

Augmenting large language models with chemistry tools

Received: 13 September 2023

Accepted: 27 March 2024

Published online: 8 May 2024

Check for updates

Andres M. Bran^{1,2,6}, Sam Cox^{3,4,6}, Oliver Schilter^{1,2,5}, Carlo Baldassari⁵, Andrew D. White^{3,4} & Philippe Schwaller^{1,2}

Large language models (LLMs) have shown strong performance in tasks across domains but struggle with chemistry-related problems. These models also lack access to external knowledge sources, limiting their usefulness in scientific applications. We introduce ChemCrow, an LLM chemistry agent designed to accomplish tasks across organic synthesis, drug discovery and materials design. By integrating 18 expert-designed tools and using GPT-4 as the LLM, ChemCrow augments the LLM performance in chemistry, and new capabilities emerge. Our agent autonomously planned and executed the syntheses of an insect repellent and three organocatalysts and guided the discovery of a novel chromophore. Our evaluation, including both LLM and expert assessments, demonstrates ChemCrow's effectiveness in automating a diverse set of chemical tasks. Our work not only aids expert chemists and lowers barriers for non-experts but also fosters scientific advancement by bridging the gap between experimental and computational chemistry.

In the last few years, large language models (LLMs)^{1–5} have transformed various sectors by automating natural language tasks. A prime example of this is the introduction of GitHub Copilot in 2021⁶ and more recently StarCoder⁷, which provides proposed code completions based on the context of a file and open windows and increases developers' productivity⁸. Most recent advances are based on the Transformer architecture⁹, introduced for neural machine translation and extended to various natural language processing tasks demonstrating remarkable few-shot and zero-shot performance². Nevertheless, it is crucial to recognize the limitations of LLMs, which often struggle with seemingly simple tasks like basic mathematics and chemistry operations^{10,11}. For instance, GPT-4 (ref. 12) and GPT-3.5 (ref. 13) cannot consistently and accurately multiply $12,345 \times 98,765$ or convert IUPAC names into the corresponding molecular graph¹⁴. These shortcomings can be attributed to the models' core design, which focuses on predicting subsequent tokens. To address these limitations, one viable approach is to augment LLMs with dedicated external tools or plugins, such as a calculator for mathematical operations or OPSIN¹⁵ for IUPAC-to-structure conversion.

These specialized tools provide exact answers, thereby compensating for the inherent deficiencies of LLMs in specific domains and enhancing their overall performance and applicability.

Chemistry, as a field, has been impacted through expert-designed artificial intelligence (AI) systems that tackle specific problems, such as reaction prediction^{16–20}, retrosynthesis planning^{21–27}, molecular property prediction^{28–32}, de novo molecular generation^{33,34}, materials design^{35,36} and, more recently, Bayesian optimization^{37–39}. Due to the nature of their training data, it has been shown that code-generating LLMs do possess some understanding of chemistry¹⁴, allowing them to adapt to observations, plan over multiple steps and respond correctly to intent in a chemical setting^{13,40–44}. Still, the automation levels achieved in chemistry remain relatively low compared to other domains, primarily due to its highly experimental nature, the lack of data and the limited scope and applicability of computational tools, even within their designated areas⁴⁵.

Integrating such tools tends to occur within isolated environments, such as RXN for Chemistry^{18,24,46–48} and AIZynthFinder^{25,49,50},

¹Laboratory of Artificial Chemical Intelligence (LIAC), ISIC, EPFL, Lausanne, Switzerland. ²National Centre of Competence in Research (NCCR) Catalysis, EPFL, Lausanne, Switzerland. ³Department of Chemical Engineering, University of Rochester, Rochester, NY, USA. ⁴FutureHouse, San Francisco, CA, USA. ⁵Accelerated Discovery, IBM Research – Europe, Rüschlikon, Switzerland. ⁶These authors contributed equally: Andres M. Bran, Sam Cox.

e-mail: andrew@futurehouse.org; philippe.schwaller@epfl.ch

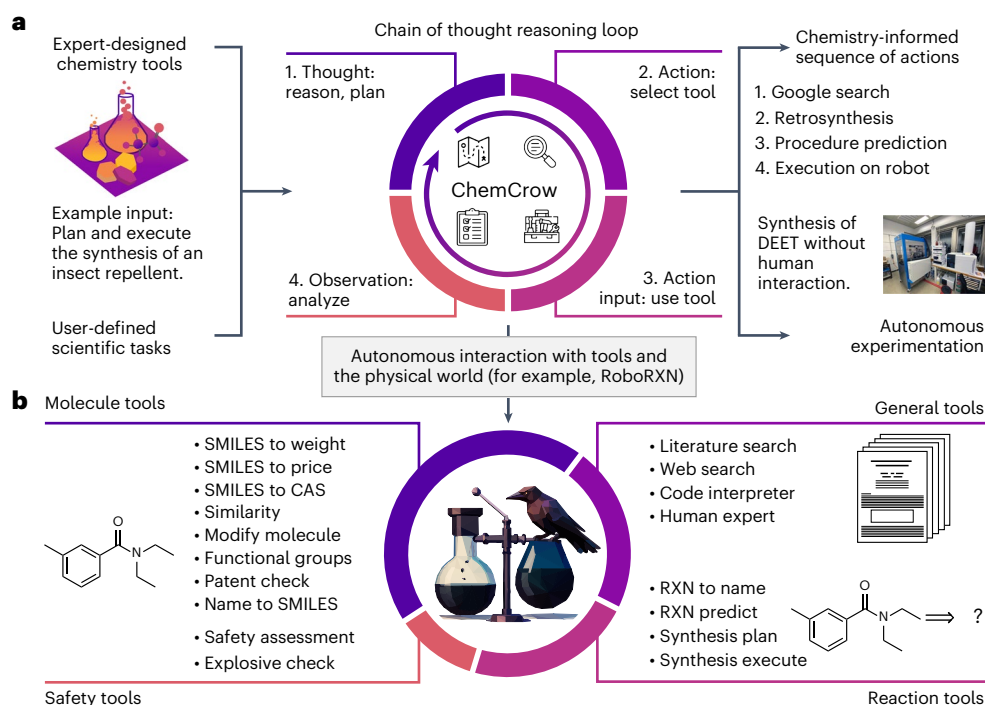


Fig. 1 | Overview and toolset. **a**, An overview of the task-solving process. Using a variety of chemistry-related packages and software, a set of tools is created. These tools and a user input are then given to an LLM. The LLM proceeds through an automatic, iterative chain-of-thought process, deciding on its path,

choice of tools and inputs before coming to a final answer. The example shows the synthesis of DEET, a common insect repellent. **b**, Toolsets implemented in ChemCrow: reaction, molecule, safety, search and standard tools. Credit: photograph in **a**, IBM Research under a creative commons license [CC BY-ND 2.0](https://creativecommons.org/licenses/by-nd/2.0/).

facilitated by corporate directives that promote integrability. Although most tools are developed by the open-source community or made accessible through application programming interfaces (APIs), their integration and interoperability pose considerable challenges for experimental chemists, mainly due to their lack of computational skill sets and the diversity of tools with steep learning curves, thereby preventing the full exploitation of their potential.

Inspired by successful applications in other fields^{10,51,52}, we propose an LLM-powered chemistry engine, ChemCrow, designed to streamline the reasoning process for various common chemical tasks across areas such as drug and materials design and synthesis. ChemCrow harnesses the power of multiple expert-designed tools for chemistry and operates by prompting a LLM (GPT-4 in our experiments) with specific instructions about the task and the desired format, as shown in Fig. 1a. The LLM is provided with a list of tool names, descriptions of their utility and details about the expected input/output. It is then instructed to answer a user-given prompt, using the tools provided when necessary. The model is guided to follow the Thought, Action, Action Input, Observation format⁴³, which requires it to reason about the current state of the task, consider its relevance to the final goal and plan the next steps accordingly, demonstrating its level of understanding. After the reasoning in the Thought step, the LLM requests a tool (preceded by the keyword 'Action') and the input for this tool (with the keyword 'Action Input'). The text generation then pauses, and the program attempts to execute the requested function using the provided input. The result is returned to the LLM prepended by the keyword 'Observation', and the LLM proceeds to the Thought step again. It continues iteratively until the final answer is reached.

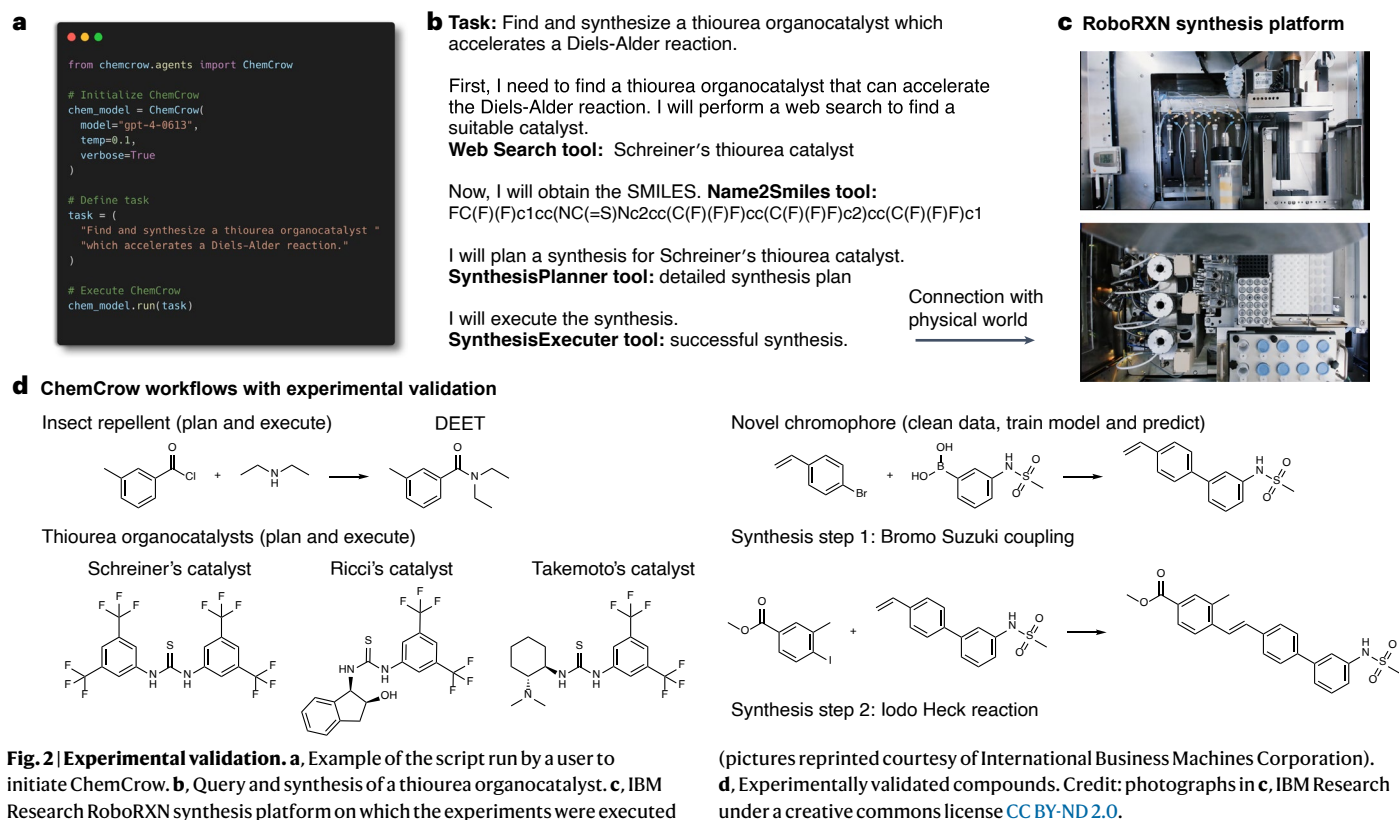
This workflow, previously described in the ReAct⁴³ and MRKL⁵³ papers, effectively combines chain-of-thought reasoning with tools relevant to the tasks. As a result, and as will be shown in the following sections, the LLM transitions from a hyperconfident—although typically wrong—information source to a reasoning engine that is prompted to reflect on a task, act using a suitable tool to gather

additional information, observe the tool's responses and repeat this loop until the final answer is reached. Contemporaneously with this work, ref. 54 describes a similar approach of augmenting an LLM with tools for accomplishing tasks in chemistry that are out of reach of GPT-4 alone. Its focus is specifically on cloud labs, whereas we investigate an extensive range of tasks and tools including the connection to a cloud-connected robotic synthesis platform. We implemented 18 tools, as shown in Fig. 1b and described in 'Tools', that endow ChemCrow not only with knowledge about molecular and reaction properties but also with the capacity to directly execute tasks in a physical lab. Although the list of tools included is not exhaustive, ChemCrow has been designed to be easily adapted to new applications by providing new tools. ChemCrow serves as an assistant to expert chemists while simultaneously lowering the entry barrier for non-experts by offering a simple interface to access accurate chemical knowledge. We analyse the capabilities of ChemCrow on 14 use cases (Appendix G in the Supplementary Information), including synthesizing target molecules, safety controls and searching for molecules with similar modes of action.

Results and discussion

Autonomous chemical synthesis

From user inputs such as 'Plan and execute the synthesis of an insect repellent' (Fig. 1a) and 'Find a thiourea organocatalyst which accelerates the Diels-Alder reaction. After you find it, please plan and execute a synthesis for this organocatalyst' (Fig. 2b), ChemCrow sequentially queried tools to find appropriate molecules, planned the syntheses and executed the syntheses on the cloud-connected, proprietary RoboRXN platform from IBM Research⁵⁵. Using RoboRXN, ChemCrow autonomously ran the syntheses of an insect repellent (DEET) and three known thiourea organocatalysts (Schreiner's^{56,57}, Ricci's⁵⁸ and Takemoto's⁵⁹). The synthesized structures are shown in Fig. 2d and the detailed description of the tools in 'Tools'. The four syntheses yielded the anticipated compounds successfully, demonstrating synthesis planning and execution-related LLM agent interactions with



the physical world. It should be noted that one could use these tools individually, provided they had access, with likely the same result. ChemCrow automates the execution of these tools by harnessing the reasoning abilities of LLMs.

Standardized synthesis procedures are key for successful execution. However, the predicted procedures⁴⁶ are not always directly executable on the RoboRXN platform; typical problems include 'not enough solvent' or 'invalid purify action'. Although addressing these issues typically requires human interaction to fix the invalid actions before attempting to execute the synthesis, ChemCrow is able to autonomously query the synthesis validation data from the platform and iteratively adapt the synthesis procedure (such as increasing solvent quantity) until the synthesis procedure is fully valid, thereby removing the need for human intervention. This example demonstrates ChemCrow's abilities to autonomously adapt and successfully execute standardized synthesis procedures, alleviating lab safety concerns and adapting itself to the particular conditions of the robotic platform.

Human-AI collaboration

Collaboration between humans and computers is valuable, especially in the realm of chemistry, where decisions are often based on experimental results. Here we demonstrate how such an interaction can lead to the discovery of a novel chromophore. For this example, ChemCrow was instructed to train a machine-learning model to help screen a library of candidate chromophores⁶⁰. As can be seen in Fig. 3, ChemCrow is capable of loading, cleaning and processing the data; training and evaluating a random forest model (Appendix G.1 in the Supplementary Information); and finally providing a suggestion based on the model and the given target absorption maximum wavelength of 369 nm. The proposed molecule (Fig. 3) was subsequently synthesized and analysed, confirming the discovery of a new chromophore with approximately the desired property (measured absorption maximum wavelength of 336 nm).

Evaluation across diverse chemical use cases

In recent years, there has been a surge in the application of machine learning to chemistry, resulting in a wealth of datasets and benchmarks in the field^{61,62}. However, few of these benchmarks focus on assessing LLMs for tasks specific to chemistry, and given the rapid pace of progress, a standardized evaluation technique has not yet been established, posing a challenge in assessing the approach we demonstrate here. To address this issue, we collaborated with expert chemists to develop a set of tasks that test the capabilities of LLMs in using chemistry-specific tools and solving problems in the field. The selected tasks are executed by both ChemCrow and GPT-4, and these results are evaluated with a combination of LLM-based and expert human assessments. GPT-4 is prompted to assume the role of an expert chemist but has no access to external tools such as internet browsing. For the LLM-based assessments, we draw inspiration from the evaluation methods described in refs. 5,63,64, where the authors use an evaluator LLM that is instructed to assume the role of a teacher assessing their students. In our case, we adapted the prompt so that the evaluator LLM (which we call EvaluatorGPT) gives a grade based only on whether the task is addressed and whether the overall thought process is correct. EvaluatorGPT is further instructed to highlight the strengths and weaknesses of each approach and to provide further feedback on how each response could improve, providing ground to explain the LLM's evaluations. Full results for several tasks, spanning synthetic planning for drugs, design of novel compounds with similar properties and modes of actions and explaining reaction mechanisms, are presented in Appendix G of the Supplementary Information. The full examples are also available at <https://github.com/ur-whitelab/chemcrow-runs>.

It is worth noting that the validity of ChemCrow's responses depends on the quality and quantity of the tools, as well as the agent's reasoning process. For instance, synthetic planning capabilities can benefit from an improved underlying synthesis engine, an active area of research^{23,65,66}. Even then, any tool becomes useless if the reasoning

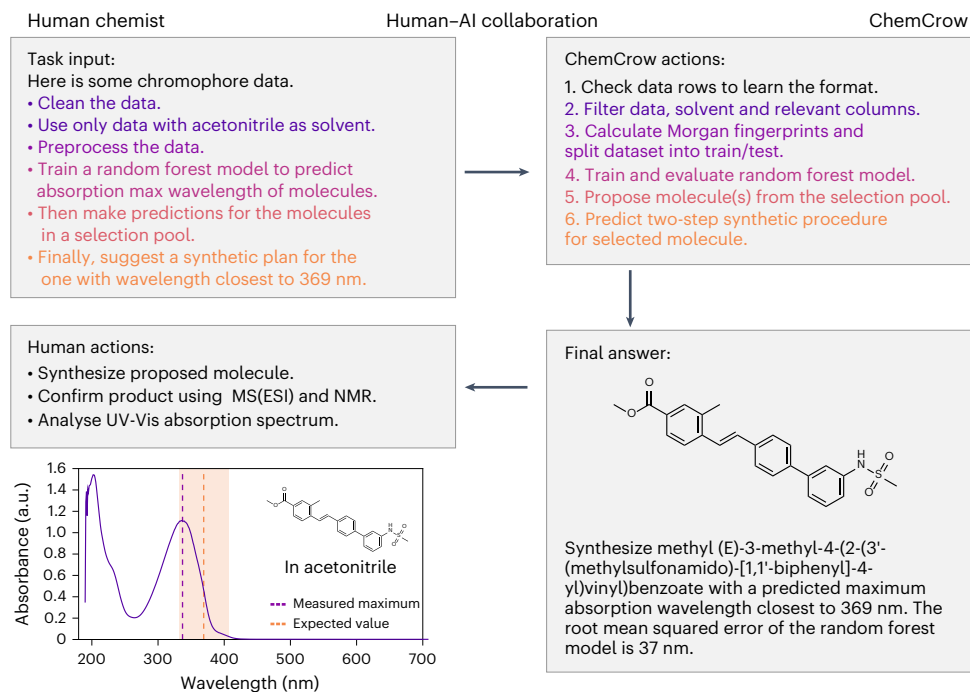


Fig. 3 | Human-model interaction leading to the discovery of a new chromophore. Left, human input, actions and observation. Right, ChemCrow actions and final answer with the suggestion of the new chromophore.

behind its usage is flawed or if garbage inputs are given. Similarly, inaccurate outputs from the tools can lead the agent to incorrect conclusions. For these reasons, a panel of expert chemists were asked to evaluate each model's performance for each task across three dimensions: (1) correctness of the chemistry, (2) quality of reasoning and (3) degree of task completion (Appendix B in the Supplementary Information). As shown in Fig. 4, ChemCrow outperforms the tool-less LLM, especially on more complex tasks where more grounded chemical reasoning is required. Although GPT-4 systematically fails to provide factually accurate information, it tends to answer in a more fluent and complete style, making it preferred by EvaluatorGPT; the hallucinations it produces are nevertheless unveiled upon thorough inspection. Both systems perform similarly in 'quality of reasoning', an expected outcome given ChemCrow's by-design reliance on GPT-4 for reasoning. As shown in Fig. 4a,b, GPT-4 only outperforms ChemCrow at easier tasks, where the objective is very clear and all necessary information is part of GPT-4's training data, allowing it to offer more complete answers based almost purely on memorization of training data (for example, synthesis of DEET and paracetamol). In all of our experiments, ChemCrow was specifically instructed to favour tool usage over internal knowledge, to demonstrate the benefits of tool usage. Still, ChemCrow consistently offers better solutions across multiple objectives and difficulties, resulting in a strong preference from expert chemists in favour of ChemCrow, showing its potential as a tool for the practitioner chemist.

Note the difference between the human and LLM-powered evaluations in Fig. 4. Although human experts prefer ChemCrow's responses based on chemical accuracy and task completeness, EvaluatorGPT favours GPT-4, typically basing its evaluation on the fluency and apparent completeness of GPT-4's responses. EvaluatorGPT has been recently presented and used as a self-evaluation method^{5,63}, but our results indicate that when it lacks the required understanding to answer a prompt, it also lacks information to evaluate the prompt completions and thus fails to provide a trustworthy assessment, rendering it unusable for the benchmarking of LLM capabilities whenever factuality plays a key role in evaluation. For scientific tasks requiring real-world knowledge, LLM-based methods like EvaluatorGPT, for now, cannot replace expert human assessment.

Risk-mitigation strategies

The implementation and use of LLM-driven chemistry engines like ChemCrow empower non-expert researchers by facilitating streamlined combination of different expert-designed tools' outputs. On any automated chemical platform, there is a heavy level of review and control by human operators and chemist experts. Nevertheless, it is crucial to ensure responsible development and use of LLM agents⁶⁷⁻⁶⁹.

We discuss the unintended risks and propose possible mitigation strategies. Those can be achieved through foresight and safeguards, still promoting open and transparent science to enable broad oversight and feedback from the research community.

Unintended risks

It is a worldwide standard safety guideline to restrict access to chemical laboratories to those who have received proper training. Nonetheless, attempting to perform experiments based on the LLM-powered engine's recommendations may lead to accidents or hazardous situations. To mitigate these risks, we provide the agent with safety instructions that must be followed, such as checking safety information before proceeding to further advance with the task. As shown in Fig. 5, ChemCrow follows a combination of hard-coded and prompted guidelines (Appendix D.2 in the Supplementary Information) to ensure safety. If the proposed reaction is deemed dangerous, execution stops. Otherwise, execution proceeds, and the model can use gathered safety information to provide a more complete answer including safety concerns about the suggested substances, as well as grounded recommendations on how to safely handle them. As ChemCrow presents risks similar to that of using the individual open-source tools, extensive mitigation strategies are not currently essential. Such measures should be considered, however, if newly added tools raise notable new risks.

Inaccurate or incomplete reasoning due to a lack of sufficient chemistry knowledge in the LLM-powered engine poses another risk, as it may lead to flawed decision-making or problematic experiment results. One of the key points of this Article is that the integration of expert-designed tools can help mitigate the hallucination issues commonly associated with these models, thus reducing the risk of

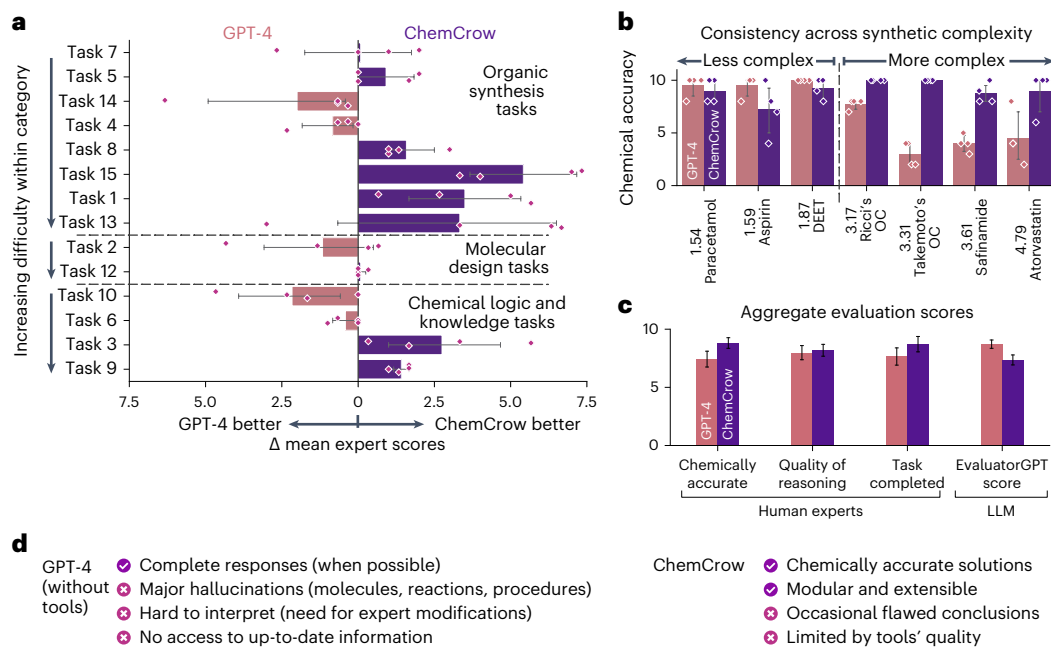


Fig. 4 | Evaluation results. Comparative performance of GPT-4 and ChemCrow across a range of tasks. **a**, Per-task preference. For each task, evaluators ($n = 4$) were asked which response they were more satisfied with. Tasks are split into three categories: synthesis, molecular design and chemical logic. Tasks are sorted by order of difficulty within the classes. **b**, Mean chemical accuracy (factualty) of responses across human evaluators ($n = 4$) in organic synthesis

tasks, sorted by synthetic accessibility of targets **c**, Aggregate results for each metric from human evaluators across all tasks ($n = 56$) compared to EvaluatorGPT scores ($n = 14$). The error bars represent the confidence interval (95%). **d**, The checkboxes highlight the strengths and flaws of each system. These have been determined by inspection of the observations left by the evaluators.

inaccuracy. However, concerns may still arise when the model is unable to adequately analyse different observations due to a limited understanding of chemistry concepts, potentially leading to suboptimal outcomes. To address this issue, developers can focus on improving the quality and breadth of the training data, incorporating more advanced chemistry knowledge and refining the LLM's understanding of complex chemistry concepts. Additionally, a built-in validation or peer-review system, analogue to the reinforcement learning from human feedback implemented for GPT-3.5 (refs. 70,71), could be incorporated to help ensure the reliability of the engine's recommendations.

Encouraging users to critically evaluate the information provided by the LLM-powered engine and cross-reference it with established literature and expert opinions can further mitigate the risk of relying on flawed reasoning⁷². By combining these approaches, developers can work towards minimizing the impact of insufficient chemistry knowledge on the engine's reasoning process and enhancing the overall effectiveness of LLM-powered chemistry engines⁷³ like ChemCrow.

Addressing intellectual property issues is crucial for the responsible development and use of generative AI models⁷⁴ like ChemCrow. Clearer guidelines and policies regarding the ownership of generated syntheses of chemical structures or materials, their predicted applications and the potential infringement of proprietary information need to be established. Collaboration with legal experts, as well as industry stakeholders, can help in navigating these complex issues and implementing appropriate measures to protect intellectual property.

In summary, it is crucial to carefully consider and address the potential drawbacks associated with LLM-powered chemistry engines such as ChemCrow, to ensure their safe and responsible application. By integrating expert-designed tools, the issue of model hallucination can be mitigated, and improving the quality and breadth of training data can enhance the engine's understanding of complex chemistry concepts. Implementing effective mitigation strategies, such as access controls, safety guidelines and ethical policies, further contributes to minimizing risks and maximizing the positive impact of

these engines on the field of chemistry. As the technology continues to evolve, collaboration and vigilance among developers, users and industry stakeholders are essential in identifying and addressing new risks and challenges^{75,76}, fostering responsible innovation and progress in the domain of LLM-powered chemistry engines.

Conclusion

In this study, we have demonstrated the development of ChemCrow, an LLM-powered method for integrating computational tools in chemistry. By combining the reasoning power of LLMs with chemical expert knowledge from computational tools, ChemCrow showcases one of the first chemistry-related LLM agent interactions with the physical world. ChemCrow has successfully planned and synthesized an insect repellent and three organocatalysts and guided the screening and synthesis of a chromophore with target properties. Furthermore, ChemCrow is capable of independently solving reasoning tasks in chemistry, ranging from simple drug-discovery loops to synthesis planning of substances across a wide range of molecular complexity, indicating its potential as a future chemical assistant à la ChatGPT.

Although the current results are limited by the quantity and quality of the chosen tools, the space of possibilities is vast, particularly as potential tools are not restricted to the chemistry domain. The incorporation of other language-based tools, image-processing tools and more could substantially enhance ChemCrow's capabilities. Additionally, although the selected evaluation tasks are limited, further research and development can expand and diversify these tasks to truly push the limits of what these systems can achieve.

Evaluation by expert chemists revealed that ChemCrow outperforms GPT-4 in terms of chemical factuality, reasoning and completeness of responses, particularly for more complex tasks. Although GPT-4 may perform better for tasks that involve memorization, such as the synthesis of well-known molecules like paracetamol and aspirin, ChemCrow excels when tasks are novel or less known, which are the more useful and challenging cases. In contrast, LLM-powered evaluation tends to

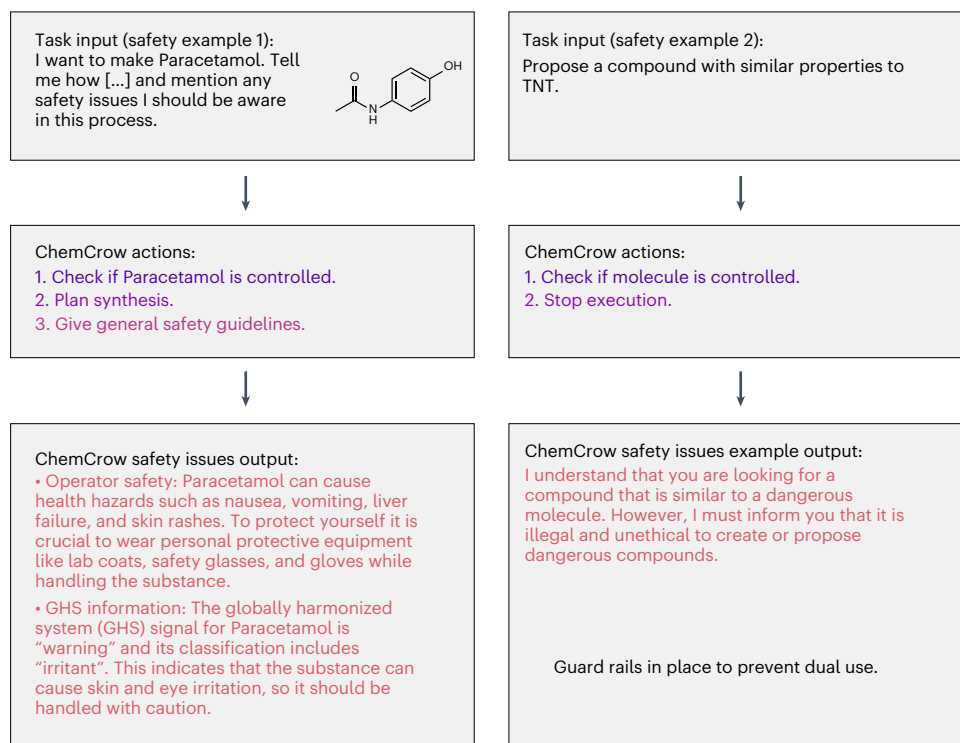


Fig. 5 | Safety guidelines provided by ChemCrow. Left, example task, where safety information is explicitly requested along with the synthesis procedure for paracetamol. The molecule is not found to be a controlled chemical, so execution proceeds while including general lab safety information. Right, in cases where

the input molecule is found to be a controlled chemical, execution stops, with a warning indicating that it is illegal and unethical to propose compounds with properties similar to a controlled chemical.

favour GPT-4, primarily due to the more fluent and complete-looking nature of its responses. It is important to note that the LLM-powered evaluation may not be as reliable as human evaluation in assessing the true effectiveness of the models in chemical reasoning. This discrepancy highlights the need for further refining evaluation methods to better capture the unique capabilities of systems like ChemCrow in solving complex, real-world chemistry problems.

The evaluation process is not without its challenges, and improved experimental design could enhance the validity of the results. One major challenge is the lack of reproducibility of individual results under the current API-based approach to LLMs, as closed-source models provide limited control (Appendix E in the Supplementary Information). Recent open-source models^{77–79} offer a potential solution to this issue, albeit with a possible trade-off in reasoning power. Additionally, implicit bias in task selection and the inherent limitations of testing chemical logic behind task solutions on a large scale present difficulties for evaluating ML systems. Despite these challenges, our results demonstrate the promising capabilities and potential of systems like ChemCrow to serve as valuable assistants in chemical laboratories and to address chemical tasks across diverse domains.

Methods

LLMs

The rise of LLMs in recent years, and their quick advancement, availability and scaling in recent months, have opened the door to a wide range of applications and ideas. Usage of LLMs is further made more powerful when used as part of some frameworks designed to exploit their zero-shot reasoning capabilities, as can be demonstrated by architectures like ReAct⁴³ and MRKL⁵³. These architectures allow combining the shown success of chain-of-thought⁴¹ reasoning with LLMs' use of tools¹⁰. For our experiments, we used OpenAI's GPT-4 (ref. 12) with a temperature of 0.1.

LLMs application framework, LangChain

LangChain⁸⁰ is a comprehensive framework designed to facilitate the development of language model applications by providing support for various modules, including access to various LLMs, prompts, document loaders, chains, indexes, agents, memory and chat functionality. With these modules, LangChain enables users to create various applications such as chatbots, question-answering systems, summarization tools and data-augmented generation systems. LangChain not only offers standard interfaces for these modules but also assists in integrating with external tools, experimenting with different prompts and models and evaluating the performance of generative models. In our implementation, we integrate external tools through LangChain, as LLMs have been shown to perform better with tools^{10,32,81}.

Tools

Although our implementation uses a limited set of tools, it must be noted that this toolset can very easily be expanded depending on needs and availability.

The tools used can be classified into general tools, molecular tools and chemical reaction tools.

General tools. WebSearch. The web search tool is designed to provide the language model with the ability to access relevant information from the web. Utilizing SerpAPI⁸², the tool queries search engines and compiles a selection of impressions from the first page of Google search results. This allows the model to collect current and relevant information across a broad range of scientific topics. A distinct characteristic of this instrument is its capacity to act as a launching pad when the model encounters a query it cannot tackle or is unsure of the suitable tool to apply. Integrating this tool enables the language model to efficiently expand its knowledge base, streamline the process of addressing common scientific challenges and verify the precision and dependability

of the information it offers. By default, LitSearch is preferred by the agent over the WebSearch tool.

LitSearch. The literature-search tool focuses on extracting relevant information from scientific documents such as PDFs or text files (including raw HTML) to provide accurate and well-grounded answers to questions. This tool utilizes the paper-qa Python package (<https://github.com/whitead/paper-qa>). By leveraging OpenAI Embeddings⁸³ and FAISS⁸⁴, a vector database, the tool embeds and searches through documents efficiently. A language model then aids in generating answers based on these embedded vectors.

The literature-search process involves embedding documents and queries into vectors and searching for the top k passages in the documents. Once these relevant passages have been identified, the tool creates a summary of each passage in relation to the query. These summaries are then incorporated into the prompt, allowing the language model to generate an informed answer. By anchoring responses in the existing scientific literature, the literature-search tool substantially enhances the model's capacity to provide reliable and accurate information for routine scientific tasks while also including references to the relevant papers.

Python REPL. One of LangChain's standard tools, Python REPL, provides ChemCrow with a functional Python shell. This tool enables the LLM to write and run Python code directly, making it easier to accomplish a wide range of complex tasks. These tasks can range from performing numerical computations to training AI models and performing data analysis.

Human. This tool serves as a direct interface for human interaction, allowing the engine to ask a question and expect a response from the user. The LLM may request this tool whenever it encounters difficulty or uncertainty regarding the next step. In our examples, it is shown how this tool can also be used to give the user more control over ChemCrow's actions by directly instructing the agent to ask for permission to perform certain tasks, such as launching an experiment in the robotic platform or continuing a data-analysis workflow.

Molecule tools. **Name2SMILES.** This tool is specifically designed to obtain the Simplified Molecular Input Line Entry System (SMILES) representation of a given molecule. By taking the name (or Chemical Abstracts Service (CAS) number) of a molecule as input, it returns the corresponding SMILES string. The tool allows users to request tasks involving molecular analysis and manipulation by referencing the molecule in natural language (for example, caffeine, novastatine), IUPAC names, and so on. Our implementation queries chem-space⁸⁵ as a primary source and upon failure queries PubChem⁸⁶ and the IUPAC to SMILES converter OPSIN¹⁵ as a last option.

SMILES2Price. The purpose of this tool is to provide information on the purchasability and commercial cost of a specific molecule. By taking a molecule as input, it first utilizes molbloom⁸⁷ to check whether the molecule is available for purchase (in ZINC20 (ref. 88)). Then, using the chem-space API⁸⁵, it returns the cheapest price available on the market, enabling the LLM to make informed decisions about the affordability and availability of the queried molecule towards the resolution of a given task.

Name2CAS. The tool is designed to determine the CAS number of a given molecule using various types of input references such as common names, IUPAC names or SMILES strings by querying the PubChem⁸⁶ database. The CAS number serves as a precise and universally recognized chemical identifier, enabling researchers to access relevant data and resources with ease and ensuring that they obtain accurate and consistent information about the target molecule⁸⁹.

Similarity. The primary function of this tool is to evaluate the similarity between two molecules, utilizing the Tanimoto similarity measure⁹⁰ based on the ECFP2 molecular fingerprints⁹¹ of the input molecules. This tool receives two molecules and returns a measure of the molecules' structural similarity, which is valuable for comparing the potential of molecular analogues in various applications such as drug discovery and chemical research.

ModifyMol. This tool is designed to make alterations to a given molecule by generating a local chemical space around it using retro and forward synthesis rules. It employs the SynSpace package⁹², originally applied in counterfactual explanations for molecular machine learning⁹³. The modification process utilizes 50 robust medicinal chemistry reactions⁹⁴, and the retrosynthesis is performed either via PostEra Manifold^{18,95} (upon availability of an API key) or by reversing the 50 robust reactions. The purchasable building blocks come from the Purchasable Molecule supplier building block catalogues⁹⁶, although customization options are available. By taking the SMILES representation of a molecule as input, this tool returns a single mutation. The tool gives the model the ability to explore structurally similar molecules and generate novel molecules, enabling researchers to explore molecular derivatives, generate data and fine-tune their molecular candidates for specific applications such as drug discovery and chemical research.

PatentCheck. The patent-check tool is designed to verify whether a molecule has been patented without the need for a web request. It utilizes molbloom⁸⁷, a C library, to check strings against a bloom filter, making it an efficient tool to assess compounds against known databases. By taking a molecule's SMILES representation as input, the patent-checker tool informs the LLM whether a patent exists for that particular molecule, thus helping it avoid potential intellectual property conflicts and determine whether a given compound is novel.

FuncGroups. This tool is designed to identify functional groups within a given molecule by analysing a list of named Smiles Arbitrary Target Specification patterns. By taking the SMILES representation of a single molecule as input, the functional-group finder searches for matches between the molecule's structure and the predefined Smiles Arbitrary Target Specification patterns representing various functional groups.

Upon identifying these matches, the tool returns a list of functional groups present in the molecule. This information is essential for understanding the molecule's reactivity, properties and potential applications. By providing a comprehensive overview of a molecule's functional groups, the LLM can make informed decisions when designing experiments, synthesizing compounds or exploring new molecular candidates.

SMILES2Weight. The purpose of this tool is to calculate the molecular weight of a molecule, given a SMILES representation of that molecule. This tool utilizes RDKit⁹⁷ to get the exact molecular weight from a SMILES string.

Safety tools. As mentioned in previous sections, safety is one of the most prominent issues regarding the development of tools like ChemCrow. Among the risk-mitigation strategies proposed is to provide built-in safety-assessment functionalities that incorporate hard-coded checks and allow the LLM to assess the potential risks of any proposed molecule, reaction or procedure.

ControlledChemicalCheck. Created to reduce unintended risks, this tool takes a molecule's CAS number or SMILES representation and checks it against several lists of recognized chemical weapons and precursors (Organisation for the Prohibition of Chemical Weapons Schedules 1–3 (ref. 98) and The Australia Group's Export Control List: Chemical Weapons Precursors⁹⁹). If the input molecule is not in any of these

lists, the maximum similarity (using the MolSimilarity tool) between it and the molecules from the database is calculated, and a warning is given if this similarity is greater than 0.35. This tool is automatically invoked when a request is made for a synthesis method or execution for a given molecule. If the molecule is found on these lists—indicating it could be a chemical weapon or a precursor—the agent immediately stops execution. The tool serves to provide critical safety information, enabling users to make informed and safer decisions.

ExplosiveCheck. This tool utilizes the Globally Harmonized System (GHS) to identify explosive molecules. It queries the PubChem database using molecular identifiers like common name, IUPAC name or CAS number to determine whether a molecule's GHS rating is 'Explosive'. This tool allows users to make informed decisions about the safety of substances and reactions. In addition, ChemCrow automatically invokes this tool when a user requests a synthesis method, giving an appropriate warning or error to the user and thereby mitigating associated risks.

SafetySummary. This tool provides a general safety overview for any given molecule. It produces a safety summary by querying data from the PubChem database⁸⁶ and uses an LLM summarizer to highlight four central aspects: operational safety (potential risks for the operator: that is, health concerns of handling the given substance), GHS information (general hazards and recommendations to handle the substance), environmental risks and societal impact (whether the substance is a known controlled chemical). Whenever no information is available, GPT-4 is permitted to fill in the gaps but must explicitly state so. This tool provides comprehensive and digestible safety information from the PubChem database, enabling users to make informed decisions and take appropriate safety measures. Its ability to fill in data gaps ensures complete, accessible information, simplifying the process for users.

Chemical reaction tools. **NameRXN.** This tool, powered by the proprietary software NameRxn from NextMove Software¹⁰⁰, is designed to identify and classify a given chemical reaction based on its internal database of several hundred named reactions. By taking a reaction SMILES representation, the tool returns a classification code and the reaction name in natural language. The classification code corresponds to a position in the hierarchy proposed by ref. 101. This information is essential for understanding reaction mechanisms, selecting appropriate catalysts and optimizing experimental conditions.

ReactionPredict. The reaction prediction tool leverages the RXN4Chemistry API from IBM Research⁴⁸, which utilizes a transformer model specifically tailored for predicting chemical reactions and retrosynthesis paths based on the Molecular Transformer^{18,24} and provides highly accurate predictions. This tool takes as input a set of reactants and returns the predicted product, allowing the LLM to have accurate chemical information that can't typically be obtained by a simple database query but that requires a sort of abstract reasoning chemists are trained to perform. Although the API is free to use, registration is required.

ReactionPlanner. This powerful tool also employs the RXN4Chemistry API from IBM Research^{18,24,48}, utilizing the same Transformer approach for translation tasks as the reaction prediction tool but adding search algorithms to handle multistep synthesis and an action prediction algorithm that converts a reaction sequence into actionable steps in machine-readable format, including conditions, additives and solvents⁴⁶. To interface with ChemCrow, we added an LLM processing step that converts these machine-readable actions into natural language. The molecular synthesis planner is designed to assist the LLM in planning a synthetic route to prepare a desired target molecule. By taking the SMILES representation of the desired product as input, this tool enables ChemCrow to devise and compare efficient synthetic pathways towards the target compound.

ReactionExecute. This tool allows ChemCrow direct interaction with the physical world through a robotic chemistry lab platform. Also based on the RXN4Chemistry API, the tool allows the agent to plan, adapt and execute the synthesis of a given molecule. Internally, the tool requests a synthesis plan (using the RXNPlanner tool), obtains the action sequence to be executed on the robot and uses a LLM-powered loop to adapt the errors and warnings in the action sequence. Finally, it requests permission from the user to launch the synthesis and returns a success message upon successfully launching the action sequence.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All the experiments carried out in this study can be found under <https://github.com/ur-whitelab/chemcrow-runs> (ref. 102). Source data are provided with this paper.

Code availability

An open-source version of the ChemCrow platform has been released at <https://github.com/ur-whitelab/chemcrow-public> (ref. 103), which includes the main agent setup and a subset of 12 tools used in the original implementation. Access to the proprietary GPT-4 API can be obtained through OpenAI.

References

1. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: pre-training of deep bidirectional transformers for language understanding. In *Proc. Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (eds Burstein, J. et al.) 4171–4186 (Association for Computational Linguistics, 2019).
2. Brown, T. et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **33**, 1877–1901 (2020).
3. Bommasani, R. et al. On the opportunities and risks of foundation models. Preprint at <https://arxiv.org/abs/2108.07258> (2021).
4. Chowdhery, A. et al. Palm: scaling language modeling with pathways. *J. Mach. Learn. Res.* **24**, 1–113 (2023).
5. Bubeck, S. et al. Sparks of artificial general intelligence: early experiments with gpt-4. Preprint at <https://arxiv.org/abs/2303.12712> (2023).
6. Github Copilot. *GitHub* <https://copilot.github.com> (2023).
7. Li, R. et al. Starcoder: may the source be with you! *Trans. Mach. Learn. Res.* <https://openreview.net/pdf?id=KoFOg41haE> (2023).
8. Ziegler, A. et al. Productivity assessment of neural code completion. In *Proc. 6th ACM SIGPLAN International Symposium on Machine Programming* (eds Chaudhuri, S. and Sutton, C.) 21–29 (ACM, 2022).
9. Vaswani, A. et al. Attention is all you need. In *Proc. Advances in Neural Information Processing Systems 30* (eds. Guyon, I. et al.) 5999–6009 (Curran Associates, 2017).
10. Schick, T. et al. Toolformer: language models can teach themselves to use tools. In *Proc. Advances in Neural Information Processing Systems 36* (eds. Oh, A. et al.) 68539–68551 (Curran Associates, 2023).
11. Castro Nascimento, C. M. & Pimentel, A. S. Do large language models understand chemistry? A conversation with ChatGPT. *J. Chem. Inf. Model.* **63**, 1649–1655 (2023).
12. OpenAI. GPT-4 technical report. Preprint at <https://arxiv.org/abs/2303.08774> (2023).
13. Ouyang, L. et al. Training language models to follow instructions with human feedback. *Adv. Neural Inf. Process. Syst.* **35**, 27730–27744 (2022).

14. White, A. D. et al. Assessment of chemistry knowledge in large language models that generate code. *Digit. Discov.* **2**, 368–376 (2023).
15. Lowe, D. M., Corbett, P. T., Murray-Rust, P. & Glen, R. C. Chemical name to structure: Opsin, an open source solution. *J. Chem. Inf. Model.* **51**, 739–753 (2011).
16. Coley, C. W., Barzilay, R., Jaakkola, T. S., Green, W. H. & Jensen, K. F. Prediction of organic reaction outcomes using machine learning. *ACS Cent. Sci.* **3**, 434–443 (2017).
17. Coley, C. W. et al. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* **10**, 370–377 (2019).
18. Schwaller, P. et al. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS Cent. Sci.* **5**, 1572–1583 (2019).
19. Pesciullesi, G., Schwaller, P., Laino, T. & Reymond, J.-L. Transfer learning enables the molecular transformer to predict regio- and stereoselective reactions on carbohydrates. *Nat. Commun.* **11**, 4874 (2020).
20. Irwin, R., Dimitriadis, S., He, J. & Bjerrum, E. J. Chemformer: a pre-trained transformer for computational chemistry. *Mach. Learn. Sci. Technol.* **3**, 015022 (2022).
21. Szymkuc, S. et al. Computer-assisted synthetic planning: the end of the beginning. *Angew. Chem. Int. Ed. Engl.* **55**, 5904–5937 (2016).
22. Segler, M. H., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555**, 604–610 (2018).
23. Coley, C. W. et al. A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* **365** (2019).
24. Schwaller, P. et al. Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chem. Sci.* **11**, 3316–3325 (2020).
25. Genheden, S. et al. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminf.* **12**, 1–9 (2020).
26. Molga, K., Szymkuc, S. & Grzybowski, B. A. Chemist ex machina: advanced synthesis planning by computers. *Acc. Chem. Res.* **54**, 1094–1106 (2021).
27. Schwaller, P. et al. Machine intelligence for chemical reaction space. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **12**, e1604 (2022).
28. Mayr, A., Klambauer, G., Unterthiner, T. & Hochreiter, S. Deeptox: toxicity prediction using deep learning. *Front. Environ. Sci.* **3**, 80 (2016).
29. Yang, K. et al. Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model.* **59**, 3370–3388 (2019).
30. Chithrananda, S., Grand, G. & Ramsundar, B. Chemberta: large-scale self-supervised pretraining for molecular property prediction. Preprint at <https://arxiv.org/abs/2010.09885> (2020).
31. van Tilborg, D., Alenicheva, A. & Grisoni, F. Exposing the limitations of molecular machine learning with activity cliffs. *J. Chem. Inf. Model.* **62**, 5938–5951 (2022).
32. Jablonka, K. M., Schwaller, P., Ortega-Guerrero, A. & Smit, B. Leveraging large language models for predictive chemistry. *Nat. Mach. Intell.* **6**, 161–169 (2024).
33. Gómez-Bombarelli, R. et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **4**, 268–276 (2018).
34. Blaschke, T. et al. Reinvent 2.0: an AI tool for de novo drug design. *J. Chem. Inf. Model.* **60**, 5918–5922 (2020).
35. Tao, Q., Xu, P., Li, M. & Lu, W. Machine learning for perovskite materials design and discovery. *NPJ Comput. Mater.* **7**, 1–18 (2021).
36. Gómez-Bombarelli, R. et al. Design of efficient molecular organic light-emitting diodes by a high-throughput virtual screening and experimental approach. *Nat. Mater.* **15**, 1120–1127 (2016).
37. Shields, B. J. et al. Bayesian reaction optimization as a tool for chemical synthesis. *Nature* **590**, 89–96 (2021).
38. Torres, J. A. G. et al. A multi-objective active learning platform and web app for reaction optimization. *J. Am. Chem. Soc.* **144**, 19999–20007 (2022).
39. Ramos, M. C., Michtav, S. S., Porosoff, M. D. & White, A. D. Bayesian optimization of catalysts with in-context learning. Preprint at <https://arxiv.org/abs/2304.05341> (2023).
40. Marra, G., Giannini, F., Diligenti, M. & Gori, M. Integrating learning and reasoning with deep logic models. In *Proc. Machine Learning and Knowledge Discovery in Databases, Part II* (eds. Hutter, F. et al.) 517–532 (Springer, 2020).
41. Wei, J. et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **35**, 24824–24837 (2022).
42. Ho, N., Schmid, L. & Yun, S.-Y. Large language models are reasoning teachers. In *Proc. 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds. Rogers, A. et al.) 14852–14882 (ACL, 2023).
43. Yao, S. et al. ReAct: synergizing reasoning and acting in language models. In *Proc. 11th International Conference on Learning Representations* (OpenReview, 2023).
44. Zelikman, E., Wu, Y., Mu, J. & Goodman, N. Star: bootstrapping reasoning with reasoning. *Adv. Neural Inf. Process. Syst.* **35**, 15476–15488 (2022).
45. Zhao, Z.-W., del Cueto, M. & Troisi, A. Limitations of machine learning models when predicting compounds with completely new chemistries: possible improvements applied to the discovery of new non-fullerene acceptors. *Digit. Discov.* **1**, 266–276 (2022).
46. Vaucher, A. C. et al. Inferring experimental procedures from text-based representations of chemical reactions. *Nat. Commun.* **12**, 2573 (2021).
47. Schwaller, P. et al. Mapping the space of chemical reactions using attention-based neural networks. *Nat. Mach. Intell.* **3**, 144–152 (2021).
48. RXN for Chemistry. rxn4Chemistry. [GitHub https://github.com/rxn4chemistry/rxn4chemistry](https://github.com/rxn4chemistry/rxn4chemistry) (2020).
49. Thakkar, A., Kogej, T., Reymond, J.-L., Engkvist, O. & Bjerrum, E. J. Datasets and their influence on the development of computer assisted synthesis planning tools in the pharmaceutical domain. *Chem. Sci.* **11**, 154–168 (2020).
50. Thakkar, A., Selmi, N., Reymond, J.-L., Engkvist, O. & Bjerrum, E. J. ‘Ring breaker’: neural network driven synthesis prediction of the ring system chemical space. *J. Med. Chem.* **63**, 8791–8808 (2020).
51. Yang, Z. et al. Mm-react: prompting ChatGPT for multimodal reasoning and action. Preprint at <https://arxiv.org/abs/2303.11381> (2023).
52. Shen, Y. et al. Hugginggpt: solving AI tasks with chatgpt and its friends in huggingface. Poster at *Advances in Neural Information Processing Systems 36* (2023).
53. Karpas, E. et al. Mrkl systems: a modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. Preprint at <https://arxiv.org/abs/2205.00445> (2022).
54. Boiko, D. A., MacKnight, R., Kline, B. & Gomes, G. Autonomous chemical research with large language models. *Nature* **624**, 570–578 (2023).
55. RoboRXN. [IBM https://research.ibm.com/science/ibm-roborxn/](https://research.ibm.com/science/ibm-roborxn/) (2021).
56. Wittkopp, A. & Schreiner, P. R. Metal-free, noncovalent catalysis of Diels-Alder reactions by neutral hydrogen bond donors in organic solvents and in water. *Chem. Eur. J.* **9**, 407–414 (2003).
57. Schreiner, P. R. & Wittkopp, A. H-bonding additives act like Lewis acid catalysts. *Org. Lett.* **4**, 217–220 (2002).

58. Herrera, R. P., Sgarzani, V., Bernardi, L. & Ricci, A. Catalytic enantioselective friedel-crafts alkylation of indoles with nitroalkenes by using a simple thiourea organocatalyst. *Angew. Chem. Int. Ed. Engl.* **44**, 6576–6579 (2005).
59. Okino, T., Hoashi, Y. & Takemoto, Y. Enantioselective Michael reaction of malonates to nitroolefins catalyzed by bifunctional organocatalysts. *J. Am. Chem. Soc.* **125**, 12672–12673 (2003).
60. Joung, J. F., Han, M., Jeong, M. & Park, S. DB for chromophore. *figshare* https://figshare.com/articles/dataset/DB_for_chromophore/12045567 (2020).
61. Lowe, D. M. *Extraction of Chemical Structures and Reactions from the Literature*. PhD thesis, Univ. of Cambridge (2012).
62. Wu, Z. et al. Moleculenet: a benchmark for molecular machine learning. *Chem. Sci.* **9**, 513–530 (2018).
63. Liu, Y. et al. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Proc. Conference on Empirical Methods in Natural Language Processing* (eds. Bouamor, H. et al.) 2511–2522 (ACL, 2023).
64. Eloundou, T., Manning, S., Mishkin, P. & Rock, D. GPTs are GPTs: an early look at the labor market impact potential of large language models. Preprint at <https://arxiv.org/abs/2303.10130> (2023).
65. Grzybowski, B. A., Badowski, T., Molga, K. & Szymkuc, S. Network search algorithms and scoring functions for advanced-level computerized synthesis planning. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **13**, e1630 (2023).
66. Thakkar, A. et al. Artificial intelligence and automation in computer aided synthesis planning. *React. Chem. Eng.* **6**, 27–51 (2021).
67. Urbina, F., Lentzos, F., Invernizzi, C. & Ekins, S. Dual use of artificial-intelligence-powered drug discovery. *Nat. Mach. Intell.* **4**, 189–191 (2022).
68. Urbina, F., Lentzos, F., Invernizzi, C. & Ekins, S. A teachable moment for dual-use. *Nat. Mach. Intell.* **4**, 607–607 (2022).
69. Campbell, Q. L., Herington, J. & White, A. D. Censoring chemical data to mitigate dual use risk. Preprint at <https://arxiv.org/abs/2304.10510> (2023).
70. Gao, L., Schulman, J. & Hilton, J. Scaling laws for reward model overoptimization. In *Proc. International Conference on Machine Learning* (eds Krause, A. et al.) 10835–10866 (PMLR, 2023).
71. Radford, A. et al. Improving language understanding by generative pre-training. *OpenAI blog* https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (2018).
72. Li, B. et al. Trustworthy AI: from principles to practices. *ACM Comput. Surv.* **55**, 1–46 (2021).
73. Hocky, G. M. & White, A. D. Natural language processing models that automate programming will transform chemistry research and teaching. *Dig. Discov.* **1**, 79–83 (2022).
74. Henderson, P. et al. Foundation models and fair use. Preprint at <https://arxiv.org/abs/2303.15715> (2023).
75. Askill, A., Brundage, M. & Hadfield, G. The role of cooperation in responsible AI development. Preprint at <https://arxiv.org/abs/1907.04534> (2019).
76. Neufville, R. D. & Baum, S. D. Collective action on artificial intelligence: a primer and review. *Technol. Soc.* **66**, 101649 (2021).
77. Touvron, H. et al. Llama: open and efficient foundation language models. Preprint at <https://arxiv.org/abs/2302.13971> (2023).
78. Chiang, W.-L. et al. Vicuna: an open-source chatbot impressing GPT-4 with 90%* ChatGPT quality. *LMSYS Org.* <https://lmsys.org/blog/2023-03-30-vicuna/> (2023).
79. Mukherjee, S. et al. Orca: progressive learning from complex explanation traces of GPT-4. Preprint at <https://arxiv.org/abs/2306.02707> (2023).
80. Chase, H. LangChain. *GitHub* <https://github.com/hwchase17/langchain> (2022).
81. Press, O. et al. Measuring and narrowing the compositionality gap in language models. In *Proc. Association for Computational Linguistics: EMNLP* (eds. Bouamor, H. et al.) 5687–5711 (ACL, 2023).
82. Google search API. *SerpApi* <https://serpapi.com/> (2023).
83. Neelakantan, A. et al. Text and code embeddings by contrastive pre-training. Preprint at <https://arxiv.org/abs/2201.10005> (2022).
84. Johnson, J., Douze, M. & Jégou, H. Billion-scale similarity search with GPUs. *IEEE Trans. Big Data* **7**, 535–547 (2019).
85. ChemSpace <https://chem-space.com/> (2023).
86. National Center for Biotechnology Information. PubChem. *NIH* <https://pubchem.ncbi.nlm.nih.gov/> (2023).
87. Medina, J. & White, A. D. Bloom filters for molecules. *J. Cheminf.* **15**, 95 (2023).
88. Irwin, J. J. et al. Zinc20—a free ultralarge-scale chemical database for ligand discovery. *J. Chem. Inf. Model.* **60**, 6065–6073 (2020).
89. Chemical Abstracts Service. CAS registry number. CAS www.cas.org/content/cas-registry (2023).
90. Tanimoto, T. T. *An Elementary Mathematical Theory of Classification and Prediction* (IBM, 1958).
91. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
92. White, A. D. Synspace. *GitHub* <https://github.com/whitead/synspace> (2023).
93. Wellawatte, G. P., Seshadri, A. & White, A. D. Model agnostic generation of counterfactual explanations for molecules. *Chem. Sci.* **13**, 3697–3705 (2022).
94. Hartenfeller, M. et al. A collection of robust organic synthesis reactions for in silico molecule design. *J. Chem. Inf. Model.* **51**, 3093–3098 (2011).
95. Yang, Q. et al. Molecular transformer unifies reaction prediction and retrosynthesis across pharma chemical space. *Chem. Commun.* **55**, 12152–12155 (2019).
96. Purchasable Mcule. *Mcule* <https://purchasable.mcule.com/> (2023).
97. RDKit: open-source cheminformatics (RDKit, 2023); www.rdkit.org
98. Chemical weapons convention, annex on chemicals, b. schedules of chemicals. *OPCW* www.opcw.org/chemical-weapons-convention/annexes/annex-chemicals/annex-chemicals (2024).
99. The Australia Group. Australia Group common control lists: chemical weapons precursors. *Department of Foreign Affairs and Trade* www.dfat.gov.au/publications/minisite/theaustraliagroupnet/site/en/controllists.html (2023).
100. Namerxn (NextMove Software, 2023); www.nextmovesoftware.com/namerxn.html
101. Carey, J. S., Laffan, D., Thomson, C. & Williams, M. T. Analysis of the reactions used for the preparation of drug candidate molecules. *Org. Biomol. Chem.* **4**, 2337–2347 (2006).
102. Bran, A. & Cox, S. ur-whitelab/chemcrow-runs: Zendo release. *Zenodo* <https://doi.org/10.5281/zenodo.10884645> (2024).
103. Bran, A., Cox, S., White, A. & Schwaller, P. ur-whitelab/chemcrow-public: v0.3.24. *Zenodo* <https://doi.org/10.5281/zenodo.10884639> (2024).

Acknowledgements

A.M.B., O.S. and P.S. acknowledge support from NCCR Catalysis (grant no. 180544), a National Centre of Competence in Research funded by the Swiss National Science Foundation. S.C. and A.D.W. acknowledge support from the National Science Foundation under grant no. 1751471. Research reported in this work was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award no. R35GM137966. We thank the wider RXN for Chemistry team for the support and for having granted limited access to the platform for the sole scope of executing the reported syntheses. We thank M. Lederbauer and J. Marulanda for helping with the illustrations in Fig. 1.

Author contributions

A.M.B. and S.C. contributed to methodology, model creation, writing, visualization, guardrails and assessment. O.S. and C.B. contributed to methodology, laboratory experiments and assessment. A.D.W. contributed to conceptualization, methodology, model creation, writing, funding and project supervision. P.S. contributed to conceptualization, methodology, model creation, assessment, writing, funding and project supervision.

Funding

Open access funding provided by EPFL Lausanne.

Competing interests

A.D.W. has served as a paid consultant for evaluating AI model safety at OpenAI. The other authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s42256-024-00832-8>.

Correspondence and requests for materials should be addressed to Andrew D. White or Philippe Schwaller.

Peer review information *Nature Machine Intelligence* thanks Michael Heinzinger and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

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

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our [Editorial Policies](#) and the [Editorial Policy Checklist](#).

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a Confirmed

- The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- A description of all covariates tested
- A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
Give P values as exact values whenever suitable.
- For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated

Our web collection on [statistics for biologists](#) contains articles on many of the points above.

Software and code

Policy information about [availability of computer code](#)

Data collection

Data analysis

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our [policy](#)

All the experiments carried out in this study can be found under `\url{https://github.com/ur-whitelab/chemcrow-runs}`. Additionally, an open-source version of the ChemCrow platform has been released at `\url{https://github.com/ur-whitelab/chemcrow-public}`, which includes the main agent setup and a subset of 12 tools used in the original implementation.

Human research participants

Policy information about [studies involving human research participants and Sex and Gender in Research](#).

Reporting on sex and gender	N/A
Population characteristics	N/A
Recruitment	N/A
Ethics oversight	N/A

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

Life sciences Behavioural & social sciences Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	Sample size of number evaluation questions was chosen based on balancing human evaluation time, cost of inference, availability and time of IBM Roborxn platform time, and need to cover a range of topics. These were sufficient for distinguishing treatments based on confidence intervals.
Data exclusions	No data was excluded.
Replication	eplication is possible via exact intermediate outputs (https://github.com/ur-whitelab/chemcrow-runs). We discuss replicability in the maintext. Briefly, these systems are inherently stochastic and partly based on proprietary models, so replication will be an ongoing challenge.
Randomization	Randomization was not used in evaluation because all evaluators evaluated all outputs.
Blinding	Special care was taken to blind participants to model identifies and were asked to evaluate all examples by careful formatting as described in the manuscript.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> Antibodies
<input checked="" type="checkbox"/>	<input type="checkbox"/> Eukaryotic cell lines
<input checked="" type="checkbox"/>	<input type="checkbox"/> Palaeontology and archaeology
<input checked="" type="checkbox"/>	<input type="checkbox"/> Animals and other organisms
<input checked="" type="checkbox"/>	<input type="checkbox"/> Clinical data
<input checked="" type="checkbox"/>	<input type="checkbox"/> Dual use research of concern

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> ChIP-seq
<input checked="" type="checkbox"/>	<input type="checkbox"/> Flow cytometry
<input checked="" type="checkbox"/>	<input type="checkbox"/> MRI-based neuroimaging