Explaining in Natural Language: A Discussion on Leveraging the Reasoning Capabilities of LLMs for XAI*

Arthur Picard[1,*,†], Yazan Mualla[1,†] and Franck Gechter[1,2,†]

[1]Université de Technologie de Belfort Montbéliard, UTBM, CIAD UR 7533, F-90010 Belfort, France
[2]Université de Lorraine, LORIA UMR CNRS 7503 SIMBIOT, 54506-Vandoeuvre-lès-Nancy, France

Abstract

This paper explores the potential application of Reinforcement Learning (RL) for reasoning in Large Language Models (LLMs) within the field of Explainable Artificial Intelligence (XAI). Deepseek recently introduced a training method that achieves strong reasoning capabilities in LLMs through unsupervised reinforcement learning on mathematical and programming problems. We discuss how a similar approach could be adapted for XAI by training a language model using the output of an existing model as ground truth. If the model converges successfully, it could replicate the outputs of the original model while also providing a natural language reasoning process leading to these outputs. While this method presents benefits such as in-depth natural language explanations and being model-agnostic, several challenges must be considered. These include the computational cost of training LLMs, the appropriate formatting of input data for different problem domains, the relevance of the relationship between the LLM and the original model, and identifying the specific applications where this method would be feasible and beneficial. We further discuss ideas such as downsizing the model, cost-effective training strategies, sequential fine-tuning, the inclusion of other XAI methods, and identifying relevant applications where this approach could provide the most value.

Keywords

Explainable Artificial Intelligence, Large Language Model, Natural Language Reasoning

1. Introduction

Ever since the release of Deepseek R1 [1], significant attention has been directed towards both this model, with over 400 citations in just two months, and towards the reasoning capabilities of Large Language Models (LLMs) in general. Discussions have centered on the model and its implications for specific domains [2], its broader impact [3], evaluations of its performance [4, 5], and advancements in key technical aspects such as Group Relative Policy Optimization (GRPO) [6, 7], among others. The field is rapidly evolving and has yet to reveal the full scope of its applications.

Explainable Artificial Intelligence (XAI) is a field that aims to provide techniques, models, and methods for developing XAI-based systems [8, 9, 10, 11]. These systems enable users and other human actors to better understand AI's decision-making, which, in turn, can improve factors such as trust and transparency [12, 13, 14], particularly in data-driven AI [15, 16, 17].

Meanwhile, Natural Language (NL) is the most commonly used way for humans to exchange information. As such, discussions on the use of NL for effective explanations in XAI have been ongoing [18], including the integration of generative AI advancements [19] and continuous interaction. [20] is a literature review that focuses on dialogue following an initial explanation, the frameworks required to set up such explanations, and methods for evaluating system performance.

We believe that recent advancements in reasoning LLMs should and will be leveraged for XAI. While this integration has the potential to yield highly valuable results, it is also constrained by several

Late-breaking work, Demos and Doctoral Consortium, colocated with the 3rd World Conference on eXplainable Artificial Intelligence: July 09–11, 2025, Istanbul, Turkey

*Corresponding author.
†These authors contributed equally.
arthur.picard@utbm.fr (A. Picard); yazan.mualla@utbm.fr (Y. Mualla); franck.gechter@utbm.fr (F. Gechter)
0009-0001-5476-6699 (A. Picard); 0000-0002-6772-6135 (Y. Mualla); 0000-0002-1172-603X (F. Gechter)

challenges. The goal of this paper is to present and discuss a methodology to apply an approach based on reinforcement learning for reasoning LLMs to XAI.

First, we introduce the motivation behind this work in Section 2, then the main idea in Section 3, followed by a discussion of the challenges and potential ways to alleviate them in Section 4. Finally, we conclude with an overview of our current work and relevant application domains in Section 5.

2. Motivation and Approach

Our primary idea is to apply Deepseek R1's training methodology [1] to develop a surrogate model capable of generating natural language reasoning that leads to the same decision. This natural language reasoning can then be used as explanation to better understand what could have lead the model to this decision. As a fully detailed reasoning going from known information to the AI's decision, this would allow human actor to identify unusual behaviors such as flaws in the logic thus fostering better Human-AI collaboration, or validate the AI's decision with a proper reasoning, leading to a more justifiably trust in the system. AI is capable to outperform human in certain task, but human cannot learn from them due to the models being opaque. As such, an other application is the use of the reasoning as learning tool. Having a reasoning exposed would allow human actors to naturally identify and learn the key logic behind the decision-making process, thus improving their own performance.

3. Methodology

The original training methodology employs a Reinforcement Learning (RL) training loop, where the LLM is evaluated on a database of mathematical and coding problems with known solutions. In this process, the model is rewarded solely based on adherence to the required output format (using "<think> reasoning <\think><answer> answer <\answer>") and the correctness of the final answer. This evaluation can be performed without human intervention, enabling large-scale RL.

3.1. Applying LLMs Reinforcement Learning to XAI

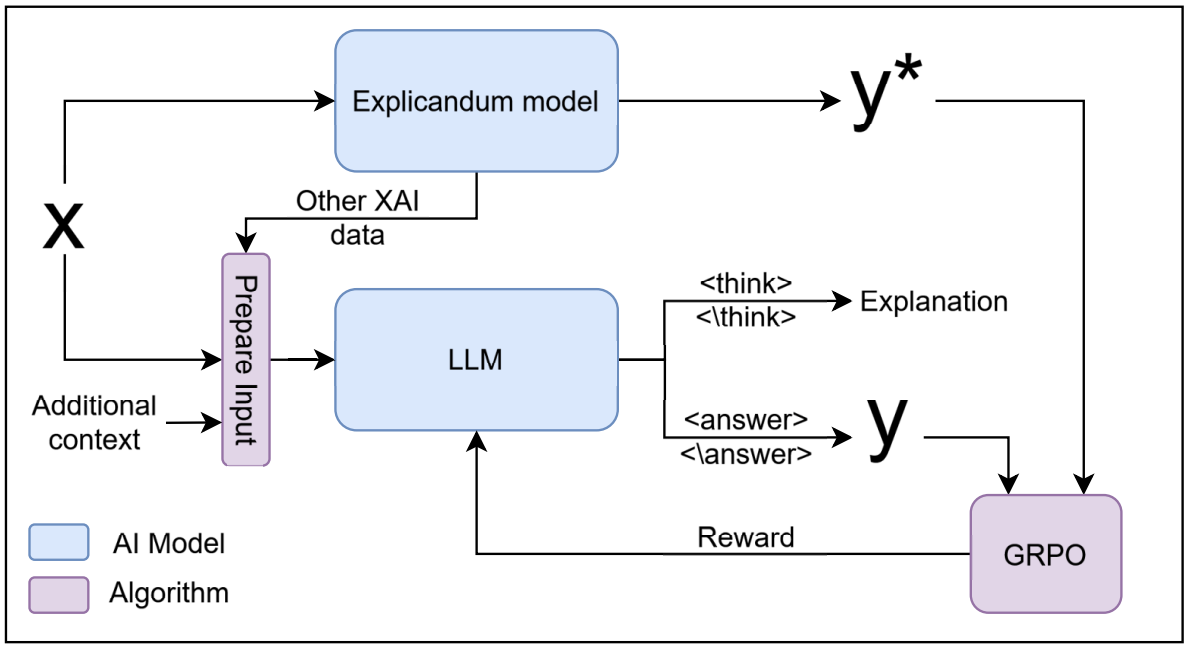

Figure 1: Schema of the Proposed Training Process.

Figure 1 illustrates the application of this training method to XAI. Assuming that we have an explicandum model (i.e., a model which is to be explained) with input-output pairs x,y*. Each instance of x,y* can be viewed as a problem-solution pair, which can then be used as input and output for the previously described training method. As the LLM converges, it effectively becomes a surrogate explicator model, producing the same final outputs as the explicandum model while generating a full NL reasoning process in the <think><\think> section. By fitting to the explicandum model, the explicator model may also learn flawed logic and reveal it within its reasoning.

To properly initiate training, the original input may need to be processed and reformatted to better suit a language model. Depending on the nature of the problem, this could be as simple as a script that adds appropriate labels to the input. More complex inputs can be handled using specialized tokenization, similar to how images are processed in recent multi-modal models [21, 22].

The same applies to the output. Since the final answer will be evaluated during RL, the model must generate responses in the correct format to fully benefit from the training process. Depending on the complexity of the problem, prompt engineering may be sufficient to achieve a properly formatted output, or a fine-tuning step may be required. Once the model can produce readable answers, reinforcement learning can begin.

3.2. Additional training steps

The formatting of the data into a proper input is essential for training and highly dependent on the specific problem at hand. Whenever possible, all relevant data should be incorporated in textual form to ensure seamless processing by the language model. However, if certain information cannot be naturally represented as text, such as images, structured numerical data, or complex graphs, an additional training step is required to handle these non-textual inputs effectively. To facilitate training, additional relevant information that may not have been usable by the original model can be included. Furthermore, additional XAI methods can be applied to the original model. Their outputs can be incorporated into the LLM's input, helping to improve training efficiency and better align the model with the explicandum.

The training process can be structured into multiple steps, which may be included depending on the complexity of the problem and the nature of the explicandum model:

1. **Fine-tuning to handle non-textual input data** – If the input includes non-textual elements such as images, graphs, or structured data which cannot be described in NL, an initial fine-tuning step is required to ensure the model can process and encode this information effectively.
2. **Fine-tuning on knowledge relevant to the context** – The model is trained on domain-specific knowledge to improve its understanding of key concepts, terminology, and factual information related to the target application. This step helps ensure that the model has the basis to be able to generate coherent reasoning.
3. **Fine-tuning on reasoning data relevant to the context** – As highlighted in Deepseek R1's paper, fine-tuning on reasoning data allows the model to internalize logical structures and patterns of thought. In our case, domain-specific reasoning data will strengthen the model's ability to generate coherent and accurate explanations.
4. **Final fine-tuning using the reinforcement learning-based method** – Once the model has a solid foundation, it undergoes the previously mentioned RL-based training process, aligning its decision-making with the explicandum model while generating NL reasoning.

In addition to format and accuracy rewards mentioned above, designing additional rewards to encourage or discourage specific behaviors will help achieve properly aligned explanations. In Deepseek R1's paper, this approach has been used to improve helpfulness and harmlessness, but can be applied to any behavior relevant to the problem at hand.

4. Challenges and limitations

While this method has the potential to provide in-depth, complex reasoning to justify a model's decision-making, several challenges must be overcome for this approach to be truly relevant.

4.1. What is truly explained?

Our objective is to explain a black-box model, however, we introduce another black-box model. LLMs are also known to exhibit unexpected behaviors. Typically, LLMs can have some transparency by tracing their outputs back to the original related data inside the training data [23]. However, recent findings suggest that fully comprehending the underlying mechanisms of LLMs is significantly more complex [24, 25]. Furthermore, the RL step encourages self-improvement without relying on the existing data, gradually diverging from the original training set, which may eventually make this traceability impossible.

Furthermore, the method does not directly explain the original model itself; rather, it creates a surrogate model that generates the same output together with a reasoning. A way to bring the LLM closer to the original model, could be, as previously mentioned, the usage of explainability methods, and the integration of their output as input of the LLM. Commonly used methods such as Lime or Shap, or, more effectively, NL-based explanations can be integrated, giving additional insight to both help the convergence and the fidelity to the original model. This additional information could also guide the LLM during the reinforcement learning process, helping it focus on the most relevant data for better convergence.

4.2. Adapted data and convergence

Certain types of data may be too complex for LLMs to handle effectively. While multimodal LLMs are being explored for other domains, existing methods typically focus on input data. However, in our case, this challenge extends beyond input data to also include output data. Multimodal output is also an active area research [22, 26], but these methods often involve calling separate models, which can create a disconnect between the reasoning process and the final output, making them less suited for this task.

Furthermore, the ability of LLMs to replicate highly complex models is inherently limited, as not fit for the task. As a result, they may struggle to converge, particularly when the reasoning behind certain models cannot be effectively translated into NL at a practical scale. The reasoning of some models is simply too complex to be fully captured within the constraints of the NL explanations, posing significant challenges to complete convergence.

4.3. Computational Cost

One of the main drawbacks of this approach is the high computational cost associated with training LLMs, which far exceeds that of traditional XAI methods. The computational cost remains one of the most significant challenges in working with LLMs. Although breakthroughs in efficiency are possible, current methods still require substantial resources. Several techniques can help reduce training costs, making this approach more viable.

4.3.1. Training: Fine-Tuning, GRPO, LoRA/QLoRA

Sequential training is a common approach that involves an initial training phase from scratch, followed by fine-tuning to adapt the model to specific tasks [27]. In our case, leveraging an existing reasoning model is the most logical choice, as it already aligns with the intention to generate structured reasoning outputs.

While full fine-tuning offers the best adaptability, it is computationally expensive. However, several techniques, widely adopted in both research and hobbyist communities, help reduce training costs while maintaining strong performance.

- **LoRA (Low-Rank Adaptation):** Reduces the number of trainable parameters, making fine-tuning more efficient [28].
- **Quantization:** Compresses model weights to lower bit precision, decreasing memory usage and computational cost [29].
- **QLoRA (Quantized LoRA):** Combines quantization with LoRA to further optimize efficiency [30].

For the RL step, methods such as Group Relative Policy Optimization (GRPO) [31], an evaluation method which bypasses the need for a critic model, saving a massive amount of training cost, or variations built upon it [6, 7], can be used, as demonstrated in DeepSeek R1. These techniques help refine the model's reasoning while keeping computational demands manageable.

4.3.2. Inference: Quantization and Distillation

Quantization is a suitable technique for optimizing inference, reducing computational costs by lowering the precision of model weights and activations. This makes large models more efficient without significantly compromising performance.

Distillation has proven to be effective for reasoning-focused LLMs [1]. It involves training a smaller model (the "student") to replicate the behavior of a larger model (the "teacher"). This significantly reduces computational requirements while preserving key aspects of the original model's reasoning capacity.

An additional approach that could be explored to focus on the generation of explanations is to incorporate the final answer as input during the distillation stage. Although this may seem counterintuitive, the reasoning behind it is that the teacher model must learn to reason without knowing the answer, ensuring that it develops a robust reasoning process. In contrast, the student model, which has more limited reasoning capabilities, is guided toward the correct answer and primarily learns the reasoning patterns from the teacher. This method could help the student model focus on explanation generation rather than answer derivation.

5. Application

5.1. Potential Fields

Given the length and depth of the explanations generated by this approach, it is best suited for scenarios with low time constraints and low cognitive load, where understanding the decision-making process is more important than speed.

This is particularly relevant in medical diagnostics, where accuracy is the most important factor. A well-explained diagnosis can help practitioners verify AI-generated conclusions. Similarly, legal analysis can benefit from AI models that provide transparent reasoning when reviewing contracts, regulations, or case law.

In fields like forecasting, whether economic, environmental, or demographic, AI-driven insights must be clearly justified to support informed policymaking and strategic planning. Financial analysis, especially in risk assessment and investment strategies, also requires explainable decision-making for regulatory compliance and trust.

Engineering and design optimization rely on iterative improvements, where understanding why a particular model or structure was recommended is as essential as the outcome itself. Teaching and education also naturally benefit from AI-generated explanations, as additional reasoning can enhance learning experiences and improve concept retention.

More broadly, this approach is well-suited to domains where NL explanations are often overlooked but could provide valuable context for AI-driven decisions.

While these applications show strong potential, handling input and output data remains a challenge, and each specific use case requires its own examination to ensure feasibility.

5.2. Current work

Our current research focuses on understanding chess AI decision-making by applying our methodology approach to chess engines. Chess has long served as a testing ground for AI research, as it provides a structured yet complex environment where AI performance can be rigorously tested. Our goal is to explain the decision making Stockfish, an open source and top performing chess engine which is able to, given a chess position, evaluate the game balance, and predict the most beneficial moves for each side. Additionally, this engine uses a configurable search depth, allowing us to control the complexity of decision-making and assess its impact on our approach.

Providing NL explanations for high-level chess AI moves could also benefit the chess community, as many advanced engine decisions remain difficult to interpret. To achieve this, we follow a sequential fine-tuning process, where the model progressively learns different aspects of the game:

- **Understanding the board state** – Answering fundamental questions such as *"Where is piece X?"* or *"Is piece X still on the board?"* ensures the input format is properly interpreted.
- **Understanding game rules** – Determining *"What are X's available moves?"* helps the model internalize the legal move set.
- **Assessing board dynamics** – Identifying *"Give the evaluation of the current game state?"* or *"What is the most impactful piece on the board?"* provides a first step in understanding strategic positioning.
- **Predicting the next best move** – Generating *"What is the best move?"* is the first and most critical step toward engine-level prediction.
- **Expanding the prediction scope** – Answering *"What are the next best moves?"* aligns the model with true engine outputs.

We anticipate limitations when attempting to verbalize deep search results, as a core component of chess engine is the analyses of millions of positions, and top human players often mention intuition when discussing their decisions. This suggests that words alone may not always fully capture the reasoning behind certain moves. Adding feature based explainability methods to the input should help alleviate this problem, as it directly guides towards the most relevant part of the game.

5.3. Future Work

Looking ahead, several directions could improve this approach. One idea is enabling continuous interactions, so the model can answer user questions and provide additional clarification in real time. Another is testing the impact of scalability, to see if the method works for larger, more complex models can be handled and deeper insights may be provided. It is also worth exploring how knowledge distillation can be improved, with the idea of including the final answer in the input to guide explanations. Choosing the right base model for explainability remains an open question, is it possible to design models for XAI which may perform better than general ones. Finally, adding multimodal capabilities, like integrating images into explanations, could make them clearer and more intuitive. These directions offer exciting opportunities to refine the approach and advance the field.

6. Conclusion

In this paper, we explored the potential application of Reinforcement Learning (RL) for reasoning in Large Language Models (LLMs) within the context of Explainable Artificial Intelligence (XAI). By adapting Deepseek R1's training methodology, we proposed an approach where LLMs could potentially generate natural language explanations for complex decision-making, replicating the outputs of existing black-box models while providing reasoning that enhances transparency.

We discussed various aspects of the methodology, including the sequential fine-tuning process to improve the model's understanding of problem domains, challenges related to multimodal inputs and

outputs, and the computational costs of training large models. These components form the foundation of our approach, which, if successful, could enable more interpretable AI systems capable of producing contextually relevant and human-understandable explanations.

While our work is still in its early stages, the next steps involve refining the methodology and evaluating its applicability across different domains. In particular, future work will need to address the challenges of scaling this approach to more complex models, handling multimodal inputs, and improving training efficiency. The eventual goal is to develop AI systems that can offer more transparent, understandable, and trustworthy decision-making, though further exploration and experimentation are required to assess the feasibility and impact of this approach in real-world applications.

Declaration on Generative AI

During the preparation of this work, the author used Chat GPT-4o to improve readability and style. After using these tools, the author reviewed and edited the content as needed and assume full responsibility for the content of the publication.

References

[1] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al., Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, arXiv preprint arXiv:2501.12948 (2025).

[2] S. Mercer, S. Spillard, D. P. Martin, Brief analysis of deepseek r1 and its implications for generative ai, SuperIntelligence-Robotics-Safety & Alignment 2 (2025).

[3] W. A. Hayder, Highlighting deepseek-r1: Architecture, features and future implications (2025).

[4] L. Faray de Paiva, G. Luijten, B. Puladi, J. Egger, How does deepseek-r1 perform on usmle?, medRxiv (2025) 2025–02.

[5] G. Mondillo, S. Colosimo, A. Perrotta, V. Frattolillo, M. Masino, Comparative evaluation of advanced ai reasoning models in pediatric clinical decision support: Chatgpt o1 vs. deepseek-r1, medRxiv (2025) 2025–01.

[6] Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, T. Fan, G. Liu, L. Liu, X. Liu, et al., Dapo: An open-source llm reinforcement learning system at scale, arXiv preprint arXiv:2503.14476 (2025).

[7] C. Li, N. Liu, K. Yang, Adaptive group policy optimization: Towards stable training and token-efficient reasoning, arXiv preprint arXiv:2503.15952 (2025).

[8] A. Picard, Y. Mualla, F. Gechter, S. Galland, Human-computer interaction and explainability: Intersection and terminology, in: World Conference on Explainable Artificial Intelligence, Springer, 2023, pp. 214–236.

[9] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, D. Pedreschi, A survey of methods for explaining black box models, ACM computing surveys (CSUR) 51 (2019) 93.

[10] Y. Mualla, I. Tchappi, T. Kampik, A. Najjar, D. Calvaresi, A. Abbas-Turki, S. Galland, C. Nicolle, The quest of parsimonious XAI: A human-agent architecture for explanation formulation, Artif. Intell. 302 (2022) 103573. URL: https://doi.org/10.1016/j.artint.2021.103573. doi:10.1016/j.artint.2021.103573.

[11] P. Hemmer, M. Schemmer, M. Vössing, N. Kühl, Human-ai complementarity in hybrid intelligence systems: A structured literature review., PACIS (2021) 78.

[12] A. Glass, D. L. McGuinness, M. Wolverton, Toward establishing trust in adaptive agents, in: Proceedings of the 13th international conference on Intelligent user interfaces, 2008, pp. 227–236.

[13] Q. V. Liao, D. Gruen, S. Miller, Questioning the ai: informing design practices for explainable ai user experiences, in: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 2020, pp. 1–15.

[14] A. Bunt, M. Lount, C. Lauzon, Are explanations always important? a study of deployed, low-cost

intelligent interactive systems, in: Proceedings of the 2012 ACM international conference on Intelligent User Interfaces, 2012, pp. 169–178.
[15] D. Gunning, Explainable artificial intelligence (XAI), Defense Advanced Research Projects Agency (DARPA), nd Web (2017).
[16] O. Biran, C. Cotton, Explanation and justification in machine learning: A survey, in: IJCAI-17 workshop on explainable AI (XAI), 1, 2017, pp. 8–13.
[17] W. Samek, T. Wiegand, K.-R. Müller, Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models, arXiv preprint arXiv:1708.08296 (2017).
[18] E. Cambria, L. Malandri, F. Mercorio, M. Mezzanzanica, N. Nobani, A survey on xai and natural language explanations, Information Processing & Management 60 (2023) 103111.
[19] J. Yu, A. I. Cristea, A. Harit, Z. Sun, O. T. Aduragba, L. Shi, N. Al Moubayed, Interaction: a generative xai framework for natural language inference explanations, in: 2022 International Joint Conference on Neural Networks (IJCNN), IEEE, 2022, pp. 1–8.
[20] D. Mindlin, F. Beer, L. N. Sieger, S. Heindorf, E. Esposito, A.-C. Ngonga Ngomo, P. Cimiano, Beyond one-shot explanations: a systematic literature review of dialogue-based xai approaches, Artificial Intelligence Review 58 (2025) 81.
[21] P. Xu, X. Zhu, D. A. Clifton, Multimodal learning with transformers: A survey, IEEE Transactions on Pattern Analysis and Machine Intelligence 45 (2023) 12113–12132.
[22] J. Wang, H. Jiang, Y. Liu, C. Ma, X. Zhang, Y. Pan, M. Liu, P. Gu, S. Xia, W. Li, et al., A comprehensive review of multimodal large language models: Performance and challenges across different tasks, arXiv preprint arXiv:2408.01319 (2024).
[23] S. Chen, F. Kang, N. Yu, R. Jia, Fasttrack: Reliable fact tracing via clustering and llm-powered evidence validation, in: Findings of the Association for Computational Linguistics: EMNLP 2024, 2024, pp. 5821–5836.
[24] E. Ameisen, J. Lindsey, A. Pearce, W. Gurnee, N. L. Turner, B. Chen, C. Citro, D. Abrahams, S. Carter, B. Hosmer, J. Marcus, M. Sklar, A. Templeton, T. Bricken, C. McDougall, H. Cunningham, T. Henighan, A. Jermyn, A. Jones, A. Persic, Z. Qi, T. Ben Thompson, S. Zimmerman, K. Rivoire, T. Conerly, C. Olah, J. Batson, Circuit tracing: Revealing computational graphs in language models, Transformer Circuits Thread (2025). URL: https://transformer-circuits.pub/2025/attribution-graphs/methods.html.
[25] J. Lindsey, W. Gurnee, E. Ameisen, B. Chen, A. Pearce, N. L. Turner, C. Citro, D. Abrahams, S. Carter, B. Hosmer, J. Marcus, M. Sklar, A. Templeton, T. Bricken, C. McDougall, H. Cunningham, T. Henighan, A. Jermyn, A. Jones, A. Persic, Z. Qi, T. B. Thompson, S. Zimmerman, K. Rivoire, T. Conerly, C. Olah, J. Batson, On the biology of a large language model, Transformer Circuits Thread (2025). URL: https://transformer-circuits.pub/2025/attribution-graphs/biology.html.
[26] S. Wu, H. Fei, L. Qu, W. Ji, T.-S. Chua, Next-gpt: Any-to-any multimodal llm, in: Forty-first International Conference on Machine Learning, 2024.
[27] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al., Improving language understanding by generative pre-training (2018).
[28] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al., Lora: Low-rank adaptation of large language models., ICLR 1 (2022) 3.
[29] R. Krishnamoorthi, Quantizing deep convolutional networks for efficient inference: A whitepaper, arXiv preprint arXiv:1806.08342 (2018).
[30] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, Qlora: Efficient finetuning of quantized llms, Advances in neural information processing systems 36 (2023) 10088–10115.
[31] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, et al., Deepseekmath: Pushing the limits of mathematical reasoning in open language models, arXiv preprint arXiv:2402.03300 (2024).