Comment on "Comparing the Performance of College Chemistry Students with ChatGPT for Calculations Involving Acids and Bases"

Joshua Schrier*

Department of Chemistry, Fordham University, 441 E. Fordham Road, The Bronx, New York 10458, USA

jschrier@fordham.edu

Abstract

In a recent paper in this Journal (*J. Chem. Educ.* 2023, 100, 3934-3944), Clark *et al.* evaluated the performance of the GPT-3.5 large language model (LLM) on ten undergraduate pH calculation problems. They reported that GPT-3.5 gave especially poor results for salt and titration problems, returning the correct results only 10% and 0% of the time, respectively, and that despite a correct application of heuristics, the LLM made mathematical errors and used flawed strategies. However, these problems are *partially* mitigated using the more advanced GPT-4 model, and *entirely corrected* using simple prompting and calculator tool use patterns demonstrated herein.

Keywords:

General Public, High-School/Introductory Chemistry, First-Year Undergraduate/General, Second-year Undergraduate, Internet/Web-Based Learning, Misconceptions, Problem-Solving/Decision Making, Testing/Assessment, Acids/Bases, pH, Generative AI, Large Language Models, ChatGPT, GPT-4

Recently in this Journal, Clark *et al.* identified deficiencies in the performance of the GPT-3.5 large language model (LLM) on undergraduate acid-base calculation problems that limit its educational applications.[1] To investigate whether their critiques remain relevant to the GPT-4 model (released November 2023), Mathematica 14.0[2] was used to programmatically access the `gpt-4-1106-preview` model and define prompts and tools (*vide infra*), and applied to the two most difficult questions reported by Clark *et al.* for the purpose of illustration—pH calculations of a salt solution and following titration of a weak base by a strong acid. The model's temperature was set to 1, resulting in different outputs in each trial, so following Clark *et al.*, each query was repeated 20 times to sample these output variations. A complete computational notebook containing all necessary code and results is provided online.[3] In principle, similar results could be obtained following Clark *et al.*'s process of typing the entries into the web browser-based ChatGPT Plus;[4] programming merely avoids tedious data entry.

GPT-4 outperforms both GPT-3.5 and human students. GPT-4 has better performance than GPT-3.5 in a variety of tasks,[5] including chemistry,[6] so one expects similar improvements for these acid-base problems. The same initial prompt was provided for both questions, written to facilitate automatic grading: `Your goal is to answer a QUESTION. When you are complete, report the final numerical answer in the form "ANSWER: number".` After that, the question was provided. The salt problem was provided as: `QUESTION: Calculate the pH of 0.25 M NH4Cl. Kb for NH3 = 1.8 * 10^-5.` The titration problem was provided as: `QUESTION: If 25.0 mL of 0.25 M HNO3 is combined with 15.0 mL of 0.25 M CH3NH2, what is the pH? Kb for CH3NH2 = 4.38 * 10^-4.` For the salt problem, GPT-4 returned the correct answer 12/20 = 60% of the time, a marked improvement over 10% and 32% correct rates reported by Clark et al. for GPT-3.5 and general chemistry students, respectively. For the titration problem, GPT-4 is correct 15/20 = 75% of the time, again a

marked improvement over the 0% and 18% they report for GPT-3.5 and general chemistry students. As observed by Clark *et al.*, incorrect GPT-4 answers often arose from calculation errors. Numerical calculation errors are a common, and expected, failure mode of any LLM,[7] which is rectified by providing external *tool* programs.[8,9] Tools (also referred to as *functional calling*, or *application programming interface (API) invocation*, or *plugins*) extend an LLM by providing the ability to call an external program which may run on the user's local computer or on a network-accessible web service. Example tools might perform a web-search or evaluate computer code and return the corresponding results.

Tool use and self-reflection prompting enable GPT-4 to achieve perfect results. Just as one would give a student a calculator to perform the task, one can provide the LLM with a calculator function tool which takes textual inputs (i.e., digit characters and other function to be performed, e.g., `+`, `Log`), computes the result, and returns it to the LLM for further use. Users of the web-based ChatGPT Plus might use the WolframAlpha plugin[10] or the (python) Code Interpreter for this purpose; for conceptual rigor, the example notebook defines a function that can *only* perform numerical calculations, to dispel criticisms that the LLM is "cheating". Merely providing a tool is insufficient; one must also instruct the LLM to make a plan and check its work. This is as simple as writing a text prompt that combines chain-of-thought ("Think step by step…") and self-reflective ("Before you return the final answer, evaluate whether it is reasonable…") instructions.[11] This prompt is not unlike how one teaches students to define a strategy and evaluate if their answer is plausible (e.g., the salt of a strong base should be a weak acid, so a basic pH is suspicious). Exploratory investigations described in the computational notebook indicated that neither tool-only nor agent-prompting-only strategies were sufficient on their own; they needed to be combined. The following initial prompt yields 100% correct

answers for both questions: `Your goal is to answer a QUESTION. Think step by step and use the available tool to perform any numerical calculations. Whenever you need to perform a numerical calculation use the provided tool. Remember that the correct expression for calculating the pH from c = the H3O+ concentration is "-Log[10, c]", because pH is calculated as Log-base-10, whereas the default Wolfram command is a natural logarithm. Before you return the final answer, evaluate whether it is a reasonable answer. If it is not a reasonable answer, repeat the calculation. When you are complete, report the final numerical answer in the form "ANSWER: number".` In summary, a prompt containing the same advice one would offer a struggling student—bring a calculator, think before using it, and consider if the result is reasonable—enables GPT-4 to achieve a perfect score on these questions.

How to write an effective prompt. Very general prompts, like the one described above, may suffice with only minor modification for solving a variety of chemistry problems using the current generation of LLMs. Indeed, for the problems considered here, the only chemistry-specific prompt content was a reminder on the proper way to convert H_3O^+ concentration to pH using the provided general-purpose calculator tool; even this might be unnecessary if a dedicated "compute pH from concentration" tool was provided. In general, prompts should provide specific, positive instructions for the intended output—the role or context ("Your goal is…"), output format ("report the final numerical answer in the form…"), and task specification (e.g., show step-by-step reasoning, use a calculator for certain tasks, etc.). For more challenging problems, one can also provide relevant worked examples, additional information, or define restrictions to guide the output towards the desired.

Implications for teaching and learning. The problems and errors of LLM outputs discussed by Clark *et al.* are only true for their specific choice of GPT-3.5; these limitations are

mostly resolved by using a better model, and *entirely* resolved by using an improved promptand calculator tool. Teachers should assume that students *already* have access to this functionality; eventually it will be the default. This significantly modifies Clark *et al.*'s conclusions. First, the integrity of unproctored examinations is far more dire than they suggest, as the strategy demonstrated above gives perfect results. Second, unless deliberated hindered, the LLM outputs are always correct, and thus no longer provide opportunities for student critique. Alternatively, the ability to generate *reliable*, human-readable explanations on demand means AI is more generally applicable than Clark *et al.* suggest. Third, there is little doubt that AI in general,[12,13] and LLMs in particular,[6,14] provide a powerful new tool for chemistry, so teaching its effective use is imperative. The procedure described here—providing tools and structured reasoning prompts—is already being used by ChemCrow[15] and AI-Coscientist[16] to direct chemical research. Finally, these results increase the urgency of Clark *et al.*'s call for further education research on the use of AI. More broadly, instructors should rethink the purpose and contents of introductory chemistry courses, automating tedious calculation and emphasizing higher level conceptual and problem-solving skills.

Acknowledgements

The author thanks Alexander J. Norquist for critical comments on the manuscript, Fordham University for granting a Faculty Fellowship during the 2023-2024 academic year to study LLMs, and the National Science Foundation for support under grants PHY-2226511 and CNS-2018427.

Data Availability

A Mathematica 13.3 compatible notebook (created in Mathematica 14.0) is available online at https://www.wolframcloud.com/obj/bfedbaff-c160-458f-b779-86b1d1e8691f [To be replaced with a notebook archive link in page proofs]

Notes

The author declares no conflict of interest.

References

(1) Clark, T. M.; Anderson, E.; Dickson-Karn, N. M.; Soltanirad, C.; Tafini, N. Comparing the Performance of College Chemistry Students with ChatGPT for Calculations Involving Acids and Bases. *J. Chem. Educ.* **2023**, *100* (10), 3934–3944. https://doi.org/10.1021/acs.jchemed.3c00500.

(2) Wolfram Research. Mathematica, Version 14.0. https://www.wolfram.com/mathematica.

(3) Schrier, Joshua. *Supporting Information for: Comment on "Comparing the Performance of College Chemistry Students with ChatGPT for Calculations Involving Acids and Bases."* Wolfram Cloud. https://www.wolframcloud.com/obj/bfedbaff-c160-458f-b779-86b1d1e8691f (accessed 2024-01-18).

(4) *Introducing ChatGPT Plus*. https://openai.com/blog/chatgpt-plus (accessed 2024-01-18).

(5) OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; Avila, R.; Babuschkin, I.; Balaji, S.; Balcom, V.; Baltescu, P.; Bao, H.; Bavarian, M.; Belgum, J.; Bello, I.; Berdine, J.; Bernadett-Shapiro, G.; Berner, C.; Bogdonoff, L.; Boiko, O.; Boyd, M.; Brakman, A.-L.; Brockman, G.; Brooks, T.; Brundage, M.; Button, K.; Cai, T.; Campbell, R.; Cann, A.; Carey, B.; Carlson, C.; Carmichael, R.; Chan, B.; Chang, C.; Chantzis, F.; Chen, D.; Chen, S.; Chen, R.; Chen, J.; Chen, M.; Chess, B.; Cho, C.; Chu, C.; Chung, H. W.; Cummings, D.; Currier, J.; Dai, Y.; Decareaux, C.; Degry, T.; Deutsch, N.; Deville, D.; Dhar, A.; Dohan, D.; Dowling, S.; Dunning, S.; Ecoffet, A.; Eleti, A.; Eloundou, T.; Farhi, D.; Fedus, L.; Felix, N.; Fishman, S. P.; Forte, J.; Fulford, I.; Gao, L.; Georges, E.; Gibson, C.; Goel, V.; Gogineni, T.; Goh, G.; Gontijo-Lopes, R.; Gordon, J.; Grafstein, M.; Gray, S.; Greene, R.; Gross, J.; Gu, S. S.; Guo, Y.; Hallacy, C.; Han, J.; Harris, J.; He, Y.; Heaton, M.;

Heidecke, J.; Hesse, C.; Hickey, A.; Hickey, W.; Hoeschele, P.; Houghton, B.; Hsu, K.; Hu, S.; Hu, X.; Huizinga, J.; Jain, S.; Jain, S.; Jang, J.; Jiang, A.; Jiang, R.; Jin, H.; Jin, D.; Jomoto, S.; Jonn, B.; Jun, H.; Kaftan, T.; Kaiser, Ł.; Kamali, A.; Kanitscheider, I.; Keskar, N. S.; Khan, T.; Kilpatrick, L.; Kim, J. W.; Kim, C.; Kim, Y.; Kirchner, H.; Kiros, J.; Knight, M.; Kokotajlo, D.; Kondraciuk, Ł.; Kondrich, A.; Konstantinidis, A.; Kosic, K.; Krueger, G.; Kuo, V.; Lampe, M.; Lan, I.; Lee, T.; Leike, J.; Leung, J.; Levy, D.; Li, C. M.; Lim, R.; Lin, M.; Lin, S.; Litwin, M.; Lopez, T.; Lowe, R.; Lue, P.; Makanju, A.; Malfacini, K.; Manning, S.; Markov, T.; Markovski, Y.; Martin, B.; Mayer, K.; Mayne, A.; McGrew, B.; McKinney, S. M.; McLeavey, C.; McMillan, P.; McNeil, J.; Medina, D.; Mehta, A.; Menick, J.; Metz, L.; Mishchenko, A.; Mishkin, P.; Monaco, V.; Morikawa, E.; Mossing, D.; Mu, T.; Murati, M.; Murk, O.; Mély, D.; Nair, A.; Nakano, R.; Nayak, R.; Neelakantan, A.; Ngo, R.; Noh, H.; Ouyang, L.; O'Keefe, C.; Pachocki, J.; Paino, A.; Palermo, J.; Pantuliano, A.; Parascandolo, G.; Parish, J.; Parparita, E.; Passos, A.; Pavlov, M.; Peng, A.; Perelman, A.; Peres, F. de A. B.; Petrov, M.; Pinto, H. P. de O.; Michael; Pokorny; Pokrass, M.; Pong, V.; Powell, T.; Power, A.; Power, B.; Proehl, E.; Puri, R.; Radford, A.; Rae, J.; Ramesh, A.; Raymond, C.; Real, F.; Rimbach, K.; Ross, C.; Rotsted, B.; Roussez, H.; Ryder, N.; Saltarelli, M.; Sanders, T.; Santurkar, S.; Sastry, G.; Schmidt, H.; Schnurr, D.; Schulman, J.; Selsam, D.; Sheppard, K.; Sherbakov, T.; Shieh, J.; Shoker, S.; Shyam, P.; Sidor, S.; Sigler, E.; Simens, M.; Sitkin, J.; Slama, K.; Sohl, I.; Sokolowsky, B.; Song, Y.; Staudacher, N.; Such, F. P.; Summers, N.; Sutskever, I.; Tang, J.; Tezak, N.; Thompson, M.; Tillet, P.; Tootoonchian, A.; Tseng, E.; Tuggle, P.; Turley, N.; Tworek, J.; Uribe, J. F. C.; Vallone, A.; Vijayvergiya, A.; Voss, C.; Wainwright, C.; Wang, J. J.; Wang, A.; Wang, B.; Ward, J.; Wei, J.; Weinmann, C. J.; Welihinda, A.; Welinder, P.; Weng, J.;

Weng, L.; Wiethoff, M.; Willner, D.; Winter, C.; Wolrich, S.; Wong, H.; Workman, L.; Wu, S.; Wu, J.; Wu, M.; Xiao, K.; Xu, T.; Yoo, S.; Yu, K.; Yuan, Q.; Zaremba, W.; Zellers, R.; Zhang, C.; Zhang, M.; Zhao, S.; Zheng, T.; Zhuang, J.; Zhuk, W.; Zoph, B. GPT-4 Technical Report. arXiv December 18, 2023. https://doi.org/10.48550/arXiv.2303.08774.

(6) Hatakeyama-Sato, K.; Yamane, N.; Igarashi, Y.; Nabae, Y.; Hayakawa, T. Prompt Engineering of GPT-4 for Chemical Research: What Can/Cannot Be Done? ChemRxiv June 5, 2023. https://doi.org/10.26434/chemrxiv-2023-s1x5p.

(7) Patel, A.; Bhattamishra, S.; Goyal, N. Are NLP Models Really Able to Solve Simple Math Word Problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., Zhou, Y., Eds.; Association for Computational Linguistics: Online, 2021; pp 2080–2094. https://doi.org/10.18653/v1/2021.naacl-main.168.

(8) Parisi, A.; Zhao, Y.; Fiedel, N. TALM: Tool Augmented Language Models. arXiv May 24, 2022. https://doi.org/10.48550/arXiv.2205.12255.

(9) Schick, T.; Dwivedi-Yu, J.; Dessi, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; Scialom, T. Toolformer: Language Models Can Teach Themselves to Use Tools. In *Advances in Neural Information Processing Systems*; Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S., Eds.; Curran Associates, Inc., 2023; Vol. 36, pp 68539–68551.

(10) *Wolfram GPT: For Computation + Curated & Real-Time Data*. https://www.wolfram.com/wolfram-plugin-chatgpt/ (accessed 2024-01-17).

(11) Weng, L. *LLM-powered Autonomous Agents*. lilianweng.github.io. https://lilianweng.github.io/posts/2023-06-23-agent/ (accessed 2024-01-16).

(12) Yano, J.; Gaffney, K. J.; Gregoire, J.; Hung, L.; Ourmazd, A.; Schrier, J.; Sethian, J. A.; Toma, F. M. The Case for Data Science in Experimental Chemistry: Examples and Recommendations. *Nat Rev Chem* **2022**, *6*, 357–370. https://doi.org/10.1038/s41570-022-00382-w.

(13) Back, S.; Aspuru-Guzik, A.; Ceriotti, M.; Gryn'ova, G.; Grzybowski, B.; Gu, G. H.; Hein, J.; Hippalgaonkar, K.; Hormázabal, R.; Jung, Y.; Kim, S.; Kim, W. Y.; Moosavi, S. M.; Noh, J.; Park, C.; Schrier, J.; Schwaller, P.; Tsuda, K.; Vegge, T.; Lilienfeld, O. A. von; Walsh, A. Accelerated Chemical Science with AI. *Digital Discovery* **2024**, *3* (1), 23–33. https://doi.org/10.1039/D3DD00213F.

(14) Jablonka, K. M.; Ai, Q.; Al-Feghali, A.; Badhwar, S.; Bocarsly, J. D.; Bran, A. M.; Bringuier, S.; Brinson, L. C.; Choudhary, K.; Circi, D.; Cox, S.; Jong, W. A. de; Evans, M. L.; Gastellu, N.; Genzling, J.; Gil, M. V.; Gupta, A. K.; Hong, Z.; Imran, A.; Kruschwitz, S.; Labarre, A.; Lála, J.; Liu, T.; Ma, S.; Majumdar, S.; Merz, G. W.; Moitessier, N.; Moubarak, E.; Mouriño, B.; Pelkie, B.; Pieler, M.; Ramos, M. C.; Ranković, B.; Rodriques, S. G.; Sanders, J. N.; Schwaller, P.; Schwarting, M.; Shi, J.; Smit, B.; Smith, B. E.; Herck, J. V.; Völker, C.; Ward, L.; Warren, S.; Weiser, B.; Zhang, S.; Zhang, X.; Zia, G. A.; Scourtas, A.; Schmidt, K. J.; Foster, I.; White, A. D.; Blaiszik, B. 14 Examples of How LLMs Can Transform Materials Science and Chemistry: A Reflection on a Large Language Model Hackathon. *Digital Discovery* **2023**, *2* (5), 1233–1250. https://doi.org/10.1039/D3DD00113J.

(15) Bran, A. M.; Cox, S.; White, A. D.; Schwaller, P. ChemCrow: Augmenting Large-Language Models with Chemistry Tools. arXiv April 12, 2023. https://doi.org/10.48550/arXiv.2304.05376.

(16) Boiko, D. A.; MacKnight, R.; Kline, B.; Gomes, G. Autonomous Chemical Research with Large Language Models. *Nature* **2023**, *624* (7992), 570–578. https://doi.org/10.1038/s41586-023-06792-0.

For Table of Contents Only