Are large language models superhuman chemists?

Adrian Mirza [1,★], Nawaf Alampara [1,★], Sreekanth Kunchapu [1,★],
Martiño Ríos-García [1,x ★], Benedict Emoekabu , Aswanth Krishnan [2],
Tanya Gupta [3,4], Mara Wilhelmi [1], Macjonathan Okereke [1],
Mehrdad Asgari [5], Juliane Eberhardt [6], Amir Mohammad Elahi [7],
Hani M. Elbeheiry [1], María Victoria Gil [x], Christina Glaubitz ,
Maximilian Greiner[1], Caroline T. Holick [1], Tim Hoffmann [1],
Abdelrahman Ibrahim [1], Lea C. Klepsch [1], Yannik Köster [1],
Fabian Alexander Kreth [8, 9], Jakob Meyer[1], Santiago Miret [10],
Jan Matthias Peschel [1], Michael Ringleb [1], Nicole Roesner [1, 11],
Johanna Schreiber [1, 10], Ulrich S. Schubert [1,8,11, 12], Leanne M. Stafast [1, 11],
Dinga Wonanke [13], Michael Pieler [14,15], Philippe Schwaller [3,4], and
Kevin Maik Jablonka [1,8,11, 12, ✉]

[1]Laboratory of Organic and Macromolecular Chemistry (IOMC), Friedrich Schiller University Jena,
Humboldtstrasse 10, 07743 Jena, Germany
[2]QpiVolta Technologies Pvt Ltd
[3]Laboratory of Artificial Chemical Intelligence (LIAC), Institut des Sciences et Ingénierie Chimiques, Ecole
Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
[4]National Centre of Competence in Research (NCCR) Catalysis, Ecole Polytechnique Fédérale de Lausanne
(EPFL), Lausanne, Switzerland
[5]Department of Chemical Engineering & Biotechnology, University of Cambridge, Philippa Fawcett Drive,
Cambridge CB3 0AS, United Kingdom
[6]Macromolecular Chemistry, University of Bayreuth, 95447 Bayreuth, Germany
[7]Laboratory of Molecular Simulation (LSMO), Institut des Sciences et Ingénierie Chimiques, Ecole Polytechnique
Fédérale de Lausanne (EPFL), Sion, Switzerland
[8]Center for Energy and Environmental Chemistry Jena (CEEC Jena), Friedrich Schiller University Jena,
Philosophenweg 7a, 07743 Jena, Germany
[9]Institute for Technical Chemistry and Environmental Chemistry (ITUC), Friedrich Schiller University Jena,
Philosophenweg 7a, 07743 Jena, Germany
[10]Intel Labs
[11]Jena Center for Soft Matter (JCSM), Friedrich Schiller University Jena, Philosophenweg 7, 07743 Jena, Germany
[12]Helmholtz Institute for Polymers in Energy Applications Jena (HIPOLE Jena), Lessingstrasse 12-14, 07743 Jena,
Germany
[13]Theoretical Chemistry, Technische Universität Dresden, Dresden 01062, Germany
[14]OpenBioML.org
[15]Stability.AI
✉`mail@kjablonka.com`
[★]These authors contributed equally.

October 15, 2024

Abstract

Large language models (LLMs) have gained widespread interest due to their ability to process human language. Also, LLMs have shown promise in addressing the limited data problem in chemistry and are increasingly being harnessed to improve and extend existing workflows.

However, we possess only a limited systematic understanding of the reasoning capabilities of LLMs, which would be required to improve models and mitigate potential harms. Here, we introduce "ChemBench," an automated framework for evaluating the chemical knowledge and reasoning abilities of state-of-the-art LLMs against the expertise of chemists.

We curated more than 2,800 question-answer pairs, evaluated leading open and closed-source LLMs, and found that, on average, the best models outperformed the chemists. The models, however, struggle with some chemical reasoning tasks that are easy for the human experts and provide overconfident, misleading predictions, such as about chemicals' safety profiles. Although LLMs demonstrate remarkable proficiency in chemical tasks, further research is critical to enhancing their safety and utility in chemical sciences.

1 Introduction

Large language models (LLMs) are machine learning (ML) models trained on massive amounts of text to complete sentences. Aggressive scaling of these models has led to a rapid increase in their capabilities,[1,2] with the leading models now being able to pass in some evaluations the United States Medical Licensing Examination.[3] They also have been shown to design and autonomously perform chemical reactions when augmented with external tools such as web search and synthesis planners.[4–6] While some see "sparks of artificial general intelligence (AGI)" in them,[7] others consider them as "stochastic parrots"—i.e., systems that only regurgitate what they have been trained on.[8] Nevertheless, the promise of these models is that they have shown the ability to solve a wide variety of tasks they have not been explicitly trained on.[9–11] This has led to tremendous economic interest and investment in such generative models, with an expected market of more than $1.3 trillion (almost $30 billion for drug discovery applications) by 2032.[12]

Chemists and materials scientists have quickly caught on to the mounting attention given to LLMs, with some voices even suggesting that "the future of chemistry is language".[13] This statement is motivated by a growing number of reports that use LLMs to predict properties of molecules or materials,[2,14–18] optimize reactions[19,20] generate materials,[21–23] extract information,[24–29] or to even prototype systems that can autonomously perform reactions in the physical world based on commands provided in natural language.[5,6,30]

In addition, since a lot—if not most—of the information about chemistry is currently stored and communicated in text, there is a strong reason to believe that there is still a lot of untapped potential in LLMs for chemistry and materials science.[31] For instance, most insights in chemical research do not directly originate from data stored in databases but rather from the scientists and their ability to interpret data. Many of these insights are in the form of text in scientific publications. Thus, operating on such texts might be our best way of unlocking these insights and learning from them. This might ultimately lead to general copilot systems for chemists that can provide answers to questions or even suggest new experiments based on vastly more information than a human could ever read. Such a usage mode is, in particular, interesting in the face of recent advances in autonomous laboratories.[5,6,30,32–36]

However, the rapid increase in capabilities of chemical ML models led (even before the recent interest in LLMs) to concerns about the potential for dual use of these technologies, e.g., for the design of chemical weapons.[37–42] To some extent, this is not surprising as any technology that, for instance, is used to design non-toxic molecules can also be used inversely to predict toxic ones (even though the synthesis would still require access to controlled physical resources and facilities). Still, it is essential to realize that such models' user base is wider than that of chemistry and materials science experts who critically reflect on every output such models produce. For example, many students frequently consult these tools—perhaps even to prepare chemical experiments.[43] This also applies to users from the general public, who might consider using LLMs to answer questions about the safety of chemicals. Thus, for some users, misleading information—especially about safety-related aspects—might lead to harmful outcomes. However, even for experts, chemical understanding and reasoning capabilities are essential as they will determine the capabilities and limitations of their models in their work, e.g., in copilot systems for chemists. Unfortunately, apart from anecdotal reports, there is little evidence on how LLMs perform compared to expert (human) chemists.

Thus, to better understand what LLMs can do for the chemical sciences and where they might be improved with further developments, evaluation frameworks are needed to allow us to measure progress and mitigate potential harms systematically. For the development of LLMs, evaluation is currently primarily performed via standardized benchmark suites such as BigBench[44] or the LM Eval Harness.[45] Among 204 tasks (such as linguistic puzzles), the former contains only two tasks classified as "chemistry related", whereas the latter contains no specific chemistry tasks. Due to the lack of widely accepted standard benchmarks, the developers of chemical language models[15,46–49] frequently utilize language-interfaced[50] tabular datasets such as the ones reported in MoleculeNet,[51] Therapeutic Data Commons[52] or MatBench.[53] In these cases, the models are evaluated on predicting very specific properties of molecules (e.g., solubility, toxicity, melting temperature or reactivity) or on predicting the outcome of specific chemical reactions. This, however, only gives a very limited view of the general chemical capabilities of the models.

While some benchmarks based on university entrance exams[54,55] or automatic text mining[56–58] have been proposed, none of them have been widely accepted. This is likely because they cannot automatically be used with black box (or tool-augmented) systems, do not cover a wide range of topics, or are not carefully validated by experts. On top of that, the existing benchmarks are not designed to be used with models that support special treatment of molecules or equations and do not provide insights on how the models compare relative to experts.

In this work, we report a novel benchmarking framework (Figure 1), which we call ChemBench, and use it to reveal limitations of current frontier models for use in the chemical sciences. Our benchmark consists of 2854 question-answer pairs compiled from diverse sources (2103 manually generated, and 751 automatically generated). Our corpus covers a large fraction of the topics taught in undergraduate and graduate chemistry curricula. It can be used to evaluate any system that can return text (i.e., including tool-augmented systems).

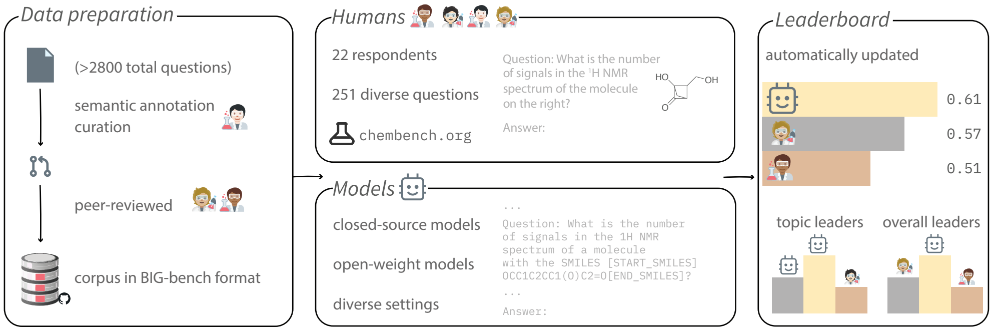

Figure 1: Overview of the ChemBench framework. The figure shows the different components of the ChemBench framework. The framework's foundation is the benchmark corpus that consists of many questions and answers that we manually or semi-automatically compiled from various sources. We then used this corpus to evaluate the performance of various models and tool-augmented systems using a custom framework. To provide a baseline, we built a web application that we used to survey experts in chemistry. The results of the evaluations are then compiled in publicly accessible leaderboards, which we propose as a foundation for evaluating future models.

To contextualize the scores, we also surveyed 47 experts in chemistry on a subset of the benchmark corpus to be able to compare the performance of current frontier models with (human) chemists of different specializations. Our results indicate that current frontier models perform "superhuman" on some aspects of chemistry but, in many cases, including safety-related ones, might be very misleading. We find that there are still numerous limitations for current models to overcome, such that they can be directly applied in autonomous systems for chemists. Moreover, we conclude that carefully created broad benchmarks represent an important stepping stone for progress in this field.

2 Results and Discussion

2.1 Benchmark corpus

To compile our benchmark corpus, we utilized a broad list of sources (see Section 4.1), ranging from university exams to semi-automatically generated questions based on curated subsets of data in chemical databases. For quality assurance, all questions have been reviewed by at least one scientist in addition to the original curator and automated checks. Importantly, our large pool of questions encompasses a wide range of topics and question types. The topics range from general chemistry to more specialized fields such as inorganic, analytical or technical chemistry. We also classify the questions based on what techniques are required to answer them. Here, we distinguish between questions that require knowledge, reasoning, calculation, intuition or a combination of these. Moreover, to allow for a more nuanced evaluation of the models capabilities, the questions are also classified by difficulty.

While many existing benchmarks are designed around multiple-choice question (MCQ), this does not reflect the reality of chemistry education and research.

"Tiny" subset It is important to note that a smaller subset of the corpus might be more practical for routine evaluations.[59] For instance, Liang *et al.* [60] report costs of more than $10,000 for application programming interface (API) calls for a single evaluation on the widely used Holistic Evaluation of Language Models (HELM) benchmark. We also used this subset to seed the web application for the human baseline study.

2.2 Model evaluation

Benchmark suite design Because the text used in scientific settings differs from typical natural language, many models have been developed that deal with such text in a particular way. For instance, the Galactica model[61] uses special tokenization or encoding procedures for molecules and equations. Current benchmarking suites, however, do not account for such special treatment of scientific information. To address this, ChemBench encodes the semantic meaning of various parts of the question or answer. For instance, molecules represented in Simplified Molecular Input Line-Entry System (SMILES) are enclosed in `[START_SMILES][\END_SMILES]` tags. This allows the model to treat the SMILES string differently from other text. ChemBench can seamlessly handle such special treatment in an easily extensible way because the questions are stored in an annotated format.

Since many widely utilized systems only provide access to text completions (and not the raw model outputs), ChemBench is designed to operate on text completions. This is also important given the growing number of tool-augmented systems that are deemed essential for building chemical copilot systems. Such systems can augment the capabilities of LLMs through the use of external tools such as search APIs or code executors.[62–64] In those cases, the LLM that returns the probabilities for various tokens (that are often used for model evaluations[65]) is only a part of the whole system, and it is not clear how to interpret the probabilities in the context of the whole system. The text completions, however, are the system's final outputs, which would also be used in a real-world application. Hence, we use them for our evaluations.

System performance To understand the current capabilities of LLMs in the chemical sciences, we evaluated a wide range of leading models[66] on the ChemBench corpus, including systems augmented with external tools. An overview of the results of this evaluation is shown in ??. In this figure, we show the percentage of questions that the models answered correctly. Moreover, we show the worst, best, and average performance of the human experts in our study, which we obtained via a custom web application (chembench.org) that we used to survey the experts. Remarkably, the figure shows that the leading LLM, Claude 3, outperforms the best human in our study in this overall metric and vastly, by more than a factor of two, exceeds the average performance of the experts in our study. Many other models also outperform the average human performance. Interestingly, the Galactica model, which was trained specifically for scientific applications, underperformed compared to many advanced commercial and some open-source models and barely exceeds the random baseline.

Given the considerable interest in tool-augmented systems, the mediocre performance of these systems (GPT-3.5 and Claude 2 augmented with tools) in our benchmark is striking. Their lack of performance, however, is partially because we limited the system to a maximum of ten LLM calls. With the default tool augmented setup (using the so-called ReAct method[64]), however, this often did not allow the system to identify the correct answer (e.g., because it repeatedly tried to search for the answer on the web and did not find a solution within ten calls to the LLM). This observation highlights the importance of communicating and measuring not only predictive performance but also computational cost (e.g., in terms of API calls) for tool-augmented systems.

Performance per topic To obtain a more detailed understanding of the performance of the models, we also analyzed the performance of the models in different subfields of the chemical sciences. For this analysis, we defined a set of topics (see Section 4.5). We classified all questions in the ChemBench corpus into these topics based on hand-crafted rules and classifier models operating on the question texts. We then computed the percentage of questions the models or humans answered correctly for each topic.

Judging chemical preference One interesting finding of recent research is that foundation models can judge interestingness.[67] If models could do so for chemical compounds, this would open opportunities for novel optimization approaches. Such open-ended tasks, however, depend on an external observer defining what interestingness is.[68] Here, we posed models the same question Choung *et al.* [69] asked chemists at a drug company: “Which of the two compounds do you prefer?” (in the context of an early virtual screening campaign setting). While chemists showed a fair degree of interrater agreement, most of our models do not correlate with the preferences of expert chemists—even if they perform well on many other tasks in

ChemBench. This indicates that using preference tuning for chemical settings is a promising approach to explore in future research.

Confidence estimates One might wonder whether the models can estimate if they can answer a question correctly. If they could do so, incorrect answers would be less problematic as one could detect when an answer is incorrect.

REWRITE

3 Conclusions

On the one hand, our findings underline the impressive capabilities of LLMs in the chemical sciences: Leading models outperform domain experts in specific chemistry questions on many topics. On the other hand, there are still striking limitations. For very relevant topics the answers models provide are wrong. On top of that, many models are not able to reliably estimate their own limitations. Yet, the success of the models in our evaluations perhaps also reveals more about the limitations of the exams we use to evaluate models—and chemists—than about the models themselves. For instance, while models perform well on many textbook questions, they struggle with questions that require some more reasoning. Given that the models outperformed the average human in our study, we need to rethink how we teach and examine chemistry. Critical reasoning is increasingly essential, and rote solving of problems or memorization of facts is a domain in which LLMs will continue to outperform humans.

Our findings also highlight the nuanced trade-off between breadth and depth of evaluation frameworks. The analysis of model performance on different topics shows that models' performance varies widely across the subfields they are tested on. However, even within a topic, the performance of models can vary widely depending on the type of question and the reasoning required to answer it.

The current evaluation frameworks for chemical LLMs are primarily designed to measure the performance of the models on specific property prediction tasks. They cannot be used to evaluate reasoning or systems built for scientific applications. Thus, we had little understanding of the capabilities of LLMs in the chemical sciences. Our work shows that carefully curated benchmarks can provide a more nuanced understanding of the capabilities of LLMs in the chemical sciences. Importantly, our findings also illustrate that more focus is required in developing better human-model interaction frameworks, given that models cannot estimate their limitations.

While our findings indicate many areas for further improvement of LLM-based systems, it is also important to realize that clearly defined metrics have been the key to the progress of many fields of ML, such as computer vision. Although current systems are far from reasoning like a chemist, our ChemBench framework will be a stepping stone for developing systems that might come closer to this goal.

4 Methods

4.1 Curation workflow

For our dataset, we curated questions from existing exams or exercise sheets but also programmatically created new questions. Questions were added via Pull Requests on our GitHub repository and only merged into the corpus after passing manual review (Figure 2) as well as automated checks (e.g., for compliance with a standardized schema).

To ensure that the questions do not enter a training dataset, we use the same canary string as the BigBench project. This requires that LLM developers filter their training dataset for this canary string.[4,44]

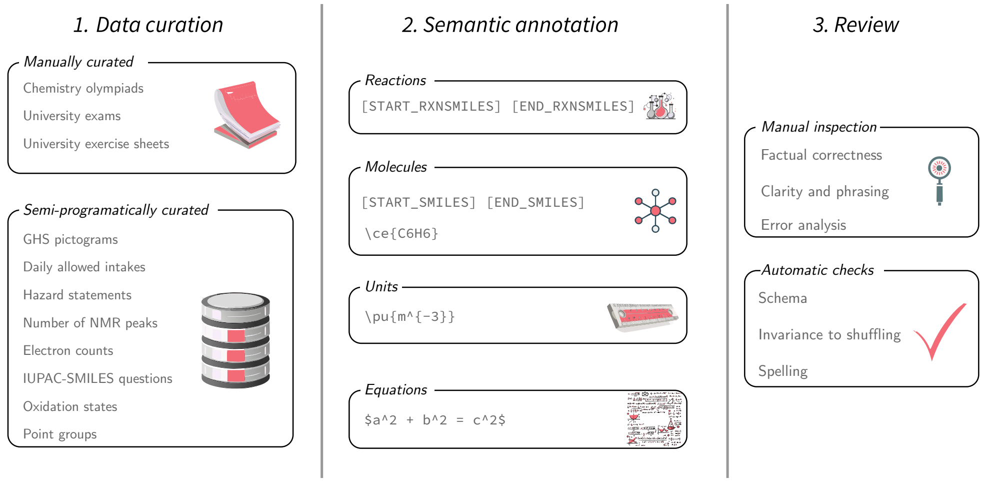

Figure 2: Overview of the workflow for the assembly of the ChemBench corpus. To assemble the ChemBench corpus, we first collected questions from various sources. Some tasks were manually curated, others semi-programmatically. We added semantic annotations for all questions to make them compatible with systems that use special processing for modalities that are not conventional natural text. We reviewed the questions using manual and automatic methods before adding them to the corpus.

Manually curated questions Manually curated questions were sourced from various sources, including university exams, exercises, and question banks. ?? provides an overview of the sources of the manually curated questions.

Semi-programmatically generated questions In addition to the manually curated questions, we also generated questions programmatically. An overview of the sources of the semi-programmatically generated questions is provided in **??**.

4.2 Model evaluation workflow

EXPAND THIS section.

Prompting We employ distinct prompt templates tailored for completion and instruction-tuned models to maintain consistency with the training. As explained below, we impose constraints on the models within these templates to receive responses in a specific format so that robust, fair, and consistent parsing can be performed. Certain models are trained with special annotations and LaTeX syntax for scientific notations, chemical reactions, or symbols embedded within the text. For example, all the SMILES representations are encapsulated within `[START_SMILES][\END_SMILES]` in Galactica[61]. Our prompting strategy consistently adheres to these details in a model-specific manner by post-processing LaTeX syntax, chemical symbols, chemical equations, and physical units (by either adding or removing wrappers). This step can be easily customized in our codebase.

Parsing Our parsing workflow is multistep and primarily based on regular expressions. In the case of instruction-tuned models, we first identify the `[ANSWER][\ANSWER]` environment we prompt the model to report the answer in. In the case of completion models, this step is skipped. From there, we attempt to extract the relevant enumeration letters (for multiple-choice questions) or numbers. In the case of numbers, our regular expression was engineered to deal with various forms of scientific notation. As initial tests indicated that models sometimes return integers in the form of words, e.g., “one” instead of “1”, we also implemented a word-to-number conversion using regular expressions. If these hard-coded parsing steps fail, we use a LLM, e.g., Claude 2, to parse the completion.

Models Rewrite - as table. The table can then also contain if it has been completion or instruction-prompted.

4.3 Confidence estimate

To estimate the models’ confidence, we prompted them with the question (and answer options for MCQ) and the task to rate their confidence to produce the correct answer on a scale from 1 to 5. We decided to use verbalized confidence estimates[70] since we found those closer to current practical use cases than other prompting strategies, which might be more suitable when implemented in systems.

4.4 Human baseline

Question selection Rewrite.

Study design Rewrite.

Participants

Comparison with models For the analysis, we treated each human as a model. We computed the topic aggregated averages per human for analyses grouped by topic and then averaged over all humans.

4.5 Classification of questions into topics

REWRITE.

Data and code availability

The code and data for ChemBench is available at https://github.com/lamalab-org/chem-bench. The code for the app for our human baseline study is available at https://github.com/lamalab-org/chem-bench-app. To ensure reproducibility, this manuscript was generated using the show your work! framework.[71] The code to rebuild the paper (including code for all figures and numbers next to which there is a GitHub icon) can be found at https://github.com/lamalab-org/chembench-paper. To facilitate reproduction, some intermediate analysis results are cached at http://dx.doi.org/10.5072/zenodo.34706.

Acknowledgements

This work was supported by the Carl Zeiss Foundation, a “Talent Fund” of the “Life” profile line of the Friedrich Schiller University Jena. We also thank Stability.AI for the access to its HPC cluster and donations. In addition, M.S-W.’s work was supported by Intel and Merck via the AWASES programme. K.M.J. is part of the NFDI consortium FAIRmat funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project 460197019.

M.A. expresses gratitude to the European Research Council (ERC) for evaluating the project with the reference number 101106377 titled “CLARIFIER” and accepting it for funding under the HORIZON TMA MSCA Postdoctoral Fellowships - European Fellowships. Furthermore, M.A. acknowledges the funding provided by UK Research and Innovation (UKRI) under the UK government’s Horizon Europe funding guarantee (Grant Reference: EP/Y023447/1; Organization Reference: 101106377).

M.R. and U.S.S. thanks the “Deutsche Forschungsgemeinschaft” for funding under the regime of the priority programme SPP 2363 “Utilization and Development of Machine Learning for Molecular Applications – Molecular Machine Learning" (SCHU 1229/63-1; project number 497115849).

In addition, we thank the OpenBioML.org community and their ChemNLP project team for valuable discussions. Moreover, we thank Pepe Márquez for discussions and support and Julian Kimmig for feedback on the web app. In addition, we acknowledge support from Sandeep Kumar with an initial prototype of the web app. We thank Bastian Rieck for developing the LaTeX-credit package (`https://github.com/Pseudomanifold/latex-credits`) and thank Berend Smit for feedback on an early version of the manuscript.

Conflicts of interest

K.M.J. was a paid consultant for OpenAI (as part of the red teaming network). M.P. is an employee of Stability.AI, and A.M. and N.A. were paid contractors of Stability.AI.

Author contributions

	Adrian Mirza	Nawaf Alampara	Sreekanth Kunchapu	Martiño Ríos-García	Benedict Emoekabu	Aswanth Krishnan	Tanya Gupta	Mara Wilhelmi	Macjonathan Okereke	Mehrdad Asgari	Juliane Eberhardt	Amir Mohammad Elahi	Hani M. Elbeheiry	María Victoria Gil	Christina Glaubitz	Maximilian Greiner	Caroline T. Holick	Tim Hoffmann	Abdelrahman Ibrahim	Lea C. Klepsch	Yannik Köster	Fabian Alexander Kreth	Jakob Meyer	Santiago Miret	Jan Matthias Peschel	Michael Ringleb	Nicole Roesner	Johanna Schreiber	Ulrich S. Schubert	Leanne M. Stafast	Dinga Wonanke	Michael Pieler	Philippe Schwaller	Kevin Maik Jablonka
Conceptualization	■	■	■	■																												■	■	■
Data curation	■	■	■	■	■		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		■	■		■	■
Formal analysis	■	■		■																														■
Funding acquisition																													■			■	■	■
Investigation	■	■	■	■	■		■	■	■					■																			■	■
Methodology																																■	■	■
Project administration																																		■
Resources																																■		■
Software	■	■		■		■																												■
Supervision																																	■	■
Validation	■	■	■	■																														■
Visualization	■																																	■
Writing – original draft																																		■
Writing – review & editing	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

References

1. Brown, T. B. *et al.* Language Models are Few-Shot Learners. arXiv: `2005.14165 [cs.CL]` (May 28, 2020).
2. Zhong, Z., Zhou, K. & Mottin, D. Benchmarking Large Language Models for Molecule Prediction Tasks. arXiv: `2403.05075 [cs.LG]` (2024).

3. Kung, T. H. *et al.* Performance of ChatGPT on USMLE: potential for AI-assisted medical education using large language models. *PLoS digit. health* **2,** e0000198 (2023).

4. OpenAI *et al.* GPT-4 Technical Report. arXiv: `2303.08774 [cs.CL]` (Mar. 15, 2024).

5. Boiko, D. A., MacKnight, R., Kline, B. & Gomes, G. Autonomous chemical research with large language models. *Nature* **624,** 570–578 (Dec. 20, 2023).

6. Bran, A. M., Cox, S., White, A. D. & Schwaller, P. ChemCrow: Augmenting large-language models with chemistry tools. arXiv: `2304.05376 [physics.chem-ph]` (2023).

7. Bubeck, S. *et al.* Sparks of Artificial General Intelligence: Early experiments with GPT-4. arXiv: `2303.12712 [cs.CL]` (2023).

8. Bender, E. M., Gebru, T., McMillan-Major, A. & Shmitchell, S. *On the dangers of stochastic parrots: Can language models be too big?* in *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency* (2021), 610–623.

9. Bommasani, R. *et al.* On the Opportunities and Risks of Foundation Models. arXiv: `2108.07258 [cs.LG]` (Aug. 16, 2021).

10. Anderljung, M. *et al.* Frontier AI regulation: Managing emerging risks to public safety. arXiv: `2307.03718 [cs.CY]` (July 6, 2023).

11. AI4Science, M. R. & Quantum, M. A. The Impact of Large Language Models on Scientific Discovery: a Preliminary Study using GPT-4. arXiv: `2311.07361 [cs.CL]` (2023).

12. Schmitt, B. Transforming qualitative research in phygital settings: the role of generative AI. *Qualitative Market Research: An International Journal* **27,** 523–526. ISSN: 1352-2752. `http://dx.doi.org/10.1108/QMR-08-2023-0107` (Dec. 2023).

13. White, A. D. The future of chemistry is language. *Nat. Rev. Chem.* **7,** 457–458 (May 19, 2023).

14. Jablonka, K. M. *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).

15. Jablonka, K. M., Schwaller, P., Ortega-Guerrero, A. & Smit, B. Leveraging large language models for predictive chemistry. *Nat. Mach. Intell.,* 1–9 (2024).

16. Xie, Z. *et al.* Fine-tuning GPT-3 for machine learning electronic and functional properties of organic molecules. *Chem. Sci.* **15,** 500–510 (2024).

17. Liao, C., Yu, Y., Mei, Y. & Wei, Y. From Words to Molecules: A Survey of Large Language Models in Chemistry. arXiv: `2402.01439 [cs.LG]` (2024).

18. Zhang, D. *et al.* ChemLLM: A Chemical Large Language Model. *arXiv preprint arXiv:2402.06852.* arXiv: 2402.06852 [cs.AI] (2024).

19. Ramos, M. C., Michtavy, S. S., Porosoff, M. D. & White, A. D. Bayesian Optimization of Catalysts With In-context Learning. arXiv: 2304.05341 [physics.chem-ph] (2023).

20. Kristiadi, A. *et al.* A Sober Look at LLMs for Material Discovery: Are They Actually Good for Bayesian Optimization Over Molecules? arXiv: 2402.05015 [cs.LG] (2024).

21. Rubungo, A. N., Arnold, C., Rand, B. P. & Dieng, A. B. Llm-prop: Predicting physical and electronic properties of crystalline solids from their text descriptions. *arXiv preprint arXiv:2310.14029* (2023).

22. Flam-Shepherd, D. & Aspuru-Guzik, A. Language models can generate molecules, materials, and protein binding sites directly in three dimensions as XYZ, CIF, and PDB files. arXiv: 2305.05708 (2023).

23. Gruver, N. *et al.* Fine-Tuned Language Models Generate Stable Inorganic Materials as Text. arXiv: 2402.04379 (2024).

24. Patiny, L. & Godin, G. Automatic extraction of FAIR data from publications using LLM. *ChemRxiv preprint doi:10.26434/chemrxiv-2023-05v1b-v2* (Dec. 2023).

25. Dagdelen, J. *et al.* Structured information extraction from scientific text with large language models. *Nature Communications* **15.** ISSN: 2041-1723. http://dx.doi.org/10.1038/s41467-024-45563-x (Feb. 2024).

26. Zheng, Z. *et al.* Image and data mining in reticular chemistry powered by GPT-4V. *Digital Discovery* **3,** 491–501. ISSN: 2635-098X. http://dx.doi.org/10.1039/d3dd00239j (2024).

27. Lála, J. *et al.* PaperQA: Retrieval-Augmented Generative Agent for Scientific Research. arXiv: 2312.07559 [cs.CL] (2023).

28. Caufield, J. H. *et al.* Structured prompt interrogation and recursive extraction of semantics (SPIRES): A method for populating knowledge bases using zero-shot learning. arXiv: 2304.02711 [cs.AI] (2023).

29. Gupta, T., Zaki, M., Krishnan, N., *et al.* DiSCoMaT: distantly supervised composition extraction from tables in materials science articles. *arXiv preprint arXiv:2207.01079.* arXiv: 2207.01079 (2022).

30. Darvish, K. *et al.* ORGANA: A Robotic Assistant for Automated Chemistry Experimentation and Characterization. arXiv: 2401.06949 [cs.RO] (2024).

31. Miret, S. & Krishnan, N. Are LLMs Ready for Real-World Materials Discovery? arXiv: 2402.05200 (2024).

32. Granda, J. M., Donina, L., Dragone, V., Long, D.-L. & Cronin, L. Controlling an organic synthesis robot with machine learning to search for new reactivity. *Nature* **559,** 377–381 (July 2018).

33. Angello, N. H. *et al.* Closed-loop optimization of general reaction conditions for heteroaryl Suzuki-Miyaura coupling. *Science* **378,** 399–405 (Oct. 28, 2022).

34. Coley, C. W. *et al.* A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* **365,** eaax1566 (Aug. 9, 2019).

35. Burger, B. *et al.* A mobile robotic chemist. *Nature* **583,** 237–241 (July 8, 2020).

36. Seifrid, M. *et al.* Autonomous chemical experiments: Challenges and perspectives on establishing a self-driving lab. *Acc. Chem. Res.* **55,** 2454–2466 (2022).

37. Gopal, A. *et al.* Will releasing the weights of future large language models grant widespread access to pandemic agents? arXiv: `2310.18233 [cs.AI]` (Oct. 25, 2023).

38. Ganguli, D. *et al.* Red Teaming Language Models to Reduce Harms: Methods, Scaling Behaviors, and Lessons Learned. arXiv: `2209.07858 [cs.CL]` (Aug. 23, 2022).

39. Urbina, F., Lentzos, F., Invernizzi, C. & Ekins, S. Dual use of artificial-intelligence-powered drug discovery. *Nat. Mach. Intell.* **4,** 189–191 (Mar. 7, 2022).

40. Campbell, Q. L., Herington, J. & White, A. D. Censoring chemical data to mitigate dual use risk. `https://arxiv.org/abs/2304.10510` (2023).

41. Moulange, R., Langenkamp, M., Alexanian, T., Curtis, S. & Livingston, M. Towards Responsible Governance of Biological Design Tools. `https://arxiv.org/abs/2311.15936` (2023).

42. Urbina, F., Lentzos, F., Invernizzi, C. & Ekins, S. A teachable moment for dual-use. *Nat. Mach. Intell.* **4,** 607–607 (July 12, 2022).

43. Intelligent.com. *One-third of college students used CHATGPT for schoolwork during the 2022-23 academic date* https://www.intelligent.com/one-third-of-college-students-used-chatgpt-for-schoolwork-during-the-2022-23-academic-date/. Oct. 2023.

44. Srivastava, A. *et al.* Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. `https://arxiv.org/abs/2206.04615` (2022).

45. Gao, L. *et al. A framework for few-shot language model evaluation* version v0.4.0. Dec. 2023. `https://zenodo.org/records/10256836`.

46. Guo, T. *et al.* What can Large Language Models do in chemistry? A comprehensive benchmark on eight tasks. arXiv: `2305.18365 [cs.CL]` (May 27, 2023).

47. Ahmad, W., Simon, E., Chithrananda, S., Grand, G. & Ramsundar, B. ChemBERTa-2: Towards Chemical Foundation Models. arXiv: `2209.01712 [cs.LG]` (Sept. 5, 2022).

48. Cai, X. *et al.* Comprehensive evaluation of molecule property prediction with ChatGPT. *Methods* **222,** 133–141 (Feb. 2024).

49. Frey, N. C. *et al.* Neural scaling of deep chemical models. *Nat. Mach. Intell.* **5,** 1297–1305 (Oct. 23, 2023).

50. Dinh, T. *et al.* Lift: Language-interfaced fine-tuning for non-language machine learning tasks. *Adv. Neur. In.* **35,** 11763–11784 (2022).

51. Wu, Z. *et al.* MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* **9,** 513–530 (2018).

52. Huang, K. *et al.* Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. arXiv: `2102.09548v2 [cs.LG]` (Feb. 18, 2021).

53. Dunn, A., Wang, Q., Ganose, A., Dopp, D. & Jain, A. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Comp. Mater.* **6** (Sept. 2020).

54. Zaki, M., Jayadeva, Mausam & Krishnan, N. M. A. MaScQA: investigating materials science knowledge of large language models. *Digit. Discov.* **3,** 313–327 (2024).

55. Arora, D., Singh, H. G. & Mausam. Have LLMs Advanced Enough? A Challenging Problem Solving Benchmark For Large Language Models. arXiv: `2305.15074 [cs.CL]` (May 24, 2023).

56. Song, Y., Miret, S., Zhang, H. & Liu, B. HoneyBee: Progressive Instruction Finetuning of Large Language Models for Materials Science. arXiv: `2310.08511 [cs.CL]` (Oct. 12, 2023).

57. Wei, Z. *et al.* *ChemistryQA: A Complex Question Answering Dataset from Chemistry* 2021. `https://openreview.net/forum?id=oeHTRAehiFF`.

58. Song, Y., Miret, S. & Liu, B. *MatSci-NLP: Evaluating Scientific Language Models on Materials Science Language Tasks Using Text-to-Schema Modeling* in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (eds Rogers, A., Boyd-Graber, J. & Okazaki, N.) (Association for Computational Linguistics, Toronto, Canada, July 2023), 3621–3639.

59. Polo, F. M. *et al.* tinyBenchmarks: evaluating LLMs with fewer examples. arXiv: `2402.14992 [cs.CL]` (Feb. 22, 2024).

60. Liang, P. *et al.* Holistic Evaluation of Language Models. arXiv: `2211.09110 [cs.CL]` (Nov. 16, 2022).

61. Taylor, R. *et al.* Galactica: A Large Language Model for Science. arXiv: 2211.09085 [cs.CL] (Nov. 16, 2022).

62. Schick, T. *et al.* Toolformer: Language models can teach themselves to use tools. *Adv. Neur. In.* **36** (2024).

63. Karpas, E. *et al.* MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. arXiv: 2205.00445 [cs.CL] (2022).

64. Yao, S. *et al.* React: Synergizing reasoning and acting in language models. arXiv: 2210.03629 [cs.CL] (2022).

65. Fourrier, C., Habib, N., Launay, J. & Wolf, T. *What's going on with the open llm leaderboard?* https://huggingface.co/blog/open-llm-leaderboard-mmlu.

66. Beeching, E. *et al. Open LLM Leaderboard* https://huggingface.co/spaces/HuggingFaceH4/open_llm _ leaderboard. 2023.

67. Zhang, J., Lehman, J., Stanley, K. & Clune, J. *OMNI: Open-endedness via Models of human Notions of Interestingness* 2024. arXiv: 2306.01711 [cs.AI]. https://arxiv.org/abs/2306.01711.

68. Hughes, E. *et al. Open-Endedness is Essential for Artificial Superhuman Intelligence* 2024. arXiv: 2406.04268 [cs.LG]. https://arxiv.org/abs/2406.04268.

69. Choung, O.-H., Vianello, R., Segler, M., Stiefl, N. & Jiménez-Luna, J. Extracting medicinal chemistry intuition via preference machine learning. *Nature Communications* **14.** ISSN: 2041-1723. http://dx.doi.org/10.1038/s41467-023-42242-1 (Oct. 2023).

70. Xiong, M. *et al.* Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. arXiv: 2306.13063 [cs.CL] (2023).

71. Luger, R. *et al.* Mapping stellar surfaces III: An Efficient, Scalable, and Open-Source Doppler Imaging Model, arXiv:2110.06271. arXiv: 2110.06271 [astro-ph.SR] (Oct. 12, 2021).

A Appendix

A.1 Desired properties of a chemistry benchmark

- *End-to-end automation.* For model development, the evaluations must be run many times (e.g., on regular intervals of a training run). Approaches that rely on humans scoring the answers of a system[1–3] can thus not be used.

- *Careful validation by experts.* Manual curation is needed to minimize the number of incorrect or unanswerable questions.[4] This is motivated by the observation that many widely used benchmarks are plagued by noisiness.[5,6]

- *Usable with models that support special treatment of molecules.* Some models, such as Galactica[7], use special tokenization or encoding procedures for molecules or equations. The benchmark system must encode the semantic meaning of various parts of the question or answer to support this.

- *Usable with black box systems.* Many relevant systems do not provide access to model weights or raw logits. This might be the case because the systems are proprietary or because they involve not only LLMs but also external tools such as search APIs or code executors.[8–10] Thus, a benchmark should not assume access to the raw model outputs but be able to operate on text completions.

- *Probing capabilities beyond answering of MCQs.* In real-world chemistry, as well as higher-level university education, multiple-choice questions are seldom utilized. Yet, most benchmarking frameworks focus on the MCQ setting because of the ease of evaluation. Realistic evaluations must measure capabilities beyond answering MCQ.

- *Cover a diverse set of topics.* Chemistry, as the "central science", bridges multiple disciplines.[11] To even just approximate "chemistry capabilities" the topics covered by a chemistry benchmark must be very diverse.

A.2 Related work

Existing benchmarks such as those from Guo *et al.* [12], Sun *et al.* [13], Schulze Balhorn *et al.* [1], Cai *et al.* [14] fail to comply with most of the requirements stipulated above. While these benchmarks could provide valuable insights in the short term, they cannot follow the rapid additions to the LLM space. ChemBench aims to correct this through a set of developments: compatibility with BigBench, end-to-end automation, a particular focus on chemical safety, employment of diverse prompting strategies, and specialized notation for molecules and mathematical symbols. Moreover, our robust framework, including the platform `chembench.org`, will engage the community in open-source contributions.

A.3 Benchmark corpus

To ensure maximal interoperability with existing benchmarks or tools, we curated the data in an extended form of the widely used BigBench format.[15] This also implies that future baselines can be built on top of our infrastructure if saved in the same format.

?? shows the distribution of the Flesch-Kincaid reading ease scores of the questions. We see that the questions are generally complex to read.

?? shows that most questions in our corpus are MCQ. A substantial fraction, in contrast to other benchmarks, is open-ended.

A.4 Model performance

We also evaluated the model performance on the entire ChemBench corpus. **??** shows the fraction of questions that were answered completely correctly by the models. Note that this ranking differs from the one on the "tiny" subset.

?? shows the performance of the models on the different topics of the ChemBench corpus. The general pattern of performance varies significantly between the different topics and is also observed when the models are evaluated on the entire corpus. However, since some subjects are composed of questions from different sources, the ranking of the models is, in some instances, different from the one on the "tiny" subset.

?? shows this data as a parallel coordinates plot. This visualization highlights the critical observation that the ranking of the models highly depends on the questions they are evaluated on. Only very broad benchmarks have the potential to provide a comprehensive view of a model's capabilities. However, even in those cases, the weighting of the different topics is crucial. Hence, we believe that fine-grained analysis of model performance is vital for the development of future benchmarks.

To further investigate the performance of the models, we also compared the performance on different data sources. Compared to topics, this is a more fine-grained analysis, as topics can be composed of questions from different sources. In Figure 3, we see that the performance of the models varies significantly between the different data sources. Interestingly, the performance of the models on questions sourced based on textbooks seems to be better for our models than some semi-programmatically created tasks, such as questions about the number of signals in an Nuclear Magnetic Resonance (NMR) spectrum.

?? shows the same analysis on the "tiny" subset.

One might wonder if questions that are more difficult to parse lead to worse performance of the models. ?? shows no clear correlation between the reading ease of the questions and the performance of the models.

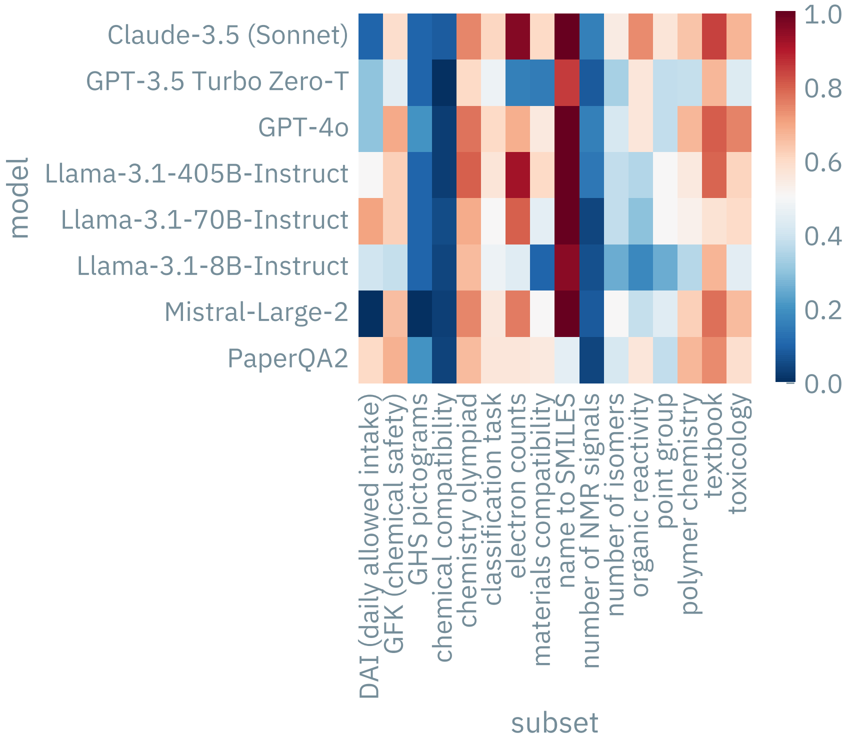

Figure 3: Fraction of completely correctly answered questions per data source. The heatmap shows, in color, the fraction of questions answered completely correctly by different systems for some of our data sources. The performance is measured as the fraction of questions answered completely correctly by the models. A score of one (red) indicates that all questions were answered completely correctly, while a score of zero (blue) indicates that none of the questions were answered completely correctly. We see that the performance of the models varies significantly between the different data sources. For instance, it is interesting to observe that questions sourced based on textbooks seem easier for our leading models than for humans. However, this performance does not correlate with performance on other sources, e.g., semi-programmatically created tasks such as questions about the number of signals in an NMR spectrum.

A.5 Performance as a function of molecular features

To better understand if the performance of the models is correlated with specific features of the molecules, we analyzed the performance of the models as a function of the number of atoms and the complexity of the molecules. **??** shows that the performance of the models is not correlated with the complexity of the molecules but rather with the number of atoms (**??**, similar trivial correlation for **??**). The corresponding Spearman correlation coefficients are listed in **??**.

A.6 Influence of model scale

REWRITE.

A.7 Human baseline

App To facilitate the collection of responses, we developed a responsive web application in Typescript using the Next.js[16] app router framework. This application handles serving the user interface and exposes various Representational State Transfer (REST) APIs for relevant operations. We utilize a Postgresql. The web application is styled with Tailwind CSS[17] using the shadcn/ui component library and uses NextAuth[18] for easy and secure user authentication. The application is hosted on the Vercel web hosting platform.

Statistics **??** shows the distribution of scores our human scorers achieved.

We also recorded the time humans took to answer the questions. This time is the time from the question being displayed to the human to the human submitting the answer.

Additionally, we prompted users to provide additional information about their experience in chemistry. While we recorded fine-grained information, e.g., their specialization, we focused on the number of years since the first university-level chemistry course. Figure 5 shows that the experience of the human scorers was correlated with the correctness of their answers (Figure 5, Spearman's $\rho \approx 0.16$, and $p \approx 0.39$).

A.8 Confidence estimates

Since it is important to understand if models can provide an indication of whether their answer might likely be incorrect, we prompted some of our top performing LLMs to return the confidence in providing a correct answer on an ordinal scale. This is similar to the verbalized confidence scores reported by Xiong *et al.* [19]. **??** plots the distribution of those scores. We find that the models show different distributions of confidence scores, which, for some, are skewed to the extremes.

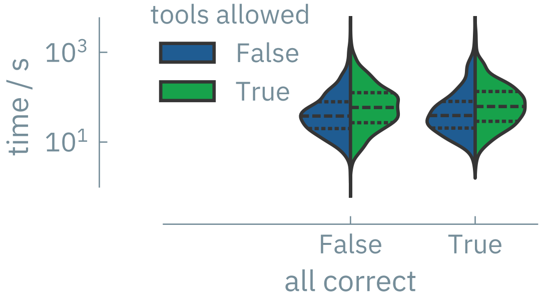

Figure 4: Time taken by human scorers to answer questions vs. correctness of their answers. From the plot, it is clear that there is no clear dependence of the correctness of the answers on the time taken by the human scorers to answer the questions. However, we see that human scorers typically took longer to correctly answer questions with tool use.

A.9 Leaderboard

Our leaderboard is based on the tool chain developed for Matbench.[20] Briefly, the ChemBench pipeline produces standardized files in `json` format that contributors can add via pull requests to the ChemBench repository. The Markdown tables and interactive plots are automatically generated and updated on the ChemBench website.

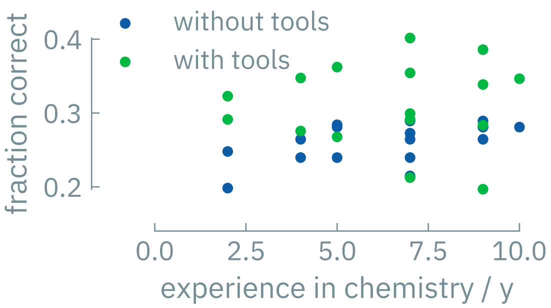

Figure 5: Experience of human scorers vs. correctness of their answers. The experience (in the number of years since the first university-level chemistry course) of the human scorers wasp correlated with the correctness of their answers.

Acronyms

AGI artificial general intelligence.

API application programming interface.

HELM Holistic Evaluation of Language Models.

LLM large language model.

MCQ multiple-choice question.

ML machine learning.

NMR Nuclear Magnetic Resonance.

REST Representational State Transfer.

SMILES Simplified Molecular Input Line-Entry System.

References

1. Schulze Balhorn, L. *et al.* Empirical assessment of ChatGPT's answering capabilities in natural science and engineering. *Sci. Rep.* **14** (Feb. 2024).
2. AI4Science, M. R. & Quantum, M. A. The Impact of Large Language Models on Scientific Discovery: a Preliminary Study using GPT-4. arXiv: `2311.07361 [cs.CL]` (2023).
3. Castro Nascimento, C. M. & Pimentel, A. S. Do Large Language Models Understand Chemistry? A Conversation with ChatGPT. *J. Chem. Inf. Model.* **63,** 1649–1655 (Mar. 16, 2023).
4. Northcutt, C. G., Athalye, A. & Mueller, J. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. arXiv: `2103.14749 [stat.ML]` (Mar. 26, 2021).
5. Frye, C. *PubMedQA noisy* Dec. 2023. `https://twitter.com/charles%5C_irl/status/1731854677711507650`.
6. Awg. *Broken benchmark: MMLU* `https://www.lesswrong.com/posts/rQBaftqKMfG2uMiWb/broken-benchmark-mmlu`.
7. Taylor, R. *et al.* Galactica: A Large Language Model for Science. arXiv: `2211.09085 [cs.CL]` (Nov. 16, 2022).
8. Schick, T. *et al.* Toolformer: Language models can teach themselves to use tools. *Adv. Neur. In.* **36** (2024).
9. Karpas, E. *et al.* MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. arXiv: `2205.00445 [cs.CL]` (2022).
10. Yao, S. *et al.* React: Synergizing reasoning and acting in language models. arXiv: `2210.03629 [cs.CL]` (2022).
11. Aspuru-Guzik, A., Lindh, R. & Reiher, M. The Matter Simulation (R)evolution. *ACS Cent. Sci.* **4,** 144–152 (Feb. 6, 2018).
12. Guo, T. *et al.* What can Large Language Models do in chemistry? A comprehensive benchmark on eight tasks. arXiv: `2305.18365 [cs.CL]` (May 27, 2023).
13. Sun, L. *et al.* SciEval: A Multi-Level Large Language Model Evaluation Benchmark for Scientific Research. arXiv: `2308.13149 [cs.CL]` (2023).
14. Cai, X. *et al.* Comprehensive evaluation of molecule property prediction with ChatGPT. *Methods* **222,** 133–141 (Feb. 2024).
15. Srivastava, A. *et al.* Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. `https://arxiv.org/abs/2206.04615` (2022).

16. Vercel. *nextjs* `https://nextjs.org/`.

17. tailwindcss. *Tailwind CSS* `https://tailwindcss.com/`.

18. NextAuth.js. *NextAuth.js* `https://next-auth.js.org/`.

19. Xiong, M. *et al.* Can LLMs Express Their Uncertainty? An Empirical Evaluation of Confidence Elicitation in LLMs. arXiv: `2306.13063 [cs.CL]` (2023).

20. Dunn, A., Wang, Q., Ganose, A., Dopp, D. & Jain, A. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Comp. Mater.* **6** (Sept. 2020).