

Large language models for isotope separation

Logan Endes, Department of Computer Science, Rochester Institute of Technology,
Rochester, NY, 14623

Carlos Soto, Computer Science Initiative, Brookhaven National Laboratory,
Upton, NY, 11973

Jasmine Hatcher-Lamarre, Collider-Accelerator Department, Brookhaven National
Laboratory, Upton, NY, 11973

Abstract

The Brookhaven National Laboratory Medical Isotope Research and Production group (MIRP) researches and produces experimental isotopes that are used in nuclear medicine for imaging or treatment of diseases like cancer using the Brookhaven Linac Isotope Producer (BLIP). Before producing any isotopes in the BLIP, scientists in MIRP must go through the time-consuming process of analyzing dozens of research papers to identify ideal extraction and separation conditions of these isotopes. This study examines the use of open-source, frozen, large language models (LLMs), such as Galactica, OPT, MPT, GPT-Neo, and LLaMA-2 to extract information on the extraction and separation conditions of isotopes from the scientific literature using inference-only prompt tuning and prompt engineering. To test these models, we will be engineering prompts with different techniques to find which technique and model are the most effective at extracting the required information from the article context; extraction quality will be evaluated with metrics defined together with a domain expert. Through this research, we found that the best approach is Chain of Thought prompting with LLaMA-2-13b which had an accuracy of 0.5, an efficiency of 7.58 gen/min, an average perplexity score of 27.06 and a wastefulness of 0.67. These findings will support further research and development of automated chemical extraction using artificial intelligence and autonomous systems superior to traditional techniques used by researchers. As a result of this summer, I have developed my skills in research and now have a grander understanding of large language models and tuning frozen models to perform tasks that are relevant to my research and goals, and I have also improved my Python programming and problem-solving skills.

1 Introduction

1.1 Background

At Brookhaven National Laboratory (BNL), the Medical Isotope Research and Production group (MIRP) prepares commercially unavailable radioisotopes, like Actinium-225, that are used for the imaging and treatment of diseases like cancer. To do this, they utilize the Brookhaven Linac Isotope Producer (BLIP) that irradiates targets with protons at an intensity of 165 A with an energy of up to 202 MeV originating from Brookhaven’s linear accelerator. Since Brookhaven produces isotopes that are in great demand, researchers at BNL pursue the production pathway of other radioisotopes as well as alternative methods for the production of known isotopes. The problem with this is that the production of isotopes is a time-consuming process in which much of the time is spent identifying chemical separation conditions from previously published scientific literature. This study proposes the use of open-source large language models (LLMs) to extract information on the separation conditions of isotopes from the scientific literature using inference-only prompt engineering in order to accelerate the production of chemical isotopes.

LLMs: Large language models are machine learning algorithms designed to process text in a word-by-word generative mode. With the revelation of the transformer architecture from the 2017 paper, “Attention is All You Need” (Vaswani et. al), large language models became more attainable, as transformer models are much more parallelizable, require much less time to train, and are generally superior in quality to the old norm of using recurrent or



Figure 1: Recently produced Actinium-225 depicted in a long-exposure image.¹

¹<https://www.bnl.gov/newsroom/news.php?a=221204>

convolutional neural networks. Recently LLMs became a hot topic in both the computing space and the media alike with the release of ChatGPT, a highly advanced, though closed-source, LLM optimized for chat-like dialogue that became renowned for its accuracy and confidence.

1.2 Relevant Literature

Attention is All You Need (Vaswani et. al. 2017) proposed the transformer architecture for machine learning models, which is based entirely on attention mechanisms consisting of matrix multiplication, scaling, masking, and softmaxing of matrices comprised of queries (Q), keys (K), and values (V).

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

Transformers are made up of what is known as “multi-head attention” layers, which are basically the concatenation of multiple attention mechanisms.

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_i) \quad (2)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (3)$$

Where W_i^Q, W_i^K, W_i^V are projections represented as parameter matrices.

One way that transformers use this multi-head attention is in encoder-decoder attention layers in which “queries come from the previous decoder layer, and the memory keys and values come from the output of the encoder.”

LLMs take the encoder-decoder attention layers and utilize the decoder aspect of that attention to create a decoder-only architecture which allows for the function of LLMs to perform text generation as a type of auto-complete.

Automated Extraction of Molecular Interactions and Pathway Knowledge using Large Language Model, Galactica: Opportunities and Challenges (Park et. al. 2023) investigates the use of Galactica in life science research related to protein interactions. Ultimately, our fellow researchers at BNL’s Computer Science Initiative concluded that Galactica performs best when contextual text is provided, indicating a relevant connection to the use of Galactica for use in isotope production based on previous research.

Med-HALT: Medical Domain Hallucination Test for Large Language Models (Umapathi et. al. 2023) looks into the challenges posed by hallucinations in LLMs. In the end, they found that while Falcon 40B is the most performant model of their suite that includes MPT-7B and all LLaMA-2 variations in information retrieval, all of the models face a multitude of hallucination problems.

Local Large Language Models for Complex Structured Tasks (Bumgardner et. al. 2023) describes the “end-to-end process of training, evaluating, and deploying” local LLMs to perform domain-specific NLP tasks; coming to the conclusion that LLaMA-based models perform much better than BERT-based models in all NLP tasks due to their size difference. This paper may provide insight on better ways to implement the models we are using in the domain of chemical separations and isotope production.

Llama 2: Early Adopters’ Utilization of Meta’s New Open-Source Pretrained Model (Roumeliotis et. al. 2023) provides a description of multiple different projects utilizing the new LLaMA-2 models. Of the described projects, “Document-based-question-answering-system-using-LLamaV2-7b”² and “H2oGPT”³ seem to be the most relevant. The former is of great interest, as it is a question-answering system based on documents, which could almost certainly be adapted to papers on isotope production. The latter functions to query and summarize documents using new models like LLaMA-2 and Falcon which seems worthwhile to look further into for the future of this project.

²https://github.com/10deepaktripathi/Document_based_question_answering_system_using_LLamaV2-7b

³<https://github.com/h2oai/h2ogpt>

1.3 Scope

For this project, we were mostly limited by hardware. All models and tests run in this study were run on a NVIDIA RTX 4090 with 24 Gigabytes of VRAM. In addition to this limitation, this project is essentially an experimental proof-of-concept for later work to be built on. All work done during this study can be found at https://github.com/Log45/LLMs-For_Isotopes

2 Methods

2.1 Document Parsing

A key part of drawing conclusions based on research papers is being able to access information from those documents. The majority of scientific literature is only accessible through PDF and HTML formats. While it is simple for humans to read papers from websites and PDFs, LLMs need access to the raw text of those documents in order to correctly tokenize the meanings of those words. To get access to the raw text of literature, we had to be able to parse through both PDFs and HTML files.

PDF Parsing: In order to access text from PDFs, we used a Python library known as PDFQuery, which allows users to simply access different attributes from PDFs, which build documents by placing text boxes and other features at specific coordinates. The most important attribute, and the one that we access, are the text elements of the PDFs which we then extract into a text file. We take this text file one step further and split it into a list of context paragraphs which are split up based on the size of gaps between text in the file.

HTML Parsing: In order to access text from HTML files, we used a Python class known as BeautifulSoup from the bs4 library, which allows users to search for key features like classes from HTML code. Since the needed text is nested behind specific classes in HTML, we were able to identify the relevant class of text and return each separate paragraph associated with that class in a list of contexts. A downside to using HTML is that each website that has

relevant articles uses different class names, so the parser currently only works for papers from mdpi.com, though it would not be much more work to make it compatible with other websites like springer.com that have relevant papers.

2.2 Context Filtering

In an attempt to avoid extracting information from irrelevant paragraphs from papers, we created a variety of different filters that are applied to the data before the LLMs return conclusions based on those contexts.

Keyword Filter: The keyword filter is defined well by its name, we sort through each context paragraph and only generate responses with the LLMs if there are any keywords present in the paragraph. The keywords that we searched for were: separation, isolation, chromatography, ion exchange, eluted, elution, elute, fraction, resin, exchange, acid, and target.

Model Filter: The model filter uses the LLM to confirm whether or not each context contains a chemical extraction in it. If the model thinks that a chemical extraction is being described and it returns “yes” as an answer, then the generation is carried on; if it does not answer “yes”, then the paragraph is thrown out and it moves onto the next.

Keyword-Model Filter: The keyword-model filter combines both the keyword filter and model filter. If a paragraph contains a relevant keyword, then the LLM is asked whether there is a chemical separation in the paragraph, and it continues on to generate conclusions if it thinks there is a separation described.

Keyword-Model-Check Filter: The keyword-model-check filter combines the keyword filter and model filter as previously described, but takes the filtering a step further by asking whether or not the generation is a truthful statement in regards to the original context. If the model responds with “yes,” then it allows the generation to be added to the list of all generations. This filter differs from previous filters in that it does not attempt to avoid extra generations and instead attempts to produce more accurate conclusions.

Keyword-Model-Expert Filter: The keyword-model-expert is less of a filter for the context and more of an attempt at manipulating the LLM to respond more accurately by telling it to act like an expert in chemistry whereas the keyword-model filter included with it filters the contexts used for conclusion generation.

Keyword-Model-Expert-Check Filter: The keyword-model-expert-check filter combines all of the previously described methods to filter as much irrelevant data as possible while also attempting to generate the most accurate responses at the cost of generation efficiency.

2.3 Prompting Techniques

Zero-Shot: Zero-shot prompting (Wei et. al. 2022)⁴ is the simplest prompting technique for LLMs done by either asking the model a question without any context to look for a response, or by providing a context paragraph and asking the model to draw a conclusion based on that. For our purpose, the latter is a relevant technique to try to generate chemical separation suggestions as simply as possible, though the zero-shot capabilities of smaller parameter models like the ones we are using are not generally as great as those of models with tens of billions of parameters like ChatGPT.

Few-Shot: Few-shot prompting is not much more complex than zero-shot prompting, but allows for a much greater potential for the LLM to generate a desired output by giving at least one example of what the user expects in the context given to the model. In this study, we observed the use of few-shot prompting with one example as well as two examples.

Chain-of-Thought: Chain-of-Thought (CoT) prompting (Wei et. al. 2022)⁵ is much more performant than standard few-shot and zero-shot prompting for certain tasks as it allows for LLMs to use reasoning in their response generations. CoT prompting can be either few-shot or zero-shot. For few-shot CoT, the user must give an example answer in the context that the model reads that breaks down logical steps to come to that conclusion. In

⁴“Finetuned Language Models are Zero-Shot Learners”

⁵“Chain-of-Thought Prompting Elicits Reasoning in Large Language Models”

doing so, the model is more inclined to use reasoning when generating conclusions due to its nature of imitation. For zero-shot CoT, the user must instill reasoning in the question itself or start off the response for the LLM to finish. The most popular zero-shot CoT phrase is “Let’s think step by step,” (Kojima et. al. 2022). In our experiments, asking the model to act as an expert in chemistry functionally emulates this instillation of reasoning.

Automatic Prompt Engineer: Automatic Prompt Engineer (APE) (Zhou et. al. 2022) is a framework that uses an LLM to generate and select an instruction to give based on a specific task. Essentially, a model is given a list of input-output pairs and is asked to give an instruction to induce a language model to generate those pairs. After that, the same LLM is given those generated tokens and returns a logit-loss score to grade each instruction and choose the best one. The issue with APE is that the original framework is built to interface with OpenAI’s API, which is closed-source and therefore out of the scope due to the open-source requirement of this project. To combat this, we spent a portion of the time dedicated to this project to implement a localized version of APE, currently only supporting Facebook’s OPT models, though it will be easy to generalize that to HuggingFace’s transformers in the future.

2.4 Models

Galactica: Galactica (Taylor et. al. 2022) is a model trained for the primary purpose of performing scientific and technical tasks trained on a corpus consisting of scientific papers, reference material, knowledge bases, and other sources that outperforms many models in LaTeX equations, mathematical reasoning, and even general tasks.

LLaMA-2: LLaMA-2 (Touvron et. al. 2023)⁶ is the second iteration of Meta AI’s LLaMA (Large Language model Meta AI) (Touvron et. al. 2023)⁷ pretrained language model. While this second iteration focuses more on overhauling chat dialogue in their model, it is generally more performant than their original model which was already known as one

⁶Llama 2: Open Foundation and Fine-Tuned Chat Models

⁷LLaMA: Open and Efficient Foundation Language Models

of the most performant LLMs in 2023. While LLaMA-2 did not reveal its training corpus, it is very performant at a variety of tasks and stands as one of the best models we could use in this study.

Falcon: Falcon⁸, a new LLM by Technology Information Institute has shown impressive performance with both its 40B and 7B parameter variants, with its 40B variant outperforming LLaMA-65B in some tests. This model was trained on a corpus of 79 percent RefinedWeb (Penedo et. al. 2023) dataset and 2 percent of scientific data from different sources like arXiv, PubMed, USPTO, and more. Based on claims by the creators, Falcon is a great candidate for the application of information extraction from research papers.

MPT-7B: Although MPT-7B (MosaicML NLP Team 2023) is definitely a great candidate for this project and should definitely be explored in the future, we were not able to get it to work due to dependency issues and therefore could not test it.

Other Models: While we were looking into many more models including GPT-Neo (Black et. al. 2021), OPT (Zhang et. al. 2022), Falcon-RW-1B (Penedo et. al. 2023), Pythia (Biderman et. al. 2023), BLOOM (BigScience Workshop 2022), and MPT-1B⁹, our initial tests allowed us to conclude that those models were not worth testing further; instead, we put all of our focus into more relevant models like Galactica, LLaMA-2, and Falcon.

Name	Publisher	Parameters	Quantized?	Scientific?
Galactica-1.3b	Meta AI	1.3 billion	No	83%
Galactica-6.7b	Meta AI	6.7 billion	Yes	83%
LLaMA-2-7b	Meta AI	7 billion	Yes	--%
LLaMA-2-13b	Meta AI	13 billion	Yes	--%
Falcon-7b	TII	7 billion	Yes	2%

Table 1: Depicts relevant models to our problem, their publisher, the parameters in the model, whether the model is quantized, and the percentage of training data which was scientific literature. LLaMA-2 did not report its training data.

⁸<https://falconllm.tii.ae/>

⁹<https://huggingface.co/mosaicml/mpt-1b-redpajama-200b>

3 Results

Automatic Prompt Engineer: Although we were able to develop an implementation of APE that generated potential prompts, we decided it was best to move on to other techniques, as the initial results were very poor and the project was heading in the direction of context-filtering as opposed to automatic prompts. Despite this, there is definitely still potential in using APE if it is further developed and fine-tuned for smaller local models.

After giving each LLM context from different research papers and asking questions based on different aspects of chemical separations like the target material, the acid used to dissolve the target, the resin, the elution acid, and the products of the separation, they successfully generated suggestions for chemical separations, though not completely accurate due to hallucination as shown in Figure 2.

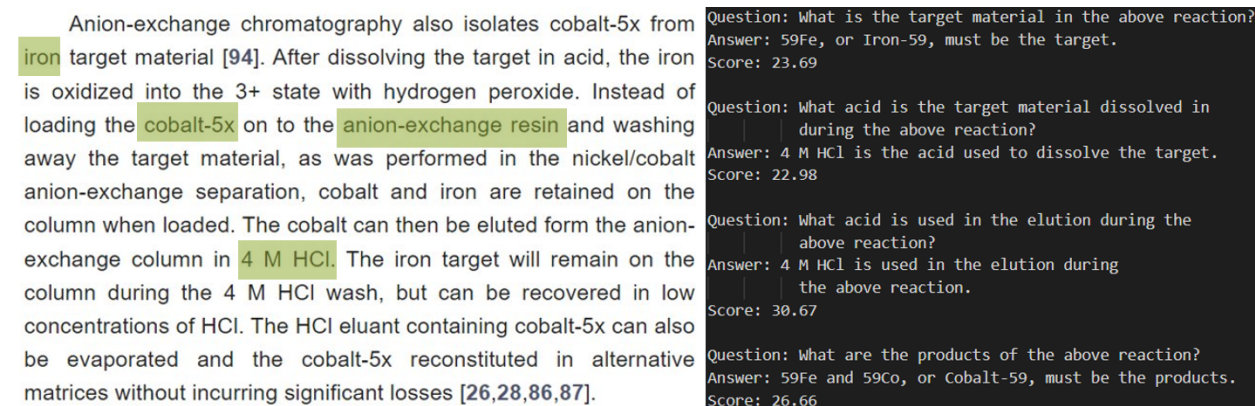


Figure 2: Human annotations (left) vs. AI generations (right) on a paragraph from “Production, Purification, and Applications of a Potential Theranostic Pair: Cobalt-55 and Cobalt-58m” (Barrett et. al. 2021)

Through a benchmark that we developed using two papers on isotope separation (Barrett et. al. 2021), (Xie et. al. 2017), we scored variations of each relevant model, filtering technique, and prompting technique on accuracy, efficiency, perplexity, and wastefulness. These results can be seen in the figures below.

Ultimately, we found that LLaMA-2-13b was the most accurate with an accuracy of 0.5, an efficiency of 7.58 gen/min, a perplexity score of 27.06, and a wastefulness of 0.67.

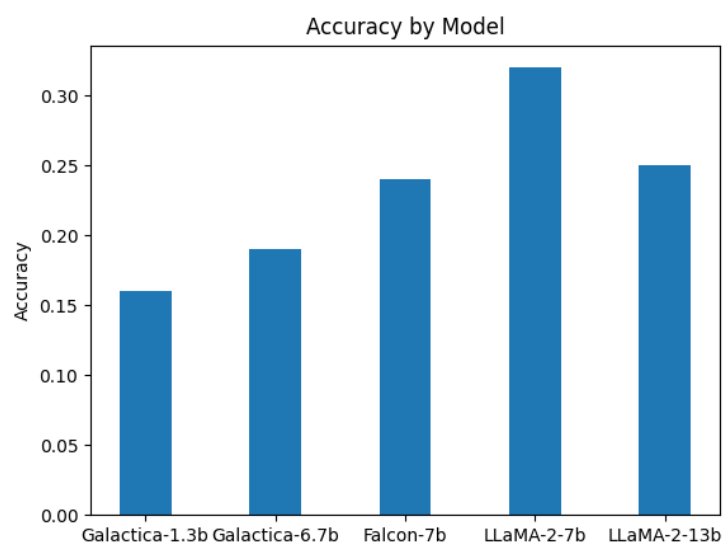


Figure 3: Average accuracy of each model

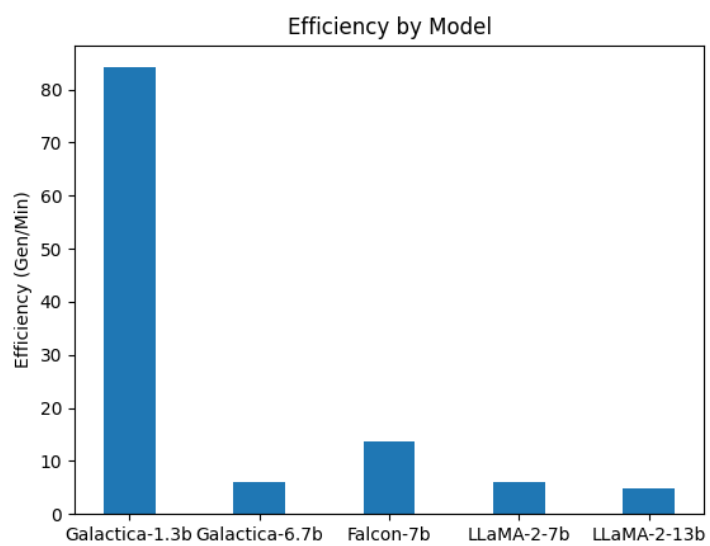


Figure 4: Average efficiency of each model

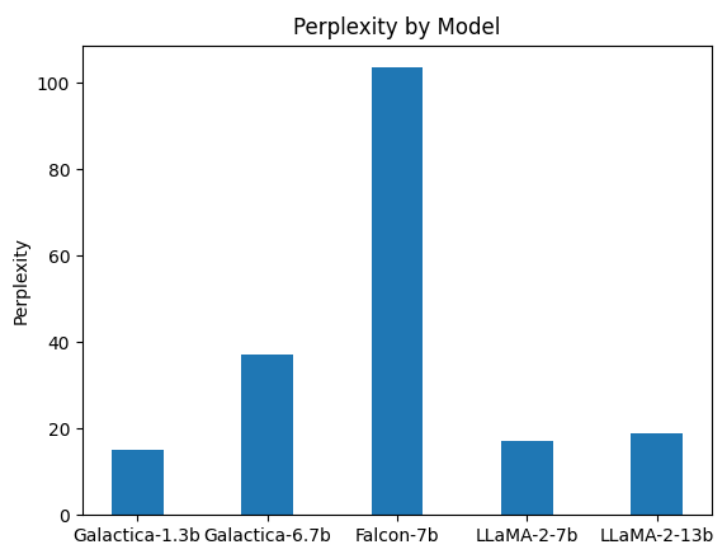


Figure 5: Average perplexity score of each model

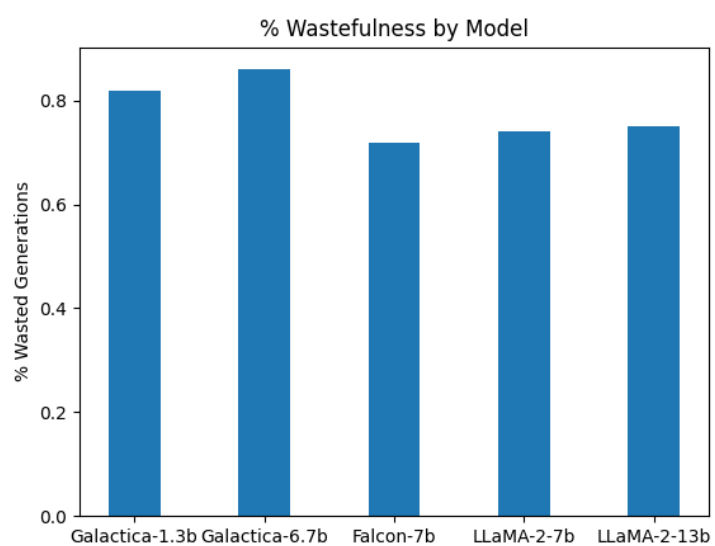


Figure 6: Average wastefulness of each model

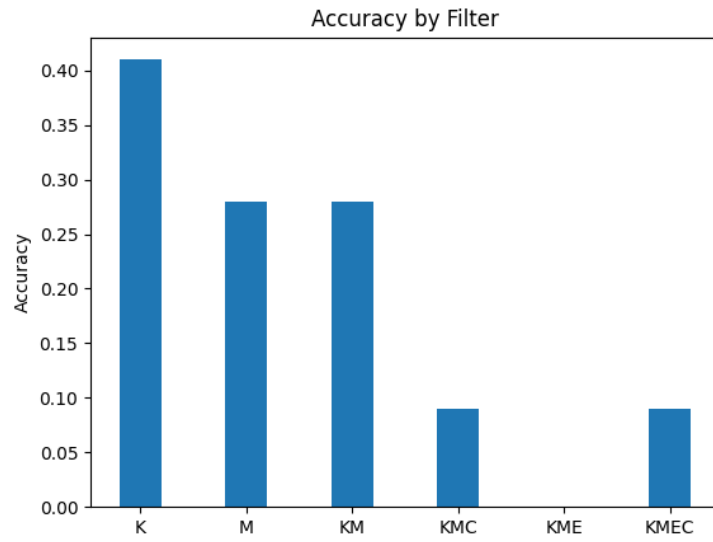


Figure 7: Average accuracy of each filter. K=Keyword, M=Model, C=Check, E=Expert

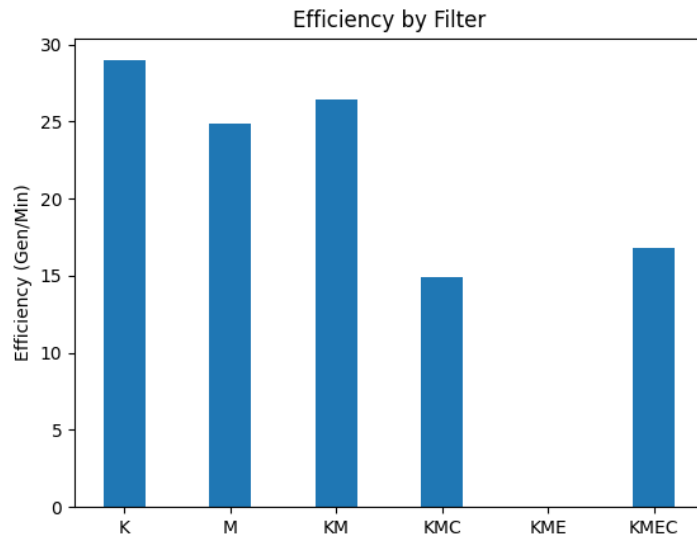


Figure 8: Average efficiency of each filter

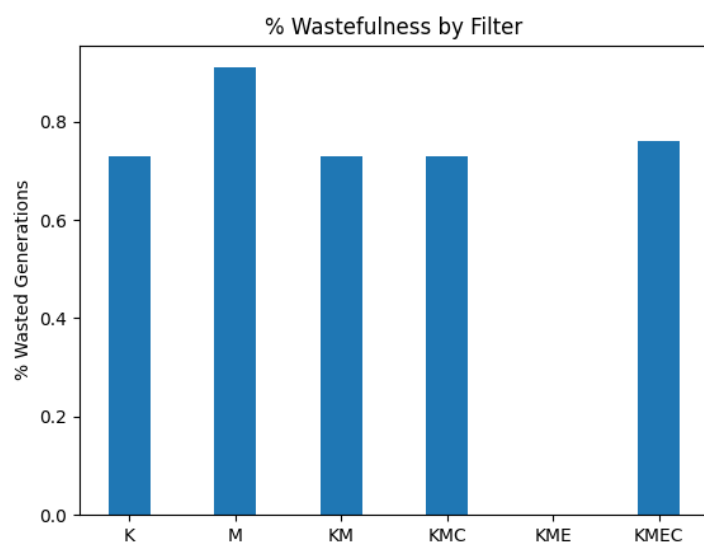


Figure 9: Average wastefulness of each filter

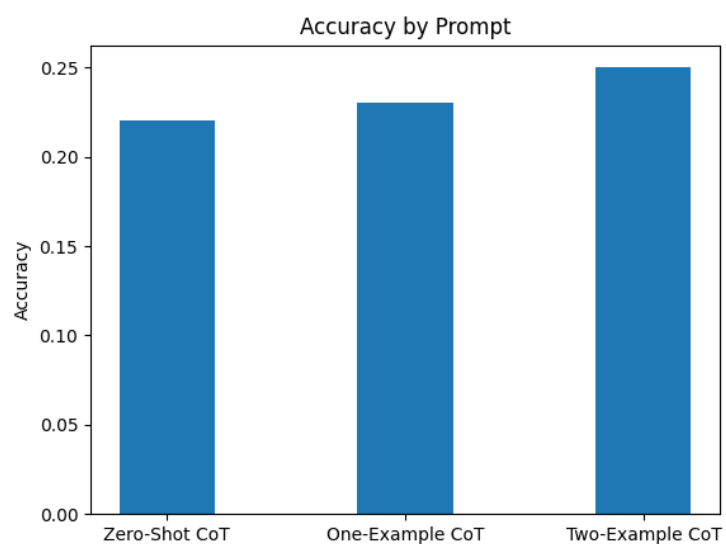


Figure 10: Average accuracy of each prompt

4 Conclusion

In this paper, we examined the use of LLMs and inference-only prompt engineering in conjunction with context-filtering techniques to aid in the research of chemical separations and isotope production. We developed document parsers for PDF and HTML file formats based on existing Python libraries. We investigated the effectiveness of different filters in avoiding wasteful generations by language models. We compared the accuracy and efficiency of different prompting techniques in conjunction with different pretrained LLMs. Ultimately, we found that although language models continue to hallucinate, there is certainly potential for further development of this project to become a very helpful asset to researchers in the future.

Discussion: The findings of this study is going to support the future development of an automated chemical separation system in which an LLM will be used to suggest components of different chemical separations that the automated system can then test, rapidly accelerating the slow process of isotope research and production. In support of this goal, I believe that both an LLM and a BERT model with name-entity recognition can be used together to ensure the simplest suggestions for the previously mentioned separations. For example, the LLM would generate a suggestion for a target, acid, resin, elution, and product of a separation in natural language, and an NER model will be applied to those generations to return only the relevant parts of those generations such as the specific element, acid, concentration, etc.

Acknowledgements

This project was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internships Program (SULI). I would like to thank my mentor, Carlos Soto, and my PI, Jasmine Hatcher-Lamarre, for the great experience this summer, and the great insights that they have provided me. This paper is not export controlled.

References

- [1] Kendall E. Barrett, Hailey A. Houson, Wilson Lin, Suzanne E. Lapi, and Jonathan W. Engle. Production, purification, and applications of a potential theranostic pair: Cobalt-55 and cobalt-58m. *Diagnostics*, 11(7), 2021.
- [2] Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O’Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling, 2023.
- [3] Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021. If you use this software, please cite it using these metadata.
- [4] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020.
- [5] V. K. Cody Bumgardner, Aaron Mullen, Sam Armstrong, Caylin Hickey, and Jeff Talbert. Local large language models for complex structured medical tasks, 2023.
- [6] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2023.
- [7] Gilchan Park, Byung-Jun Yoon, Xihai Luo, Vanessa Lopez-Marrero, Patrick Johnstone, Shinjae Yoo, and Francis Alexander. Automated extraction of molecular interactions

- and pathway knowledge using large language model, galactica: Opportunities and challenges. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 255–264, Toronto, Canada, July 2023. Association for Computational Linguistics.
- [8] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only, 2023.
- [9] Konstantinos I. Roumeliotis, Nikolaos D. Tselikas, and Dimitrios K. Nasiopoulos. Llama 2: Early adopters’ utilization of meta’s new open-source pretrained model. *Preprints*, August 2023.
- [10] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science, 2022.
- [11] MosaicML NLP Team. Introducing mpt-7b: A new standard for open-source, commercially usable llms, 2023. Accessed: 2023-05-05.
- [12] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi,

- Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.
- [13] Logesh Kumar Umapathi, Ankit Pal, and Malaikannan Sankarasubbu. Med-halt: Medical domain hallucination test for large language models, 2023.
- [14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [15] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *CoRR*, abs/2109.01652, 2021.
- [16] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903, 2022.
- [17] BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muenighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou,

Chris Emezue, Christopher Klammm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raulnak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max

Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette,
 Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette,
 Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden
 Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet
 Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névél,
 Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Takta-
 sheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf,
 Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Ka-
 sai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim,
 Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen
 Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas
 Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav
 Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kas-
 ner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana
 Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Ay-
 cha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat
 Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky,
 Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne
 Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani
 Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh
 Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot
 Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed
 Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy,
 Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sar-
 mad Shubber, Silas Wang, Sourav Roy, Sylvain Viguié, Thanh Le, Tobi Oyebade,
 Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano,
 Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin

Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourier, Daniel León Perrián, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrmann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sängner, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. Bloom: A 176b-parameter open-access multilingual language model, 2023.

- [18] Qinghua Xie, Hua Zhu, Feng Wang, Xiangxi Meng, Qiushi Ren, Chuanqin Xia, and Zhi Yang. Establishing reliable cu-64 production process: From target plating to molecular specific tumor micro-pet imaging. *Molecules*, 22(4), 2017.
- [19] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. Opt: Open pre-trained transformer language models, 2022.

- [20] Yongchao Zhou, Andrei Ioan Muresanu, Ziwon Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers, 2023.