — in which multiple input–output pairs are provided as examples — the appropriate use of examples can be very powerful in an LLM's learning and generation of the desired response[15]. Examples are particularly useful for classification tasks, in which instructions alone may be insufficient. For instance, when prompting the model to recognize whether a given paragraph describes a synthesis, rather than defining the characteristics of a synthesis paragraph in the prompt, the user can provide several examples, both positive (such as sample synthesis paragraphs) and negative (such as characterization preparation, post-synthetic modification and other misleading examples), and indicate which should and should not be classified as a synthesis paragraph. The model is then given a new paragraph to classify based on these examples (Fig. 2c, middle). A similar procedure of providing examples can be applied when extracting information from tables, where ambiguous symbols or terms should be included in the prompt alongside their correct counterparts for accurate extraction. Further discussions on applying in-context learning can be found in the literature[30,36,43,44].

To further enhance the reasoning capabilities of LLMs – enabling them to tackle logical problems, task planning and critical thinking – techniques such as chain-of-thought[35,37,45], tree-of-thought[46], text transformation graphs[47], self-consistency[48] and self-reflection[49,50] have been developed. For example, when generating code for a liquid handler to prepare reaction mixtures, chain-of-thought can be used to break down complex tasks into sequential steps (for example, selecting reagents, determining volumes, setting pipette parameters and specifying well positions), facilitating more accurate execution (Fig. 2c, bottom). Self-consistency, which involves generating multiple answers and selecting the best one, can simulate code before actual experiments, whereas self-reflection, in which the output of the model is fed back into itself, helps to revise plans or correct errors. We note that the application of these techniques can vary depending on the specific task. It is important to recognize that any given LLM may require careful prompting and continuous experimentation to yield optimal outcomes. A practical approach is to start with a simple human-written prompt or an existing reported prompt, interact with the LLM, evaluate its responses and iteratively refine the prompt to include more detailed instructions, examples or advanced prompting methods mentioned earlier to optimize performance and enhance the utility of the model.

Augmentation

External data and toolkits can be integrated with LLMs to unlock more user-defined scientific tasks[16,51–53]. This approach is different from prompt engineering in that it allows LLMs to access external resources and interact more closely with the real world (Table 1). Broadly speaking, there are two main categories of augmentation, data augmentation and tool augmentation, which can be used together or independently to extend the capabilities of the LLM.

Data augmentation enhances the knowledge and contextual accuracy of LLM by integrating supplementary external information sources and is exemplified by retrieval-augmented generation (RAG)[54,55], in which the LLM can access up-to-date reliable information and knowledge and generate more accurate answers, by retrieving data from relevant web pages, scientific literature and databases. This can be done either by automated database lookups using semantic similarity matching or by using a web search module that pulls in real-time data to supplement the responses of LLM[14,51]. For instance, in naive RAG, when a chemist asks, 'What are the synthesis conditions of MOF-321?' – a compound not initially included in the LLM's training – the model might provide hallucinations or no answer. A pre-written function[15] can convert the text string of this question into a vector form and match it with the embeddings of sentences from a literature database, such as 'MOF-321 was prepared using …' or 'Herein, we report the synthesis of MOF-321 in …' based on the highest similarity (usually cosine similarity). These sentences are then combined with the original query and sent to the LLM, enabling it to use the provided reliable information to generate a precise answer. Similarly, when web search engine modules are used, the returned information is passed to the LLM to enhance its knowledge and provide more accurate responses. This approach offers two benefits: it reduces hallucinations and allows dynamic

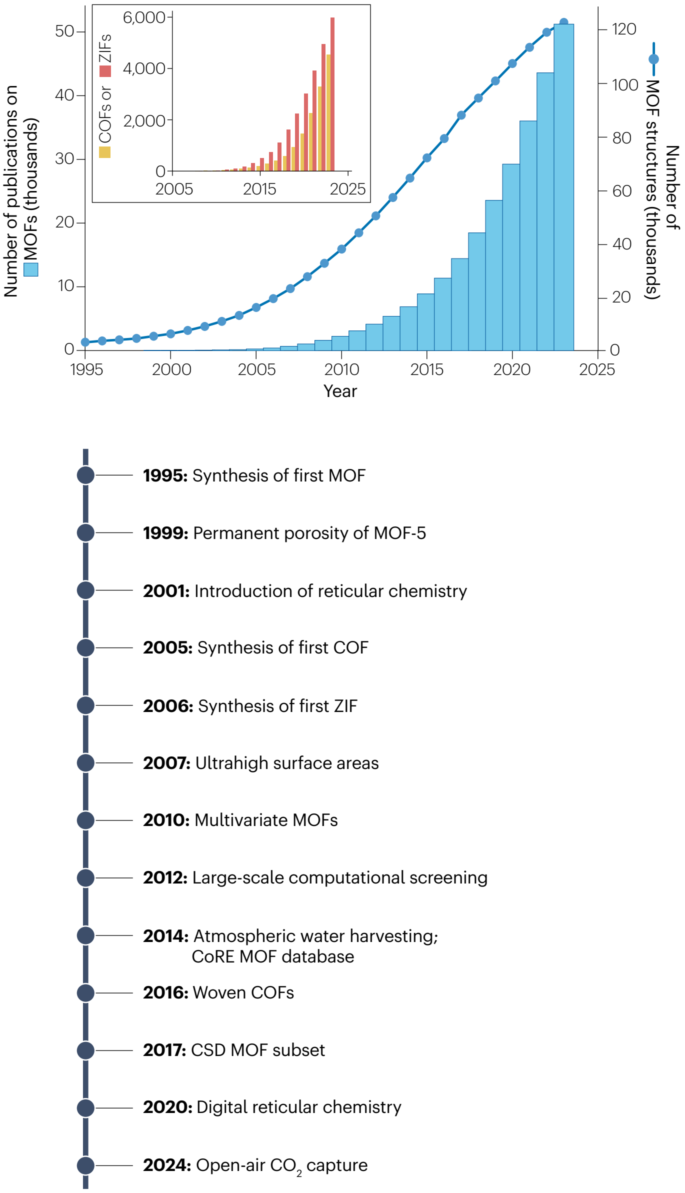


Fig. 1 | Progress in reticular chemistry over the past three decades. The expanding field of metal–organic frameworks (MOFs), covalent organic frameworks (COFs) and zeolitic imidazolate frameworks (ZIFs), illustrated by number of publications and crystal structure depositions, has generated a volume of scientific literature that exceeds the limits of manual exploration, highlighting the need for tools such as large language models to automate processes such as summarization and data extraction. The search on publication data was restricted to the terms[104] MOF, COF and ZIF in original articles and reviews collected on Web of Science, accessed as of 1 August 2024. The number of MOF structures refers to the cumulative yearly totals of crystal structures deposited in the Cambridge Structural Database (CSD) MOF subset[105]. The timeline highlights key milestones, including the synthesis of the first MOF[106], COF[107] and ZIF[108], and developments in their designing principles[1,109–114], synthesis methods[115–118], databases[2,105,119–123] and applications[124–130].

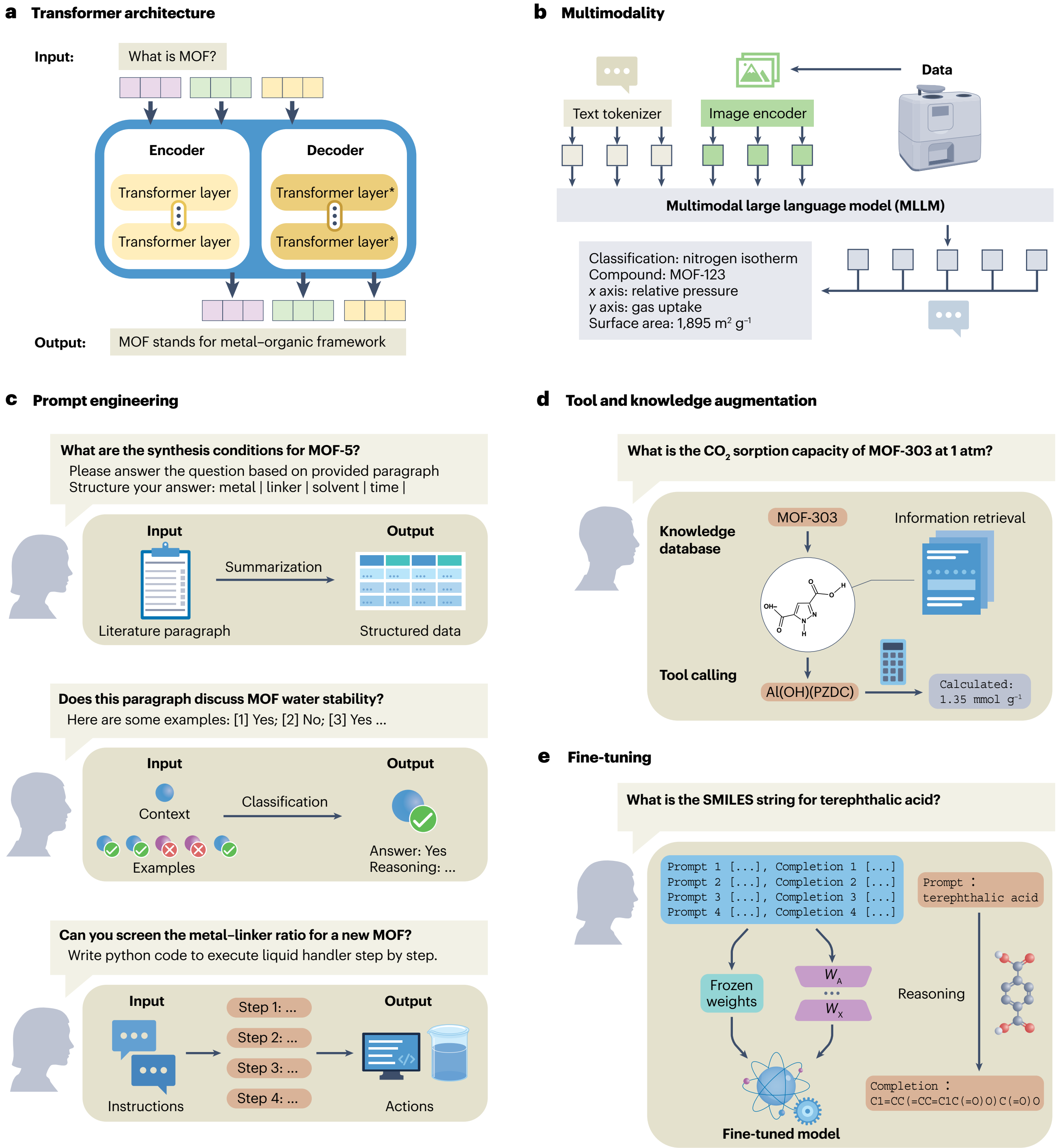

a Transformer architecture
Input:
What is MOF?
Encoder
Transformer layer
Transformer layer
Decoder
Transformer layer*
Transformer layer*
Output:
MOF stands for metal–organic framework
b Multimodality
Data
Text tokenizer
Image encoder
Multimodal large language model (MLLM)
Classification: nitrogen isotherm
Compound: MOF-123
x axis: relative pressure
y axis: gas uptake
Surface area: 1,895 m² g⁻¹
c Prompt engineering
What are the synthesis conditions for MOF-5?
Please answer the question based on provided paragraph
Structure your answer: metal | linker | solvent | time |
Input
Output
Summarization
Literature paragraph
Structured data
Does this paragraph discuss MOF water stability?
Here are some examples: [1] Yes; [2] No; [3] Yes ...
Input
Output
Context
Classification
Examples
Answer: Yes
Reasoning: ...
Can you screen the metal–linker ratio for a new MOF?
Write python code to execute liquid handler step by step.
Input
Output
Step 1: ...
Step 2: ...
Step 3: ...
Step 4: ...
Instructions
Actions
d Tool and knowledge augmentation
What is the CO₂ sorption capacity of MOF-303 at 1 atm?
Knowledge database
MOF-303
Information retrieval
Tool calling
Al(OH)(PZDC)
Calculated: 1.35 mmol g⁻¹
e Fine-tuning
What is the SMILES string for terephthalic acid?
Prompt 1 [...], Completion 1 [...]
Prompt 2 [...], Completion 2 [...]
Prompt 3 [...], Completion 3 [...]
Prompt 4 [...], Completion 4 [...]
Prompt : terephthalic acid
Frozen weights
W_A
W_X
Reasoning
Completion : C1=CC(=CC=C1C(=O)O)C(=O)O
Fine-tuned model

Fig. 2 | Overview of key concepts in leveraging large language models for reticular chemistry. Illustration of the transformer architecture, which can be an encoder-only model (auto-encoding), a decoder-only model (auto-regressive) or a combined encoder–decoder model (sequence-to-sequence) (part **a**). Multimodality refers to the ability of the models to process different types of data, such as text, images, audio and code, often through separate encoders. Note that such design does not guarantee accurate interpretation or precise numerical extraction from complex input data (part **b**). The commonly used techniques to modify the behaviour of large language models are prompt engineering (part **c**), which tailors model outputs for tasks such as summarizing synthesis parameters into structured formats (top), classifying paragraphs using in-context learning (middle) or implementing code for laboratory experiments using chain-of-thought reasoning (bottom); external data and tool augmentation (part **d**), which expands large language model capabilities by accessing external databases for real-time knowledge or integrating with tools to perform tasks such as molecular property calculations; and fine-tuning (part **e**), which retrains models on domain-specific data to specialize in downstream tasks. These techniques can be used individually or in combination, depending on the specific task in chemistry research. MOF, metal–organic framework; PZDC, 1H-pyrazole-3,5-dicarboxylate; SMILES, Simplified Molecular Input Line Entry System.

access to frequently updated external data and knowledge. Note that in cases in which external data sources are unreliable or inaccurate, data augmentation may not necessarily become better than zero-shot or few-shot strategies. Additionally, one should be aware that external data retrieval can sizably increase the total number of tokens for each query, which as the interaction progresses may result in increased model runtime; higher costs, particularly when using API (Application Programming Interface) calls; and lower performance. For example, model performance tends to be highest when the relevant information is located at the beginning or end of the input prompt, whereas it substantially declines when the model needs to retrieve details embedded in the middle[56]. To ensure reliable performance while managing efficient resource consumption, careful optimization of augmentation workflows and thorough data curation (for example, removing irrelevant information and unusual symbols) are necessary.

In contrast to data augmentation, tool augmentation empowers LLMs to perform specialized tasks or calculations by integrating with external software or toolkits. Providing LLMs with external toolkits allows them not only to bypass their inherent weakness in numerical operations or predictions requiring more mathematical rigour, but also to interface with many programmes and become the meta-tool in many automated workflows[16,42,51,52,57]. There are various methods, with differing levels of coding complexity, to equip LLMs with tools or arrays of tools. Common methods include running generated code or making API calls. Typically, the LLM needs to be informed about the available tools, and coding is required to detect when the LLM decides to use a tool, execute it and return the results. For instance, when tasked with calculating the molecular weight of a compound or predicting its property, the LLM can generate a JSON (JavaScript Object Notation) string containing the molecular formulas, which can then be input into an existing function (tool) in the code, with the function's outcome fed back to the LLM to give the answer (Fig. 2d). Other examples of tools include converting molecule names to SMILES (Simplified Molecular Input Line Entry System) strings, converting units, calculating properties from structural files, processing literature images, preparing procedures, executing tasks on robotic platforms, checking inventory availability, capturing images with a microscope camera, analysing raw characterization data and so on. Compared with rule-based coding, the major advantage of tool augmentation lies in the LLM's ability to understand what tools exist, when to use them and how to configure the inputs to correctly execute these models through reasoning[16,42,51].

Fine-tuning

Fine-tuning an LLM involves retraining a base model on a specific data set (for example, scientific papers, synthesis condition data sets, chemical representations, structural information, material design principles and material properties) to tailor its performance to particular tasks or domains[41,58–63]. Unlike prompt engineering and augmentation, which do not alter the base model, fine-tuning adjusts the model's weight parameters and results in a new instance of model (Fig. 2e). This process usually requires a high-quality, domain-specific data set to ensure that relevant information will be demonstrated to the LLM during training, and it statistically improves the LLM's performance in generating accurate and contextually appropriate responses (Table 1). As a result, key advantages of fine-tuning over the previous two methods are that it incorporates more data and examples into the model than can be accommodated in a single prompt, and that it enables the model to learn from the data, rather than just accessing it.

Models that can be fine-tuned include LLaMA[20,21], Mistral[64], SciBERT[65], BART[66], T5 (ref. 67), GPT-3.5-turbo, GPT-4o-mini and Claude 3.5, among others, and common fine-tuning methods include full fine-tuning, instruction tuning and parameter-efficient fine-tuning[68,69]. It should be noted that not all LLMs have open-sourced their weights; for models such as the GPT series and the Claude series, fine-tuning is only possible via company APIs. In fact, for beginners, we recommend starting with LLMs that can be fine-tuned with API support from companies such as OpenAI, Anthropic, Amazon and Databricks, as using these models requires less coding expertise and offloads the computational resources to the service provider, simplifying the process to make it user-friendly.

Consider instruction tuning with API support as an example, which trains LLMs using examples that demonstrate the desired responses to queries. To address the observation that base LLMs might struggle with understanding SMILES[70] strings or abbreviations of organic linkers, additional chemical representation data were incorporated into a model to help it learn better (Fig. 2e). In such a case, a data set of typically hundreds or thousands of pairs of examples that show the syntactically and semantically correct 'translation' from the original name to a specific representation can be developed to train the model on new knowledge[41]. This data set allows the model to 'think' in a new niche way and effectively perform the given task. By contrast, using a few-shot prompt with hundreds of examples in a single long prompt is usually ineffective, and RAG does not teach any patterns either.

The next step is to format these pairs in the data set into queries and expected answers and upload them via the API service. The subsequent fine-tuning process can take minutes to days, depending on the size of the data. An advantage of using API services is that there are no computational or hardware requirements on the user's end, as the service provider runs fine-tuning tasks on its end and charges for a cost. The user can evaluate the fine-tuned model through the provided API by checking its performance with a few organic linker names not included in the training data set. Once fine-tuning is complete, the model is made available via a dedicated API end point to be accessed. It is important to note that after fine-tuning, the resulting model is often best used for the specialized downstream tasks it was fine-tuned for, as it may lose some of its general capabilities. For each different task, a tailored data set often needs to be developed – highlighting a limitation of this method. In many situations, chemistry-related tasks (such as organic linker design, property prediction and synthesis planning) frequently require

Table 1 | Comparison of strategies for optimizing the performance of large language models for chemists

Method	Prompt engineering	Augmentation	Fine-tuning
Implementation effort	Low	Medium	High
Learning mechanism	In-context learning	Tool utilization	Change model weights and behaviours
External data requirement	None or low	Required	Extensive
Coding complexity	None or low	Medium to high	Medium
Use cases	Tasks on summarization, reasoning	Enhanced factual knowledge, tool integration	Domain-specific tasks, specialized models

highly specific data and customization to ensure accurate performance, making fine-tuning desirable yet resource-intensive. This trade-off underscores the importance of carefully selecting the fine-tuning strategy to maximize utility while managing potential constraints.

Practical scenarios and workflows

After one knows how to develop domain-specific or task-specific LLMs, the next step is to apply these concepts in reticular chemistry research. The following case studies provide contextualized insight from our group's experience as well as relevant work from other researchers in the field, spanning the spectrum of data mining, material design and optimization.

Data mining

Both experimental and computational chemists in reticular chemistry need to extract knowledge and insights from complex and diverse chemical texts, particularly to understand synthesis–structure–property relationships[2,71]. Traditionally, curating data from scientific literature in this field (Fig. 3) requires intensive human annotation or coding expertise – including rule-based systems and machine-learning (ML) models – to extract information sparsely located within the text, such as synthesis protocols[72,73], porosity[74,75], topology[74], decomposition temperature[76,77] and water stability[78,79]. It is also common that workflows developed for one specific type of data are not generalizable to others.

LLMs offer a scalable way to accelerate this process by automating literature selection, semantic analysis, named entity recognition and post-processing steps (Fig. 3). Each step can be 'programmed' using natural language prompt engineering in various chemical vocabularies, making this approach particularly user-friendly and accessible for reticular chemists[15,71,80,81]. In addition to using only prompt engineering, research over the past year has demonstrated that the fine-tuning strategy can be used to enhance an LLM's performance accuracy in data mining tasks[62,63].

To illustrate, consider the task of extracting MOF synthesis parameters. To define the research scope (step 1), a list of synthesis conditions that lead to MOF crystallization is conceived, which includes variables

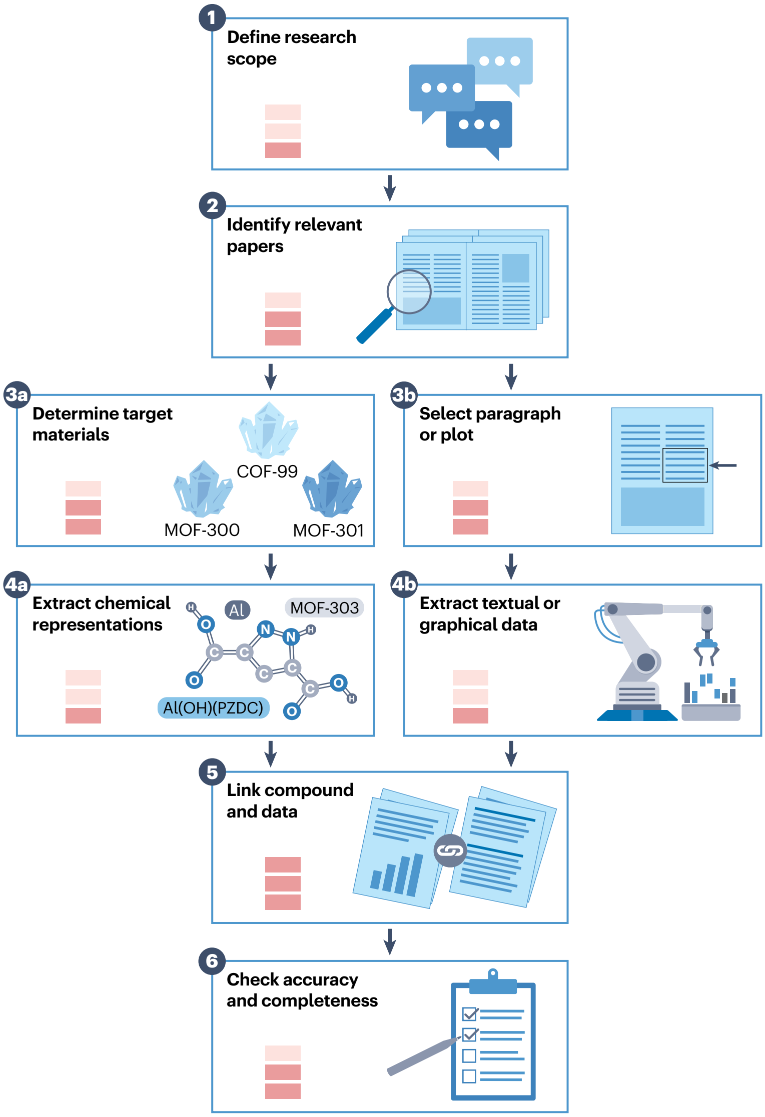
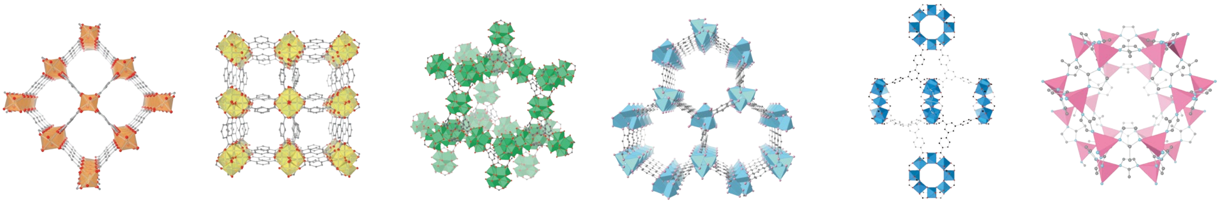


Fig. 3 | Overview of key steps in data mining from scientific literature. Although steps such as defining data types to be extracted (step 1) and data extraction itself (steps 4a and 4b) are straightforward, selecting relevant papers to read (step 2) and identifying specific targets for analysis (steps 3a and 3b) are more complicated. The most challenging step (step 5) involves linking trivial names or labels of compounds with their synthesis conditions or characterization data found elsewhere in the paper. The final step is to verify the accuracy and completeness of the data extracted (step 6). The number of red bars at each step indicates the relative difficulty. COF, covalent organic framework; MOF, metal–organic framework; PZDC, 1H-pyrazole-3,5-dicarboxylate.



Compound name	Al-fum	CAU-10	MOF-808	MOF-74	MOF-520	ZIF-8
Metal	$AlCl_3·6H_2O$	$Al_2(SO_4)_3·18H_2O$	$ZrOCl_2·8H_2O$	$Zn(NO_3)_2·4H_2O$	$Al(NO_3)_3·9H_2O$	$Zn(NO_3)_2·4H_2O$
Linker	Fumaric acid	Isophthalic acid	Trimesic acid	2,5-Dihydroxyterephthalic acid	Benzenetribenzoic acid	2-Methylimidazole
Solvent	H_2O	DMF, H_2O	DMF	DMF	DMF	DMF
Temperature and time	100 °C, 6 h	135 °C, 12 h	100 °C, 168 h	105 °C, 20 h	140 °C, 96 h	140 °C, 24 h
Observation	White precipitate	White precipitate	Octahedral colourless crystals	Yellow needle crystals	Block-shaped clear crystals	Polyhedral crystals

Fig. 4 | Examples of MOF synthesis parameters extracted from literature. Large language models have demonstrated their chemical knowledge and ability to recognize domain-specific name entities, enabling the summarization of diverse synthesis data into user-preferred formats. The obtained insights and knowledge in the large language model-mined data set can be later converted into actionable outcomes such as synthesis prediction or a knowledge graph. DMF, *N*,*N*-dimethylmethanamide; MOF, metal–organic framework; ZIF, zeolitic imidazolate framework.

such as metal, linker, solvent, reaction time and temperature. The workflow begins with the selection of relevant papers using an LLM (step 2). The model is taught to recognize user-defined criteria (for example, 'MOF synthesis', 'experimental', 'not post-synthetic modification', 'not a review paper') by examining the titles and abstracts in a library of papers. Once relevant papers have been identified, each paper is fed into the model to extract specific parameters needed by the user (steps 3 and 4). The desired output format (for example, table, JSON dictionary, categorical labels) is specified in the prompt. Some examples of output synthesis parameters are shown in Fig. 4. In cases in which data may be in different sections of a paper (such as abbreviations, general procedures, reference codes), a holistic approach is necessary to synchronize such information (step 5). Finally, the performance of the LLM is evaluated by comparing the output with the ground truth (step 6).

Notably, the flexibility of the query language allows this process to be easily adapted to diverse research needs and lowers the barrier for a reticular chemist with less experience in data science to conduct data mining; for example, by replacing 'synthesis parameters' with 'BET surface area' (where BET refers to the Brunauer–Emmett–Teller method) in the prompt, the focus shifts from synthesis conditions to porosity data extraction.

Following data mining, the data must be processed. LLMs have also demonstrated their utility in this regard, streamlining labour-intensive processes so that human creativity can be used on other aspects. For instance, these models can convert abbreviations to conventional names or canonical SMILES codes, apply helper functions for calculating molarity or concentration, identify outliers in synthesis data, correct formatting errors or observe variable interactions and correlations and so on[15]. Although tagging and associating different names or abbreviations within the same paper has been demonstrated[15,80], standardizing data across papers remain challenging owing to the various naming conventions used in the literature (for example, MOF-74 versus CPO-27, H_2PDC versus H_2PyDC). We envision that such challenges might be alleviated by using LLMs with a stronger reasoning ability and a more comprehensive knowledge base specific to reticular materials and their building blocks. Alternatively, it might be possible in the future to assemble a group of LLMs (multi-agent LLM systems)[42,80,82] or leverage fine-tuned LLMs[63] to streamline the standardization of raw data extracted from literature during the text-mining process.

In addition to using LLMs to extract synthesis information and properties from papers converted to plain text, multimodal LLMs can process images as input and classify plots or extract associated information[29,83]. For instance, an LLM with vision capability[28] can be prompted to classify whether a thermogravimetric analysis graph or nitrogen sorption isotherm plot exists on a given page[29] (Fig. 2b). The input in this case is a human-written customized prompt and an attached JPEG file of a full-page view from a paper. Once classification is completed, irrelevant pages can be eliminated, and the remaining ones can be further processed to extract information such as compound names in the figure, measurement parameters and reported property values. It should be acknowledged that although current models are able to recognize categorized plot types and read accompanying figure text boxes, annotations and captions[29,83], extracting numerical values from data points in figures remains challenging (for example, reading water uptake at a given pressure), even with models that can classify and interpret these images. This difficulty arises because the model must discern the underlying scaling and axis values that the plotted data represent – where points lie on the axes, what those axes represent and how to translate a position on the plot into a number – rather than simply reading text and interpreting the contents.

Overall, the use of LLMs in data mining can greatly reduce human labour and enhance access to useful information. Our experience indicates that a practical approach to developing this entire data mining process is to break it down into steps, start from what a human chemist would do in each step and 'teach' the LLM through prompts by treating it as an apprentice. Once a preliminary workflow has been established, it can be tested on a few papers to determine whether further instructions should be added to the prompt. After achieving the results in a consistent format on smaller data sets with low uncertainty over the goal, the workflow can be applied repetitively to publication corpuses.

Designing reticular frameworks

Over the past two decades, much attention has been paid to the rational design of reticular frameworks. Given the virtually infinite ways to engineer frameworks at the atomic level, it remains a formidable challenge for humans to enumerate all structural possibilities and conduct experiments. LLMs, as a class of generative AI, are able to streamline this process: these models can be trained to generate a library of new building blocks or even entire structures, whose synthetic feasibility and properties can then be evaluated to narrow down the list to the most promising candidates, thereby accelerating materials discovery and synthesis.

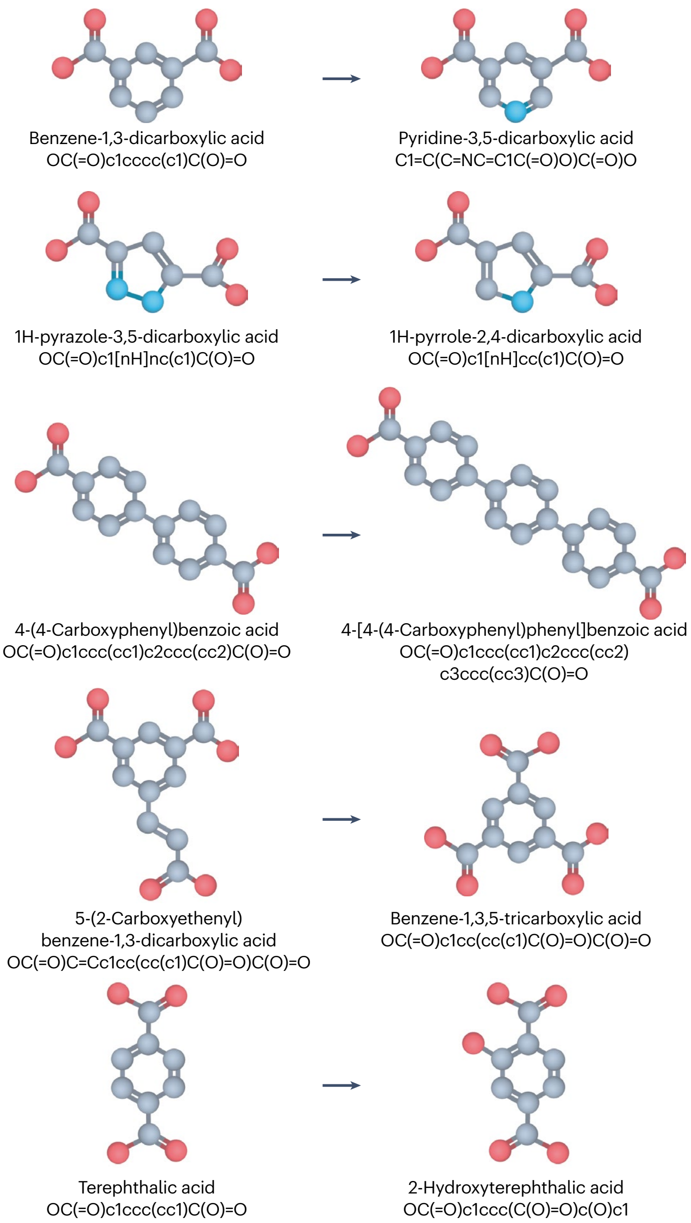

Benzene-1,3-dicarboxylic acid
OC(=O)c1cccc(c1)C(O)=O
Pyridine-3,5-dicarboxylic acid
C1=C(C=NC=C1C(=O)O)C(=O)O
1H-pyrazole-3,5-dicarboxylic acid
OC(=O)c1[nH]nc(c1)C(O)=O
1H-pyrrole-2,4-dicarboxylic acid
OC(=O)c1[nH]cc(c1)C(O)=O
4-(4-Carboxyphenyl)benzoic acid
OC(=O)c1ccc(cc1)c2ccc(cc2)C(O)=O
4-[4-(4-Carboxyphenyl)phenyl]benzoic acid
OC(=O)c1ccc(cc1)c2ccc(cc2)
c3ccc(cc3)C(O)=O
5-(2-Carboxyethenyl)
benzene-1,3-dicarboxylic acid
OC(=O)C=Cc1cc(cc(c1)C(O)=O)C(O)=O
Benzene-1,3,5-tricarboxylic acid
OC(=O)c1cc(cc(c1)C(O)=O)C(O)=O
Terephthalic acid
OC(=O)c1ccc(cc1)C(O)=O
2-Hydroxyterephthalic acid
OC(=O)c1ccc(C(O)=O)c(O)c1

Fig. 5 | Generating molecular building block structures using fine-tuned large language models. The base large language models were trained on data sets of SMILES (Simplified Molecular Input Line Entry System) strings and IUPAC names of metal–organic framework linkers, respectively. The resulting fine-tuned models demonstrated an understanding of molecular structure and were capable of generating syntactically and chemically valid structures for new linker designs. Various molecular editing techniques, including functionalization, insertion and heteroatom introduction, were applied to generate new molecules based on given primitive molecules. Colour code: C, grey; O, red; N, blue. Hydrogen atoms are omitted for clarity.

Rather than building a generative AI model from scratch, which is resource-intensive, a reticular chemist can leverage pre-trained LLMs and fine-tune them on specific tasks. The key to this process is to obtain a suitable data set that encompasses new knowledge necessary for fine-tuning. For instance, in 2023, we demonstrated the use of LLMs to mine data on MOF synthesis parameters[15], which included hundreds of organic linkers. Building upon this, we explored whether this MOF linker data set could assist an LLM with learning how to edit organic linkers – such as functional group modifications, molecular length variations and heteroatom incorporation – and generate new and valid molecules[41,84] (Fig. 5). In other words, for a given 'mother' linker (A_0), we were interested in training an LLM to generate a series of variants (A_1, A_2, A_3 and so on) in its output. This task required the LLM to understand the syntax of IUPAC names or SMILES strings and to ensure that generated molecular structures were chemically valid (that is, not violating bonding rules).

To this end, the data set included 3,943 example pairs (from A_0 to A_1, A_0 to A_2, B_0 to B_1 and so on) that were used for instructional tuning to guide the GPT-3.5-Turbo in learning the chemical information and rules embedded within it. The resulting fine-tuned model could generate SMILES strings or IUPAC names for edited MOF linkers with a higher accuracy (>84.8%) compared with the base model (10.2%), and this process was iterated many times, making multiple modifications, to create a library of MOF linkers. Promising candidates could be manually identified or computationally selected based on their commercial availability or synthetic feasibility determined by other ML models, leading to successful discovery of new water-harvesting MOFs[41]. More recently, the ability of LLMs to generate crystal structure CIFs (Crystallographic Information Files) through fine-tuning has been demonstrated[61], suggesting further avenues for using LLMs as generative models in high-throughput computational screening.

LLMs also offer possible solutions to automating property prediction, aiding the design of reticular materials. Although LLMs cannot be directly trained to predict properties owing to their limitations in numerical operations[85], they can be augmented with predictive tools or computational packages to enhance mathematical rigour[52]. In this case, LLMs could help to select and interact with the proper computational package for reticular chemists. For example, we envision that in the future, calculation tools that determine accessible surface areas, such as Zeo++[86–88], PoreBlazer[89,90] and RASPA[91], could potentially be integrated with LLMs in the workflow. The LLM could recognize input files in the relevant formats (for example, XYZ and CIF), reason about the necessary parameters, execute the tasks and interpret the outcomes. Although current LLMs may struggle with such complex integrations, advancements in this area could enable this capability. Ultimately, this process can be repeated for thousands of structures with LLM-in-the-loop. Moreover, tools such as trained ML models that assess water stability, thermal stability and CO_2 uptake under specific conditions can also be introduced for function calling. In this setup, the LLM acts as a decision-maker, selecting the appropriate tool based on the user query and assigning the correct inputs to ensure that the tool executes accurately and analyses the results. In essence, a suite of individual LLMs could be developed to collect data sets, automate the generation of new materials and predict their properties, working sequentially to accelerate discovery in the field of reticular chemistry (Fig. 6).

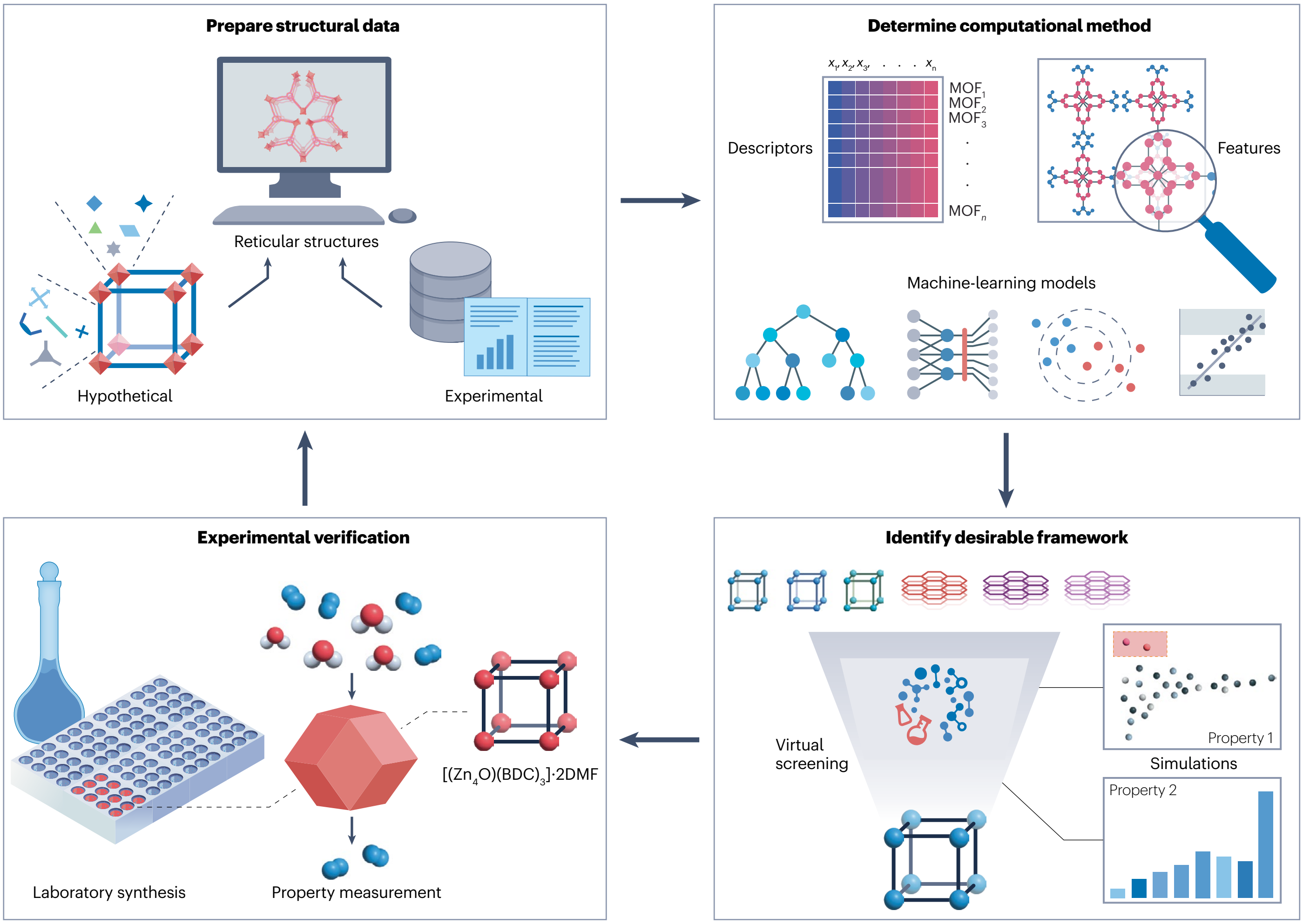


Fig. 6 | The roles LLMs can take in a data-driven selection process for reticular frameworks. The process begins with the large language model (LLM)-assisted preparation of structural data, encompassing both hypothetical and experimental metal–organic framework (MOF), covalent organic framework or zeolitic imidazolate framework structures. Next, LLMs can help to determine critical descriptors and use predictive models to evaluate the performance of these framework structures. This is followed by using computational analysis as a tool to identify promising frameworks with simulations and algorithms (such as decision trees, support vector machines, neural networks and so on). The final step involves the experimental verification of the computationally selected MOFs, which can be accelerated with an LLM-driven laboratory, to confirm their predicted properties and applications. BDC, 1,4-benzenedicarboxylate; DMF, *N*,*N*-dimethylmethanamide.

Synthesis exploration and automation

It is a dream and critical goal for every synthetic chemist working in reticular chemistry to obtain single crystals of the MOF, COF or ZIF they are studying. Not only does the attainment of crystals enable unambiguous structural characterization through X-ray and electron diffraction techniques but it is also a hallmark of synthesis quality, suggesting that the synthesis conditions are optimal. Nevertheless, crystallization persists as a challenge, requiring time and experience for chemists to master. Unfortunately, such empirical knowledge is often non-transferable between similar isoreticular compounds, as the optimal crystallization conditions for a given structure might shift with even the slightest alteration in the composition of a building block (such as from -H to -CH_3). Furthermore, it is difficult to keep up with the growing number of published papers and synthesis procedures, so one can easily miss information that could save months or even years of work.

To help reticular chemists more effectively crystallize their target framework structures, we propose leveraging LLMs to enhance reticular material synthesis in three critical ways: by providing laboratory guidance, predicting synthesis outcomes and integrating automation for synthesis optimization. Although LLMs provide guidance on laboratory activities, plan syntheses and generate hypotheses, humans can verify these outputs experimentally and report back to the LLM to generate guidance on the next steps. This iterative process was shown to discover a series of isoreticular aluminium MOFs through multiple rounds of human–AI collaboration[38]. In the prompt, a memory storage section was introduced that asks the LLM to summarize the progress in the collaboration so far and add the latest activity. This historical record accumulates over time, documenting the success and failures over iterations and enabling the LLM to avoid repeating suggestions and dynamically adjust its guidance for the next steps.

Glossary

API
(Application Programming Interface). A set of rules and protocols that allow different software applications to communicate and share data or commands.

CIFs
(Crystallographic Information Files). A standardized text file format that records crystal structure data, including atomic coordinates and unit cell parameters, enabling consistent sharing of crystal structures.

JSON
(JavaScript Object Notation). A lightweight, text-based format used to structure, store and transfer data between systems in a human-readable manner.

LLM-in-the-loop
A workflow in which a large language model (LLM) continuously participates and provides input, just as a human expert would in a 'human-in-the-loop' scenario. The agent may propose actions, analyse data or suggest refinements, and then adapt its guidance based on feedback from experimental results, computational tools or human researchers.

Neural networks
A computational model inspired by the structure of the human brain, composed of layers of interconnected nodes (neurons) that process and learn patterns from data.

SMILES
(Simplified Molecular Input Line Entry System). A textual notation for representing chemical structures, allowing for easy storage, manipulation and computational handling of molecular information.

Tokens
The smallest units of text (such as words, parts of words or symbols) that a language model processes and generates during text analysis and processing.

We have found that this type of human–AI collaboration, leveraging the domain knowledge of the LLM in reticular chemistry, helps a human with no prior experience in material discovery to effectively navigate the literature search, organic linker synthesis, MOF screening process and measurement of properties. The LLM provides relevant suggestions, makes reasonable hypotheses based on observations and offers logical trials, akin to a helpful and experienced co-worker readily available in the laboratory.

Giving experimentalists the ability to predict whether given synthesis parameters (linker, solvent, reaction time, temperature and so on) can form reticular frameworks would allow them to identify key parameters to tune and reduce the screening burden. The coding and programming capabilities of LLM agents are particularly interesting for this purpose, as these agents can create and use tools (such as ML programs or a bespoke data post-processing script) that aid such prediction. GPT-3.5 was used in a 2023 study to mine synthesis conditions, which were then fed back to a more advanced GPT-4 to develop an ML model to predict single-crystal formation[15]. This binary classification model achieved more than 90% accuracy. Although tool-using features were not implemented at the time, this ML model can now be integrated with LLMs to select input synthesis parameters and return prediction outcomes. As an alternative approach, LLMs themselves can be fine-tuned to predict the formation of inorganic compounds based on a given set of precursors[62]. It is envisioned that with an appropriate synthesis data set in reticular chemistry (MOF, COF and ZIFs), a specialized LLM could predict synthesis outcomes. Collectively, these examples demonstrate that, through augmentation or fine-tuning, LLMs can assist in the prediction of synthesis conditions of reticular materials.

Manual experimentation in reticular chemistry is usually labour-intensive and can be slow owing to extensive screening of crystallization conditions. To accelerate the synthesis step in this field, LLMs can be integrated with real laboratory environments, interfacing seamlessly with human researchers, digital systems and physical hardware[42,51,53,92–94]. This integration can bridge the gap for those unfamiliar with digital tools or automation platforms, enhancing productivity by utilizing LLMs as meta-tools. Our group has developed task-specific LLMs[15,38] that collaboratively interacted to discover and optimize the synthesis conditions of two novel MOFs and a new COF. Each LLM is modularized and orchestrated within a multi-agent system, in which distinct models handle specific tasks and they can talk to each other[42]. For instance, one LLM could handle planning and guidance, another could focus on data mining in reticular chemistry literature, while others could manage document search and data analysis, ML for suggesting and optimizing synthesis conditions, robotic platform operation for high-throughput synthesis based on ML-guided synthesis conditions and laboratory safety. This kind of integration edges closer to the vision of an AI-powered smart laboratory[16,51,57] and gives a single human chemist the productivity of an entire team — speeding up the optimization of crystallinity from the typical months to a few days. Many synthesis steps can be modularized and assigned to LLMs to unlock a complete experiment–computation–ML loop and streamline routine tasks.

Outlook

In this Perspective, we have explored how LLMs can improve reticular chemists' understanding and practice of material design, synthesis and discovery, and we have outlined the conceptual and methodological blueprint for applying LLMs in reticular chemistry research. This blueprint remains valid for many other fields in chemistry and materials science as well. Despite these advances, ongoing efforts are needed to enhance generative models further, making intelligent predictions in chemistry more routine and reliable.

First, there is room for improvement in the quality and breadth of data accessible to LLMs. Successful data points are frequently reported in the literature, but the inclusion of failures is equally important. Additionally, the reliance on simulated or computationally generated artificial data, although more accessible and less expensive than experimental data, may introduce uncertainties about the reliability of the output given by LLMs. Despite preliminary efforts having shown promise in using LLMs to extract structured data and subsequently use the mined data set to train a more specialized LLM with better performance[41,63], structured and high-quality data sets are currently scarce in reticular chemistry because most of the data are distributed across different sources or repositories[2], and making sure they are accurate is another challenge.

Moreover, the development of benchmark data sets for evaluating and comparing LLM performance is still in its early stages. Although platforms such as Chatbot Arena[95] provide benchmarks for general applications, very few equivalent benchmarks exist for scientific tasks. Such benchmarks are crucial for systematically advancing the field, as they allow for various models to be assessed on specific tasks in reticular chemistry such as question-answering[14,81,96], synthesis condition extraction[82,97] and property prediction[98]. Additionally, as scientific literature is continually evolving, LLMs may not be able to judge the reliability of certain information without additional context or updates, particularly when newer studies present conclusions that conflict with earlier reports. Thus, the development and curation of comprehensive data sets to train and evaluate LLMs — encompassing chemical structures, synthesis conditions, characterization results and properties such as porosity, gas uptake and stability — demand concerted efforts from the scientific community.

The seamless integration of LLMs within existing laboratory experimentation infrastructure and computational tools presents both challenges and opportunities. Although LLMs are adept at making reasoned decisions that surpass rule-based systems and can utilize various tools, these tools typically require initial human development with detailed documentation and manual integration with LLMs through coding, prompt engineering or both. In this context, developing robust interfaces and APIs to facilitate interactions between LLMs and digital systems in terms of perception and detection of the real-world activities is essential.

For experimentalists with less computational experience, the ability of the LLM to implement code can democratize access to ML tools and AI technologies, enabling more scientists to leverage data-driven approaches tailored to their research projects without needing deep programming expertise. Furthermore, exploring different ways to allow LLMs to write their own code to obtain new data and tools when needed, as well as developing multimodal data integration for the LLMs to learn cross-connections, can enhance their utility and reduce the reliance on pre-existing human-coded tools. This capability would enable a more dynamic and responsive laboratory environment in which LLMs can continuously adapt and optimize their performance.

It should also be acknowledged that the deployment and utilization of LLMs come with substantial computational and energy costs[99,100]. Training and operating these large models require considerable resources and consume a sizable amount of energy, costing money and impacting the environment[101,102]. Thus, there is no 'free lunch' when adopting LLMs; the benefits come with the responsibility to consider sustainability and efficiency in their use.

Importantly, just as raising children requires patience and guidance, one should not expect superhuman productivity from LLMs from the outset. Instead, these models should be soberly viewed as helpful assistants or agents for sophisticated data mining tasks, material design and laboratory synthesis, which serve to simplify and expedite workflows that would otherwise require manual human labour and domain expertise[9,103]. By lowering the barriers to access and application, LLMs will enable a broader range of individuals to engage in reticular chemistry research and innovation across diverse academic and industrial settings. Over time, as the demand for LLM-driven approaches increases, so too will the need for more high-quality data sets and more convenient and versatile LLMs, regardless of open-sourced or closed-sourced. As less computationally intensive LLMs are developed to ensure sustainable and responsible use of these powerful tools, the integration of various AI-assisted tools into routine reticular chemistry research practices will likely accelerate. We envision that the field's transformation from a largely empirical science of synthesis to a data-driven science will allow more sophisticated challenges to be solved and more important discoveries to be made.

Published online: 01 February 2025

References

1. Yaghi, O. M. et al. Reticular synthesis and the design of new materials. *Nature* **423**, 705–714 (2003).
2. Lyu, H., Ji, Z., Wuttke, S. & Yaghi, O. M. Digital reticular chemistry. *Chem* **6**, 2219–2241 (2020).
3. Moosavi, S. M. et al. Understanding the diversity of the metal–organic framework ecosystem. *Nat. Commun.* **11**, 4068 (2020).
4. Jablonka, K. M., Rosen, A. S., Krishnapriyan, A. S. & Smit, B. An ecosystem for digital reticular chemistry. *ACS Cent. Sci.* **9**, 563–581 (2023).
5. Yaghi, O. M. & Zheng, Z. Reticular chemistry and new materials. In *26th Int. Solvay Conf. Chem. Chem. Chall. 21st Century* (eds Wüthrich, K., Feringa, B. L., Rongy, L. & De Wit, A.) 155–160 (World Scientific, 2024).
6. Wang, H. et al. Scientific discovery in the age of artificial intelligence. *Nature* **620**, 47–60 (2023).
7. Gupta, P., Ding, B., Guan, C. & Ding, D. Generative AI: a systematic review using topic modelling techniques. *Data Inf. Manag.* **8**, 100066 (2024).
8. Bandi, A., Adapa, P. V. S. R. & Kuchi, Y. E. V. P. K. The power of generative AI: a review of requirements, models, input–output formats, evaluation metrics, and challenges. *Future Internet* **15**, 260 (2023).
9. Bubeck, S. et al. Sparks of artificial general intelligence: early experiments with GPT-4. Preprint at https://arxiv.org/abs/2303.12712 (2023).
10. Walters, W. P. & Murcko, M. Assessing the impact of generative AI on medicinal chemistry. *Nat. Biotechnol.* **38**, 143–145 (2020).
11. Ren, Z., Ren, Z., Zhang, Z., Buonassisi, T. & Li, J. Autonomous experiments using active learning and AI. *Nat. Rev. Mater.* **8**, 563–564 (2023).
12. Microsoft Research AI4Science & Microsoft Azure Quantum. The impact of large language models on scientific discovery: a preliminary study using GPT-4. Preprint at https://arxiv.org/abs/2311.07361 (2023).
13. White, A. D. The future of chemistry is language. *Nat. Rev. Chem.* **7**, 457–458 (2023).
14. Lála, J. et al. PaperQA: retrieval-augmented generative agent for scientific research. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2023).
15. Zheng, Z., Zhang, O., Borgs, C., Chayes, J. T. & Yaghi, O. M. ChatGPT chemistry assistant for text mining and the prediction of MOF synthesis. *J. Am. Chem. Soc.* **145**, 18048–18062 (2023).
16. Bran, A. M. et al. Augmenting large language models with chemistry tools. *Nat. Mach. Intell.* **6**, 525–535 (2024).
17. OpenAI et al. GPT-4 technical report. Preprint at https://arxiv.org/abs/2303.08774 (2023).
18. Ouyang, L. et al. Training language models to follow instructions with human feedback. In *36th Conf. Neural Inform. Process. Syst.* (Morgan Kaufmann, 2022).
19. Gemini Team et al. Gemini: a family of highly capable multimodal models. Preprint at https://arxiv.org/abs/2312.11805 (2023).
20. Touvron, H. et al. LLaMA: open and efficient foundation language models. Preprint at https://arxiv.org/abs/2302.13971 (2023).
21. Touvron, H. et al. LLaMA 2: open foundation and fine-tuned chat models. Preprint at https://arxiv.org/abs/2307.09288 (2023).
22. Vaswani, A. et al. Attention is all you need. In *31st Conf. Neural Inform. Process. Syst.* (Curran Associates, 2017).
23. Wei, J. et al. Emergent abilities of large language models. *Trans. Mach. Learn. Res.* https://openreview.net/forum?id=yzkSU5zdwD (2022).
24. Xu, P., Zhu, X. & Clifton, D. A. Multimodal learning with transformers: a survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 12113–12132 (2023).
25. Zhang, D. et al. MM-LLMs: recent advances in multimodal large language models. In *Find. Assoc. Comput. Linguist.* 12401–12430 (ACL, 2024).
26. Radford, A. et al. Learning transferable visual models from natural language supervision. In *Proc. 38th Int. Conf. Machine Learning* 8748–8763 (PMLR, 2021).
27. Liu, H., Li, C., Wu, Q. & Lee, Y. J. Visual instruction tuning. In *37th Conf. Neural Inform. Process. Syst.* (NeurIPS, 2023).
28. Yang, Z. et al. The dawn of LMMs: preliminary explorations with GPT-4V(ision). Preprint at https://arxiv.org/abs/2309.17421 (2023).
29. Zheng, Z. et al. Image and data mining in reticular chemistry powered by GPT-4V. *Digit. Discov.* **3**, 491–501 (2024).
30. Zhao, W. X. et al. A survey of large language models. Preprint at https://arxiv.org/abs/2303.18223 (2023).
31. Naveed, H. et al. A comprehensive overview of large language models. Preprint at https://arxiv.org/abs/2307.06435 (2024).
32. Ramos, M. C., Collison, C. J. & White, A. D. A review of large language models and autonomous agents in chemistry. Preprint at https://arxiv.org/abs/2407.01603 (2024).
33. Lei, G., Docherty, R. & Cooper, S. J. Materials science in the era of large language models: a perspective. *Digit. Discov.* **3**, 1257–1272 (2024).
34. Min, B. et al. Recent advances in natural language processing via large pre-trained language models: a survey. *ACM Comput. Surv.* **56**, 1–40 (2024).
35. Wei, J. et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **35**, 24824–24837 (2022).
36. Dong, Q. et al. A survey on in-context learning. Preprint at https://arxiv.org/abs/2301.00234 (2024).
37. Huang, J. & Chang, K. C.-C. Towards reasoning in large language models: a survey. In *Find. Assoc. Comput. Linguist.* 1049–1065 (ACL, 2023).
38. Zheng, Z. et al. A GPT-4 reticular chemist for guiding MOF discovery. *Angew. Chem. Int. Ed.* **62**, e202311983 (2023).
39. Maik Jablonka, K. et al. 14 examples of how LLMs can transform materials science and chemistry: a reflection on a large language model hackathon. *Digit. Discov.* **2**, 1233–1250 (2023).
40. Maynez, J., Narayan, S., Bohnet, B. & McDonald, R. On faithfulness and factuality in abstractive summarization. In *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.* 1906–1919 (ACL, 2020).

41. Zheng, Z. et al. Shaping the water-harvesting behavior of metal–organic frameworks aided by fine-tuned GPT models. *J. Am. Chem. Soc.* **145**, 28284–28295 (2023).
42. Zheng, Z. et al. ChatGPT research group for optimizing the crystallinity of MOFs and COFs. *ACS Cent. Sci.* **9**, 2161–2170 (2023).
43. Chung, H. W. et al. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.* **25**, 1–53 (2024).
44. Wang, Y. et al. Super-natural instructions: generalization via declarative instructions on 1600+ NLP tasks. In *Proc. 2022 Conf. Empir. Methods Nat. Lang. Process.* 5085–5109 (ACL, 2022).
45. Kim, S. et al. The CoT collection: improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning. In *Proc. 2023 Conf. Empir. Methods Nat. Lang. Process.* (ACL, 2023).
46. Yao, S. et al. Tree of thoughts: deliberate problem solving with large language models. In *37th Conf. Neural Inform. Process. Syst.* (NeurIPS, 2023).
47. Khattab, O. et al. DSPy: compiling declarative language model calls into self-improving pipelines. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2024).
48. Wang, X. et al. Self-consistency improves chain of thought reasoning in language models. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2024).
49. Ji, Z. et al. Towards mitigating LLM hallucination via self reflection. In *Find. Assoc. Comput. Linguist.* (eds Bouamor, H., Pino, J. & Bali, K.) 1827–1843 (ACL, 2023).
50. Asai, A., Wu, Z., Wang, Y., Sil, A. & Hajishirzi, H. Self-RAG: learning to retrieve, generate, and critique through self-reflection. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2023).
51. Boiko, D. A., MacKnight, R., Kline, B. & Gomes, G. Autonomous chemical research with large language models. *Nature* **624**, 570–578 (2023).
52. Kang, Y. & Kim, J. ChatMOF: an artificial intelligence system for predicting and generating metal–organic frameworks using large language models. *Nat. Commun.* **15**, 4705 (2024).
53. Ruan, Y. et al. An automatic end-to-end chemical synthesis development platform powered by large language models. *Nat. Commun.* **15**, 10160 (2024).
54. Lewis, P. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Inform. Process. Syst.* Vol. 33 9459–9474 (Curran Associates, 2020).
55. Gao, Y. et al. Retrieval-augmented generation for large language models: a survey. Preprint at https://arxiv.org/abs/2312.10997 (2024).
56. Liu, N. F. et al. Lost in the middle: how language models use long contexts. *Trans. Assoc. Comput. Linguist.* **12**, 157–173 (2024).
57. Ruan, Y. et al. Accelerated end-to-end chemical synthesis development with large language models. Preprint at https://doi.org/10.26434/chemrxiv-2024-6wmg4 (2024).
58. Jablonka, K. M., Schwaller, P., Ortega-Guerrero, A. & Smit, B. Leveraging large language models for predictive chemistry. *Nat. Mach. Intell.* **6**, 161–169 (2024).
59. Gupta, T., Zaki, M., Krishnan, N. M. A. & Mausam MatSciBERT: a materials domain language model for text mining and information extraction. *npj Comput. Mater.* **8**, 1–11 (2022).
60. Antunes, L. M., Butler, K. T. & Grau-Crespo, R. Crystal structure generation with autoregressive large language modeling. *Nat. Commun.* **15**, 10570 (2024).
61. Gruver, N. et al. Fine-tuned language models generate stable inorganic materials as text. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2024).
62. Kim, S., Jung, Y. & Schrier, J. Large language models for inorganic synthesis predictions. *J. Am. Chem. Soc.* **146**, 19654–19659 (2024).
63. Zhang, W. et al. Fine-tuning large language models for chemical text mining. *Chem. Sci.* **15**, 10600–10611 (2024).
64. Jiang, A. Q. et al. Mistral 7B. Preprint at https://arxiv.org/abs/2310.06825v1 (2023).
65. Beltagy, I., Lo, K. & Cohan, A. SciBERT: a pretrained language model for scientific text. Preprint at https://arxiv.org/abs/1903.10676 (2019).
66. Lewis, M. et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.* (ACL, 2020).
67. Raffel, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **21**, 1–67 (2020).
68. Hu, E. J. et al. Lora: low-rank adaptation of large language models. In *Proc. 10th Int. Conf. Learn. Represent.* (ICLR, 2021).
69. Han, Z., Gao, C., Liu, J., Zhang, J. & Zhang, S. Q. Parameter-efficient fine-tuning for large models: a comprehensive survey. *Trans. Mach. Learn. Res.* https://openreview.net/forum?id=lIsCS8b6zj (2024).
70. Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
71. Bai, X., Xie, Y., Zhang, X., Han, H. & Li, J.-R. Evaluation of open-source large language models for metal–organic frameworks research. *J. Chem. Inf. Model.* **64**, 4958–4965 (2024).
72. Luo, Y. et al. MOF synthesis prediction enabled by automatic data mining and machine learning. *Angew. Chem. Int. Ed.* **61**, e202200242 (2022).
73. Park, H., Kang, Y., Choe, W. & Kim, J. Mining insights on metal–organic framework synthesis from scientific literature texts. *J. Chem. Inf. Model.* **62**, 1190–1198 (2022).
74. Glasby, L. T. et al. DigiMOF: a database of metal–organic framework synthesis information generated via text mining. *Chem. Mater.* **35**, 4510–4524 (2023).
75. Park, S. et al. Text mining metal–organic framework papers. *J. Chem. Inf. Model.* **58**, 244–251 (2018).
76. Nandy, A. et al. MOFSimplify, machine learning models with extracted stability data of three thousand metal–organic frameworks. *Sci. Data* **9**, 74 (2022).
77. Nandy, A., Duan, C. & Kulik, H. J. Using machine learning and data mining to leverage community knowledge for the engineering of stable metal–organic frameworks. *J. Am. Chem. Soc.* **143**, 17535–17547 (2021).
78. Batra, R., Chen, C., Evans, T. G., Walton, K. S. & Ramprasad, R. Prediction of water stability of metal–organic frameworks using machine learning. *Nat. Mach. Intell.* **2**, 704–710 (2020).
79. Terrones, G. G. et al. Metal–organic framework stability in water and harsh environments from data-driven models trained on the diverse WS24 data set. *J. Am. Chem. Soc.* **146**, 20333–20348 (2024).
80. Lee, W., Kang, Y., Bae, T. & Kim, J. Harnessing large language model to collect and analyze metal-organic framework property dataset. Preprint at https://arxiv.org/abs/2404.13053.
81. Rampal, N. et al. Single and multi-hop question-answering datasets for reticular chemistry with GPT-4-turbo. *J. Chem. Theory Comput.* **20**, 9128–9137 (2024).
82. Ansari, M. & Moosavi, S. M. Agent-based learning of materials datasets from the scientific literature. *Digit. Discov.* **3**, 2607–2617 (2024).
83. Leong, S. X., Pablo-García, S., Zhang, Z. & Aspuru-Guzik, A. Automated electrosynthesis reaction mining with multimodal large language models (MLLMs). Preprint at https://doi.org/10.26434/chemrxiv-2024-7fwxv (2024).
84. Liu, S. et al. Conversational Drug Editing Using Retrieval and Domain Feedback. In *Proc. 12th Int. Conf. Learn. Represent.* (ICLR, 2024).
85. Ahn, J. et al. Large language models for mathematical reasoning: progresses and challenges. In *Proc. 18th Conf. Eur. Ch. Assoc. Comput. Linguist.* 225-237 (ACL, 2024)
86. Pinheiro, M., Martin, R. L., Rycroft, C. H. & Haranczyk, M. High accuracy geometric analysis of crystalline porous materials. *CrystEngComm* **15**, 7531–7538 (2013).
87. Willems, T. F., Rycroft, C. H., Kazi, M., Meza, J. C. & Haranczyk, M. Algorithms and tools for high-throughput geometry-based analysis of crystalline porous materials. *Microporous Mesoporous Mater.* **149**, 134–141 (2012).
88. Pinheiro, M. et al. Characterization and comparison of pore landscapes in crystalline porous materials. *J. Mol. Graph. Model.* **44**, 208–219 (2013).
89. Sarkisov, L. & Harrison, A. Computational structure characterisation tools in application to ordered and disordered porous materials. *Mol. Simul.* **37**, 1248–1257 (2011).
90. Sarkisov, L. & Kim, J. Computational structure characterization tools for the era of material informatics. *Chem. Eng. Sci.* **121**, 322–330 (2015).
91. Dubbeldam, D., Calero, S., Ellis, D. E. & Snurr, R. Q. RASPA: molecular simulation software for adsorption and diffusion in flexible nanoporous materials. *Mol. Simul.* **42**, 81–101 (2016).
92. Su, Y. et al. Automation and machine learning augmented by large language models in a catalysis study. *Chem. Sci.* **15**, 12200–12233 (2024).
93. Zheng, Z. et al. Integrating machine learning and large language models to advance exploration of electrochemical reactions. *Angew. Chem. Int. Ed.* **63**, e202418074 (2024).
94. Mahjour, B., Hoffstadt, J. & Cernak, T. Designing chemical reaction arrays using phactor and ChatGPT. *Org. Process Res. Dev.* **27**, 1510–1516 (2023).
95. Chiang, W.-L. et al. Chatbot arena: an open platform for evaluating LLMs by human preference. In *Proc. 41st Int. Conf. Mach. Learn.* (ICML, 2024).
96. An, Y. et al. Knowledge graph question answering for materials science (KGQA4MAT): developing natural language interface for metal-organic frameworks knowledge graph (MOF-KG) using LLM. In *17th Int. Conf. Metadata Semantics Res.* (Springer, 2023).
97. Shi, L. et al. LLM-based MOFs synthesis condition extraction using few-shot demonstrations. Preprint at https://arxiv.org/abs/2408.04665 (2024).
98. Rubungo, A. N., Li, K., Hattrick-Simpers, J. & Dieng, A. B. LLM4Mat-bench: benchmarking large language models for materials property prediction. Preprint at https://arxiv.org/abs/2411.00177 (2024).
99. de Vries, A. The growing energy footprint of artificial intelligence. *Joule* **7**, 2191–2194 (2023).
100. Wu, C.-J. et al. Sustainable AI: environmental implications, challenges and opportunities. Preprint at https://arxiv.org/abs/2111.00364 (2022).
101. Xu, M. et al. A survey of resource-efficient LLM and multimodal foundation models. Preprint at https://arxiv.org/abs/2401.08092 (2024).
102. Stojkovic, J., Choukse, E., Zhang, C., Goiri, I. & Torrellas, J. Towards greener LLMs: bringing energy-efficiency to the forefront of LLM inference. Preprint at https://arxiv.org/abs/2403.20306 (2024).
103. Morris, M. R. et al. Levels of AGI: operationalizing progress on the path to AGI. Preprint at https://arxiv.org/abs/2311.02462 (2024).
104. Gropp, C. et al. Standard practices of reticular chemistry. *ACS Cent. Sci.* **6**, 1255–1273 (2020).
105. Li, A. et al. The launch of a freely accessible MOF CIF collection from the CSD. *Matter* **4**, 1105–1106 (2021).
106. Yaghi, O. M., Li, G. & Li, H. Selective binding and removal of guests in a microporous metal–organic framework. *Nature* **378**, 703–706 (1995).
107. Côté, A. P. et al. Porous, crystalline, covalent organic frameworks. *Science* **310**, 1166–1170 (2005).
108. Park, K. S. et al. Exceptional chemical and thermal stability of zeolitic imidazolate frameworks. *Proc. Natl Acad. Sci. USA* **103**, 10186–10191 (2006).
109. Deng, H. et al. Multiple functional groups of varying ratios in metal–organic frameworks. *Science* **327**, 846–850 (2010).
110. Liu, Y. et al. Weaving of organic threads into a crystalline covalent organic framework. *Science* **351**, 365–369 (2016).
111. El-Kaderi, H. M. et al. Designed synthesis of 3D covalent organic frameworks. *Science* **316**, 268–272 (2007).
112. Yang, J. et al. Principles of designing extra-large pore openings and cages in zeolitic imidazolate frameworks. *J. Am. Chem. Soc.* **139**, 6448–6455 (2017).

113. Cmarik, G. E., Kim, M., Cohen, S. M. & Walton, K. S. Tuning the adsorption properties of UiO-66 via ligand functionalization. *Langmuir* **28**, 15606–15613 (2012).
114. Wang, Z. & Cohen, S. M. Postsynthetic covalent modification of a neutral metal–organic framework. *J. Am. Chem. Soc.* **129**, 12368–12369 (2007).
115. Li, H., Eddaoudi, M., O'Keeffe, M. & Yaghi, O. M. Design and synthesis of an exceptionally stable and highly porous metal–organic framework. *Nature* **402**, 276–279 (1999).
116. Seo, J. S. et al. A homochiral metal–organic porous material for enantioselective separation and catalysis. *Nature* **404**, 982–986 (2000).
117. Ni, Z. & Masel, R. I. Rapid production of metal–organic frameworks via microwave-assisted solvothermal synthesis. *J. Am. Chem. Soc.* **128**, 12394–12395 (2006).
118. Pichon, A., Lazuen-Garay, A. & James, S. L. Solvent-free synthesis of a microporous metal–organic framework. *CrystEngComm* **8**, 211–214 (2006).
119. Wilmer, C. E. et al. Large-scale screening of hypothetical metal–organic frameworks. *Nat. Chem.* **4**, 83–89 (2012).
120. Chung, Y. G. et al. Computation-ready, experimental metal–organic frameworks: a tool to enable high-throughput screening of nanoporous crystals. *Chem. Mater.* **26**, 6185–6192 (2014).
121. Bobbitt, N. S. et al. MOFX-DB: an online database of computational adsorption data for nanoporous materials. *J. Chem. Eng. Data* **68**, 483–498 (2023).
122. Chung, Y. G. et al. Advances, updates, and analytics for the computation-ready, experimental metal–organic framework database: CoRE MOF 2019. *J. Chem. Eng. Data* **64**, 5985–5998 (2019).
123. Rosen, A. S. et al. Machine learning the quantum-chemical properties of metal–organic frameworks for accelerated materials discovery. *Matter* **4**, 1578–1597 (2021).
124. Rosi, N. L. et al. Hydrogen storage in microporous metal-organic frameworks. *Science* **300**, 1127–1129 (2003).
125. Millward, A. R. & Yaghi, O. M. Metal–organic frameworks with exceptionally high capacity for storage of carbon dioxide at room temperature. *J. Am. Chem. Soc.* **127**, 17998–17999 (2005).
126. Horcajada, P. et al. Metal–organic frameworks as efficient materials for drug delivery. *Angew. Chem. Int. Ed.* **45**, 5974–5978 (2006).
127. Feng, D. et al. Zirconium-metalloporphyrin PCN-222: mesoporous metal–organic frameworks with ultrahigh stability as biomimetic catalysts. *Angew. Chem. Int. Ed.* **51**, 10307 (2012).
128. Furukawa, H. et al. Water adsorption in porous metal–organic frameworks and related materials. *J. Am. Chem. Soc.* **136**, 4369–4381 (2014).
129. Zhou, Z. et al. Carbon dioxide capture from open air using covalent organic frameworks. *Nature* **635**, 96–101 (2024).
130. Sheberla, D. et al. Conductive MOF electrodes for stable supercapacitors with high areal capacitance. *Nat. Mater.* **16**, 220–224 (2017).

Acknowledgements

The authors thank the Defense Advanced Research Projects Agency (DARPA) for the financial support under contract HR0011-21-C-0020. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA.

Author contributions

O.M.Y. contributed to writing, review and editing, and supervision. Z.Z. contributed to writing, review and editing, and graphics. N.R., T.J.I., J.T.C. and C.B. contributed to review and editing. Z.Z. used ChatGPT-4 for grammar and typos checking during the review and editing of this manuscript. All authors have read, corrected and verified all information presented in this manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Peer review information *Nature Reviews Materials* thanks Seyed Mohamad Moosavi and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.