PACuna: Automated Fine-Tuning of Language Models for Particle Accelerators

Antonin Sulc*
DESY,
Hamburg, Germany
antonin.sulc@desy.de

Raimund Kammering
DESY,
Hamburg, Germany

Annika Eichler
DESY,
Hamburg, Germany

Tim Wilksen
DESY,
Hamburg, Germany

Abstract

Navigating the landscape of particle accelerators has become increasingly challenging with recent surges in contributions. These intricate devices challenge comprehension, even within individual facilities. To address this, we introduce PACuna, a fine-tuned language model refined through publicly available accelerator resources like conferences, pre-prints, and books. We automated data collection and question generation to minimize expert involvement and make the code available. PACuna demonstrates proficiency in addressing accelerator questions validated by experts. Our approach shows adapting language models to scientific domains by fine-tuning technical texts and auto-generated corpora capturing the latest developments can further produce pre-trained models to answer some specific questions that commercially available assistants cannot and can serve as intelligent assistants for individual facilities.

1 Introduction

Modern AI assistants like `ChatGPT` OpenAI [2021] and `Claude` Anthropic [2023] have revolutionized artificial intelligence, showcasing impressive creativity. However, they rely on questionable internet sources and can generate plausible but incorrect responses when lacking knowledge, well known as hallucinations Welleck et al. [2019], which is a known and limiting feature of LLMs. Furthermore, the complexity of the numerous commercial large language models makes it difficult to easily fine-tune these models to incorporate the latest advancements. On the other hand, fine-tuning smaller but powerful LLMs like Touvron et al. [2023] or Chiang et al. [2023] for particular domains on more reputable sources like scientific papers addresses these issues. Scientific papers provide a more trustworthy source of information for creating a more reliable training dataset. This brings a potentially wide range of applications for the community. For instance, one can fine-tune and automate an LLM to perform department-specific tasks that consist of a specific language. These intelligent assistants can improve FAIR principles and can help in diverse applications like logbooks which are critical infrastructure components in particle accelerator research and development. To tackle this, one can consider using off-the-shelf parameter-efficient fine-tuning methods Hu et al. [2021], Chen et al. [2023] that adapt smaller models Chiang et al. [2023], Touvron et al. [2023] into domain or facility-specific assistants that are able to work with complex conceptual questions where commercial chatbots will often fail. However, the challenging part is the automated preparation of high-quality training data tailored to the domain.

*https://github.com/sulcantonin/LLM_NeuralIPS23.git

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.

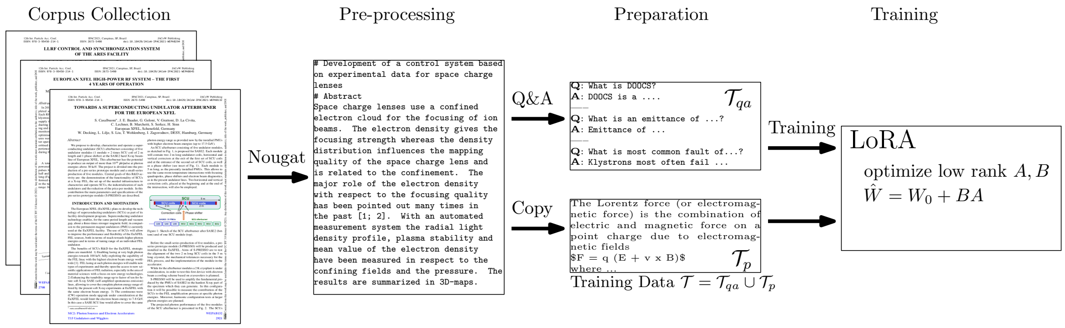

Figure 1: Overview of the training pipeline. The shown pipeline compiles complementary text sources (Step 1). These sources are processed with Nougat OCR Blecher et al. [2023] and standardized formats to extract text, equations, and tables (Step 2). The structured data is used for text prediction training $\mathcal{T}_p$ and to auto-generated question-answer pairs $\mathcal{T}_{qa}$ (Step 3). Finally, the prepared dataset fine-tunes an LLM using memory efficient LoRA Hu et al. [2021] for robust training (Step 4).

In this work, we present an approach that automates data preparation for fine-tuning publicly available models, enabling training customized AI assistants. We introduce the first accelerator assistant, trained without a human in the loop in dataset preparation, which can answer questions that commercially available chatbots are not able to answer. Furthermore, we show an automated pipeline that enables automation for arbitrary domains without the laborious creation of a supervised dataset and automates this process instead.

2 LLM in Particle Accelerator Technology

LLMs have the potential to benefit the particle accelerator R&D community, though applications are not limited to this domain, some examples might be:

1. LLMs can advance accelerator control and automation by developing systems that respond to condition changes, improving stability and safety, improving the FAIR-ness of logbooks, and helping operators to interpret data, see Section 7.
2. LLMs can assist in interpreting facility conditions by harvesting existing (multi-modal) data.
3. A unified LLM could enhance communication and collaboration within and outside facilities by providing natural language interfaces for discussing complex concepts, sharing results, and collaborating on experiments.
4. LLMs can assist in accelerator design and optimization by consolidating knowledge about facilities and making it accessible to personnel, aiding the exploration of configurations and parameters from past or published results via incorporating additional multimodal inputs (images, screenshots, machine parameters), see Section 7 for more details.
5. LLMs can be used to teach accelerator principles, making this specialized knowledge more accessible to anyone interested.

3 Pipeline of Fine-tuning the LLM

The pipeline can be summarized in the following steps: Preparing training data begins with collecting corpus sources described in Section 4.1 consisting of books, conference proceedings, and pre-prints to create a knowledge base in the form of PDF documents. These sources are processed with optical character recognition (OCR) to extract text, equations, and tables as we explain in Section 4.2. The structured data is directly used for text prediction training (unsupervised dataset $\mathcal{T}_p$). For question answering (supervised dataset $\mathcal{T}_{qa}$), sections of text are prompted to auto-generate question-answer pairs explained in Section 4.3. In the final step, in Section 5, the prepared dataset then fine-tunes LLM optimized for long contexts and efficient training, using memory efficient LoRA Hu et al. [2021] to enable robust training on conventional hardware.

The result is an AI assistant tailored to a scientific domain by leveraging accessible training data $\mathcal{T}$ spanning current publications and foundational texts. The automated pipeline enables continuous updating to remain updated with the state-of-the-art in the field. The steps are shown Figure 1.

4 Dataset

4.1 Corpus Sources

To create a dataset, we need text corpus sources. This work uses three complementary data sources to generate a robust dataset covering foundational knowledge and recent progress. Publicly available books provide broad, established knowledge to build a strong basis, referred to as the `books` corpus. Conference proceedings from JACoW contain the most up-to-date facility developments, referred to as `jacow`. arXiv arXiv includes currently peer-reviewed pre-prints in category `physics.acc-ph` (accelerator physics) from 2015 till this day and is denoted as `arxiv`.

The source list, including automated acquisition procedures, will be published with the dataset to enable continuous updates with new publications. This corpus establishes a comprehensive foundation while capturing the latest advancements.

4.2 Dataset Pre-processing

In the pre-processing step, we focus on digitizing documents into machine-readable formats.

Nougat OCR Blecher et al. [2023] uses a visual transformer to process scientific documents into a computer-readable format. It can transform a scientific paper into computer-readable text, equations, and tables. Nougat converts inputs to MultiMarkdown, which is better suited for training than raw (La)TeX, however since LaTeX is the standard for equation typesetting, we keep the equations in LaTeX.

We transformed all computer-readable outputs from Nougat into a common format with text as Markdown, tables as plain text, and equations in LaTeX. Nougat initially outputs MultiMarkdown, which requires some minor modifications, namely we transformed equations to LaTeX via regular expressions and tables to plain text with Pandoc MacFarlane [2023].

Only `books` and `papers` sources were processed by Nougat. `arXiv` texts are already in LaTeX, so we used Pandoc MacFarlane [2023] to convert to MultiMarkdown, replaced equation delimiters with LaTeX (`$`), and rendered other macros (like image captions) as plain text that are better suited for training.

4.3 Dataset Preparation

The unsupervised dataset $\mathcal{T}_p$ for the text prediction does not require any further steps after pre-processing and we pass the input data as we describe in the pre-processing step, see Section 4.2.

To generate question-answer supervised pairs we prompted `vicuna-7b-16k-v1.5` with a following prompt: `Generate ten questions for a paper:"$TEXT"` where `$TEXT` is the actual paper to generate ten questions per either a section (or subsection) form `book` corpus or paper from `jacow` corpus. If the generated answer did not follow a pre-specified format, it was discarded.

The entire dataset consists of 633759 samples, from which 24949 are unsupervised `arXiv` samples (papers), 1689 samples are from `books` corpus (sections or subsections, depending on length), and 338207 samples (papers) are from `papers` corpus. Generated supervised question-answer pairs consist of 13705 samples from `book` (single question and answer pair, if processed properly) corpus and 255209 `papers` corpus (single question and answer pair, if processed properly).

5 Fine-tuning

We leverage the pre-trained transformer language model `vicuna-7b-16k-v1.5` Chiang et al. [2023]. It provides a robust foundational model while being reasonably small (7 billion parameters) and is trainable on conventional GPUs. Furthermore, it can handle up to 16k input tokens, essential for often lengthy scientific texts. Compared to larger models like `llama-70b` Touvron et al. [2023], it has fewer parameters and thus it is easier to be continuously updated and fine-tuned on conventional hardware. Fine-tuning uses LoRA Hu et al. [2021] to make large language model training accessible on standard hardware. This combination provides a capable foundation while remaining efficient and accessible. Considering the input lengths, one may consider saving computational time by using LongLoRA Chen et al. [2023] in the future. We set the training LoRA rank to 64 since datasets contain some intricate texts like equations and tables, α is set to 128. We optimized the query, key, value, and projection weights. Per-device train batch size set to 2 and gradient accumulation 16 turned out to be a good trade-off between training time and model efficiency. The training was performed with community-suggested 4 epochs and with a learning rate 5×10^{-5}. The training was done on NVIDIA A100.

6 Evaluation

We evaluated our model with state-of-the-art open-source `falcon-180b` AI Almazrouei et al. [2023] and `ChatGPT` OpenAI [2021]. The shown results are on the model which was trained on papers that contain mention of DESY. We also publish a model that is trained on the entire $\mathcal{T}$. When the output didn't contain a definitive answer, we omitted the text, indicated by ellipsis.

Question 1 about DESY ARES : *Does ARES use normal or superconducting cavities?*
`PACuna`: *ARES uses normal cavities for its accelerator*, Correct
`falcon-180b`: *I do not have information about a specific ares system. However, ares is a general acronym that can refer to various systems....* Wrong, the answer is missing.

`ChatGPT`: *As of my last knowledge update in September 2021, ARES (Accelerator for Research in Subatomic Physics) had plans to use superconducting radiofrequency (SRF) cavities ...*, Wrong, they are not superconducting.

Question 2 about DESY European XFEL : *How many superconducting radio frequency (SRF) cavities does European XFEL have?*
`PACuna`: *European XFEL has a total of 800 superconducting radio frequency (SRF) cavities.*, Approximately correct, it is 808.
`falcon-180b`: *The European XFEL has a total of 16 superconducting radio-frequency (SRF) cavity stations that contain approximately 170 individual SRF cavities ...*, Wrong.
`ChatGPT`: *As of my last knowledge update in September 2021, the European XFEL (X-ray Free-Electron Laser) facility had a total of 100 superconducting radiofrequency (SRF) cavities ...*, Wrong, European XFEL has 100 modules.

Question 3 about beam emittance at REAGE : *What is the significance of measuring the beam emittance in the REGAE linear accelerator?*
`PACuna`:*Measuring the beam emittance in the REGAE linear accelerator is significant because it enables the calculation of the beam's emittance, which is a measure of the beam's transverse size and intensity. This information is crucial for optimizing the performance of the REGAE linear accelerator and the other devices it serves.*, Partially correct, word intensity is not correct.
`falcon-180b`: *Measuring the beam emittance in the REGAE linear accelerator is significant because it allows for the optimization of the accelerator's performance. Emittance is a measure of how tightly packed the particles in the beam are, and a smaller emittance means a more focused and uniform beam...*, Correct.
`ChatGPT`: *Measuring the beam emittance in a linear accelerator like REGAE (Relativistic Electron Gun for Atomic Exploration) is of significant importance because emittance is a fundamental parameter that characterizes the quality of the charged particle beam. The emittance measurement provides crucial information about the beam's properties, and it has several significant implications for accelerator operation and scientific experiments...*, Correct.

Experiments show the power of fine-tuning facility-specific questions, which is the main strength of our model. The main strength is the model's ability to answer facility-specific questions (Questions 1, 2), which is understandable considering the generality of the larger models. However, the other models provided a very good explanation for the third question, where `PACuna` provided misleading information.

7 Future Work

In the future, we plan to expand our data sources to incorporate more relevant and text-rich sources like Moskovic [2021], while simultaneously enhancing our data preparation techniques for the sources already used. Notably, some sources were regrettably excluded due to the absence of source files in the case of `arXiv` corpus. This can be mitigated by applying the same processing methodology successfully used for the JACoW papers. Additionally, we aim to generate more intricate and nuanced questions through specialized techniques like Ushio et al. [2023], refining existing prompts, or instruction mining Cao et al. [2023]. Currently, our generated questions are robust but generic since we always generate a fixed number of question-answer pairs per input, which may lead to ignoring some knowledge. We also plan a more detailed evaluation in future work with a human expert in the loop in the assessment of the generated data including hallucinations. What we found particularly crucial is preventing hallucinations, therefore in the future we would like to provide countermeasures against it either by scoring the answer or giving a source Tian et al. [2023].

Recently, work combining text, LLMs, and multi-modal data like images Wang et al. [2023] or even point sets Xu et al. [2023] has emerged. This is particularly interesting for particle accelerators which generate large amounts of multi-modal data. AI assistants then can for instance help with searching parameters (from point sets) or interpret the descriptions and images from logbooks.

8 Conclusion

In conclusion, this work demonstrates an approach for automating the fine-tuning of language models for specialized domains, using particle accelerator technology as an example application. We introduced `PACuna`, an LLM tailored for the particle accelerators through automated fine-tuning on a large corpus of domain-specific texts. The corpus was compiled from publicly available sources like books, conference proceedings, and pre-prints, providing both foundational knowledge and cutting-edge developments. Equations and tables were extracted and converted to machine-readable formats with Blecher et al. [2023] and question-answer pairs were automatically generated to create training data for fine-tuning. The model shows reasonable performance in addressing highly facility-specific problems in particle accelerators, as we show in the evaluation section, and is often on par with theoretical knowledge available with the pre-trained assistants.

The shown methodology provides a generalizable framework for creating domain-specific LLMs, overcoming some limitations of the general-purpose chatbots. This contributes a proof-of-concept for the feasibility of automated fine-tuning to align LLMs with rapidly advancing scientific fields. By concentrating knowledge and reasoning skills, PACuna and models derived through similar approaches can enhance accelerator R&D, control systems, collaboration, and education.

Acknowledgement

We acknowledge DESY (Hamburg, Germany), a member of the Helmholtz Association HGF, for its support in providing resources and infrastructure. Furthermore, we would like to thank all colleagues of the MCS and MSK groups and management for their contributions to this work and help in preparing this paper.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Maitha Alhammadi, Mazzotta Daniele, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. The falcon series of language models: Towards open frontier models. 2023.

Anthropic. Claude.ai, 2023. URL `https://www.anthropic.com`. Anthropic company website.

arXiv. arxiv.org e-print archive. `https://arxiv.org/`, 2023. Accessed: 2023-09-25.

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents, 2023.

Yihan Cao, Yanbin Kang, and Lichao Sun. Instruction mining: High-quality instruction data selection for large language models. *arXiv preprint arXiv:2307.06290*, 2023.

Yukang Chen, Shengju Qian, Haotian Tang, Xin Lai, Zhijian Liu, Song Han, and Jiaya Jia. Longlora: Efficient fine-tuning of long-context large language models. *arXiv:2309.12307*, 2023.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL `https://lmsys.org/blog/2023-03-30-vicuna/`.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

JACoW. Jacow. `https://www.jacow.org/`, 2023. Accessed: 2023-09-25.

John MacFarlane. `https://pandoc.org/index.html`, 2023. Accessed: 2023-09-25.

Micha Moskovic. The INSPIRE REST API. 12 2021. doi: 10.5281/zenodo.5788550. URL `https://github.com/inspirehep/rest-api-doc`.

OpenAI. Chatgpt: A large-scale generative language model for conversations. *OpenAI Blog*, 2021. URL `https://www.openai.com/blog/chatgpt`.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D Manning, and Chelsea Finn. Fine-tuning language models for factuality. *arXiv preprint arXiv:2311.08401*, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Asahi Ushio, Fernando Alva-Manchego, and Jose Camacho-Collados. A practical toolkit for multilingual question and answer generation. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Toronto, Canada, July 2023. Association for Computational Linguistics.

Wenhai Wang, Zhe Chen, Xiaokang Chen, Jiannan Wu, Xizhou Zhu, Gang Zeng, Ping Luo, Tong Lu, Jie Zhou, Yu Qiao, and Jifeng Dai. Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. `https://arxiv.org/abs/2305.11175)`, 2023.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. *arXiv preprint arXiv:1908.04319*, 2019.

Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. *arXiv preprint arXiv:2308.16911*, 2023.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] At present, we have not encountered a benchmark dataset in the published work that aligns closely with the aims of our paper.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] Most sources are publicly available on-line and we compile a list of sources on the paper's website.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] Retaining names, affiliations, and contact details in training data upholds norms of scholarly attribution and enables potential collaboration between researchers.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]