Exposé on Using Large Language Models for Automated Data Extraction from Scientific Literature

Felix Karg*

February 20, 2023

1 Introduction

A large amount of scientific knowledge is scattered across millions of research papers. Often, this research is not in standardized machine-readable formats, which makes it difficult or impossible to build on prior work using powerful tools to extract further knowledge.

2 Motivation

Take for example the field of synthesizing Metal-Organic Frameworks (MOFs) [14]. While numerous detailed descriptions of synthesis procedures exist, they are not in a machine-readable format, which prevents effective application of state-of-the-art techniques such as automated experimentation [8] or synthesis prediction [6]. Thus, we intend to create a pipeline for deriving machine-readable information on MOF synthesis parameters from given questions on provided scientific articles.

3 Background

Rule-Based Entity Recognition There have long been rule-based approaches for the recognition of individual entities. ChemTagger [4] clearly demonstrated that simple rule-based systems can sometimes extract much of the requested information. While they often achieve high precision for simple tasks, they fail in answering more complex queries, such as the relation between two entities.

Language Models With 'Attention is All you Need' [10], Google introduced the transformer architecture for language models and demonstrated significant improvements. Soon, BERT [2] followed, a model which is conceptually simple and empirically powerful. It was soon demonstrated that BERT can be easily fine-tuned for named entity recognition in materials science [13]. OpenAI pushed scaling forward with their GPT2 [7] model, which was substantially larger than BERT. Step-by-step, these models enabled more sophisticated extraction requests.

*This work has partially been augmented using ChatGPT

Large Language Models With the introduction of GPT3 [1] OpenAI trailblazed the era of Large Language Models. This model enabled more sophisticated information extraction requests with little fine-tuning [3]. Soon, open-source variants such as OPT [11] followed. It was also demonstrated with Chinchilla [5] and CoTR [12], that these large language models are substantially overparametrized and undertrained.

4 Scientific Questions

Use Large Language Models to demonstrate automated extraction of unstructured text from scientific literature for the creation of a database with otherwise unavailable information on MOF synthesis. By doing so, we create a pipeline that can easily be adapted to numerous other data extraction tasks.

Specifically, using OPT [11] empirically test if accuracy can be improved via 1) fine-tuning and 2) prompt engineering. Additionally, test if 3) model size can be reduced by using distillation [9], and how it will affect accuracy as well as compute and memory requirements. Distillation would enable considerable model parameter reduction with little loss in accuracy, which could make it substantially less compute intensive to fine-tune and run.

5 Schedule

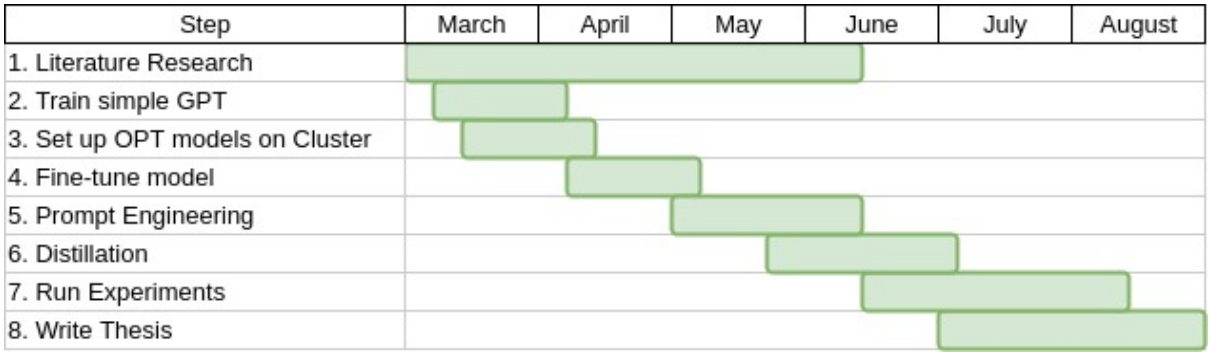

Figure 1: Exemplary timeline. Details are subject to change.

5.1 Literature Research

In this step, we will conduct a thorough review of the relevant literature related to our topic. This includes reading academic articles, research papers, and other publications to gain a better understanding of the current state of knowledge in the field. By continuously reading relevant literature, we can stay up-to-date with the latest advancements and identify gaps in the research that we can address in our own work.

5.2 Train Simple GPT

We will train a small GPT model to gain a deeper understanding of its architecture, training procedure, and properties. By doing this, we can get hands-on experience with

the model and better understand its strengths and limitations. This will help us to make informed decisions about how to fine-tune the model for our specific task.

5.3 Set up OPT models on Cluster

To set up the open-source OPT models on a cluster, we will need to carefully configure the models to ensure that our experiments are run on a reliable and scalable computing infrastructure. While the models can be downloaded, the configuration process may not be straightforward, and it will be critical to get it right to ensure the reliability and scalability of our experiments.

5.4 Fine-tune model

In this step, we will construct and train the model on select examples, with intermediate annotations or validation in-between, similar to what is described in [3]. The first goal is to increase the task success rate, which means answering questions in the requested machine-readable format. The second goal is to increase accuracy. To achieve high accuracy, we may manually annotate a few examples, and augment using partially annotated ones.

5.5 Prompt Engineering

In this step, we will apply deep introspection and automatic prompt engineering [15] to increase the accuracy of generated databases. Prompt engineering involves designing the prompts that the model will use to generate responses. By optimizing the prompts, we can improve the quality of the model's output and make it more accurate.

5.6 Distillation

Distillation is a technique for reducing the size of a large model while maintaining its accuracy. In this step, we will apply distillation [9] to our model to reduce its parameter size while keeping accuracy high. This will make the model more efficient and easier to deploy in production.

5.7 Run Experiments

In this step, we will run detailed experiments to evaluate the performance of our model. We will generate graphs, tables, and databases of extracted information to analyze and interpret the results. By running experiments, we can validate the effectiveness of our approach and identify areas for improvement.

5.8 Write Thesis

In the final step, we will write an extensive scientific article as the concluding work of our master's degree. This will involve summarizing our research, detailing our methodology, presenting our results, and discussing the implications of our findings. Writing the thesis is an essential part of the research process and allows us to share our insights and contributions with the broader academic community.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, and Amanda Askell. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[3] Alexander Dunn, John Dagdelen, Nicholas Walker, Sanghoon Lee, Andrew S. Rosen, Gerbrand Ceder, Kristin Persson, and Anubhav Jain. Structured information extraction from complex scientific text with fine-tuned large language models, December 2022.

[4] Lezan Hawizy, David M. Jessop, Nico Adams, and Peter Murray-Rust. ChemicalTagger: A tool for semantic text-mining in chemistry. *Journal of Cheminformatics*, 3(1):17, May 2011.

[5] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models, March 2022.

[6] Yi Luo, Saientan Bag, Orysia Zaremba, Adrian Cierpka, Jacopo Andreo, Stefan Wuttke, Pascal Friederich, and Manuel Tsotsalas. MOF Synthesis Prediction Enabled by Automatic Data Mining and Machine Learning**. *Angewandte Chemie International Edition*, 61(19):e202200242, 2022.

[7] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

[8] Yao Shi, Paloma L. Prieto, Tara Zepel, Shad Grunert, and Jason E. Hein. Automated experimentation powers data science in chemistry. *Accounts of Chemical Research*, 54(3):546–555, 2021.

[9] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient Knowledge Distillation for BERT Model Compression, August 2019.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, \ Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[11] 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, June 2022.

[12] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal Chain-of-Thought Reasoning in Language Models. *arXiv preprint arXiv:2302.00923*, 2023.

[13] Xintong Zhao, Jane Greenberg, Yuan An, and Xiaohua Tony Hu. Fine-Tuning BERT Model for Materials Named Entity Recognition. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 3717–3720. IEEE, 2021.

[14] Hong-Cai Zhou, Jeffrey R. Long, and Omar M. Yaghi. Introduction to Metal–Organic Frameworks. *Chemical Reviews*, 112(2):673–674, February 2012.

[15] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large Language Models Are Human-Level Prompt Engineers, November 2022.