Deliberate Minds, Smarter Moves: Optimizing Agent Actions through Synthetic Reasoning

Gheorghe-Adrian Dina
AI Multimedia Lab
National University of Science and Technology POLITEHNICA Bucharest
Bucharest, Romania
gheorghe.dina@stud.aero.upb.ro

Bogdan Ionescu
AI Multimedia Lab
National University of Science and Technology POLITEHNICA Bucharest
Bucharest, Romania
bogdan.ionescu@upb.ro

***Abstract*—Recent developments in large language models reveal strong reasoning skills but are hindered by their reliance on static knowledge, which can result in inaccuracies. In a solution proposal, the integration of real-time actions, such as API queries or web searches, can emulate human reasoning by allowing models to obtain and use up-to-date information during processing. This dynamic approach is crucial for LLM-based agents operating in changing environments, promising improved accuracy and decision-making by enabling models to gather essential information when needed.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

In recent years, large-scale language models have shown exceptional performance, suggesting the substantial potential for human-like intelligence [1]–[6]. This potential emerges from the utilization of extensive training datasets in conjunction with a sizable number of model parameters. Consequently, a burgeoning field has emerged centered on the use of LLMs as central control systems to develop autonomous agents capable of decision making and action generation [7].

In this context, the article proposes a new version of model that simulates human-like thinking processes with a tool called, in this process of reasoning. The model is finetuned using a synthetic version of an established dataset, which comprises three distinct elements: question, reasoning with and without tool use, and result, having a starting point the GSM8K [8] and HotpotQA [9] datasets. The QLoRA [10] fine-tuning strategy is used to perform the optimization process. Inserting action calls (calculator and web-search queries) during the reasoning process, rather than before generation, has been shown to significantly increase the precision and relevance of models' responses. The contributions beyond state-of-the-art can be summarized with the proposed action-in-the-loop architecture that solves parts of major LLM difficulties by enabling just-in-time adaptive retrieval and processing, ultimately supporting more accurate decision-making, nuanced responses, and robust interaction patterns for agentic systems.

II. Proposed approach

In this research, we will approach the possibility of training a reasoning model to use different tools through API calls. Each API call is represented as a function call with three variables, input, query and type of API, and one output, representing the response of the function [11]. That requires that inputs and output for each API can be represented as text sequences. For that, we generate a synthetic dataset corresponding to training requirements.

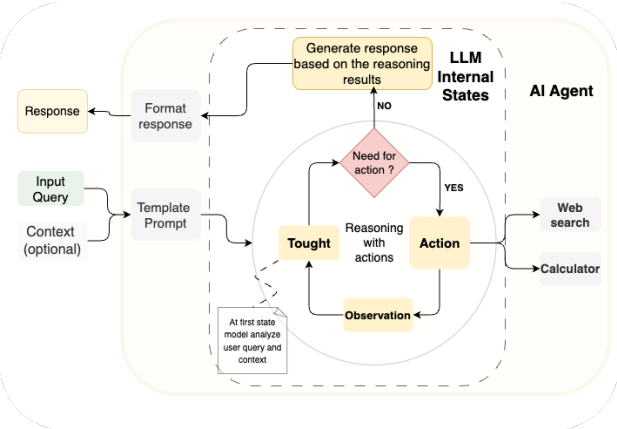

Fig. 1: Proposed AI Agent workflow with actions in the reasoning process.

A. Dataset generation

In that case, we need to have a data set $D = \{x_1, x_2, ..., x_n\}$ that can be transformed into this format $D^* = \{(x_1, r_1, a_1), (x_2, r_2, a_2), ..., (x_n, r_n, a_n)\}$ to fine-tune where x_n is the question and the true response of the ground, r_n is the reasoning process, $a = f(type, q) \rightarrow r$ is a function that takes as arguments $type$ as a type of action (e.g. Math or QA), q as a query for the tool, and r as a response [12]. Compared with other strategies, our model will be finetuned so that the process of generating the right response is a part of the reasoning process augmented with the tool response.

Initially, the model is finetuned using a synthetic version of an established dataset, which comprises three distinct elements: question, reasoning with and without tool use, and result, having a starting point the GSM8K [8] and HotpotQA [9] datasets. GSM8K is used to assess model mathematical

2nd Workshop on Artificial Intelligence for Multimedia, AI Multimedia Lab, CAMPUS Research Institute, POLITEHNICA Bucharest, Bucharest, Romania, 15 July, 2025

reasoning and problem-solving capabilities. HotpotQA, on the other hand, is a question-answering dataset specifically crafted to promote multihop reasoning in responses.

In that case, the model is trained to use two type of API, **Websearch** and **Calculator**. For the Web search API, we employ the external Tavily API, while the calculator functionality is implemented using Python code. These data sets represent a starting point for the generation of synthetic data, as shown in Figure 2 . Additionally, within the reasoning model's framework, a prefix prompt is appended to the training data to ensure that the model adheres to instructions and complies with a predefined format.

The process of adding reasoning in the dataset is based on DeepSeek R1 [6] response from asking to response to a question from HotpotQA and GSM8K in a specific format.

The next step is to insert an action into the reasoning process. For that we used two types of tools, the Calculator for GSM8K equations and the Websearch for HotpotQA questions.

At the end of the generation we compute the cross-entropy loss from data with [3] and without [**??**] function calls and combine them based on the condition that the function call makes it easier for the model to predict future tokens [11].

$$D^* = \{d_1, d_2, \ldots, d_n\} \text{ where} \tag{1}$$

$$d_i = \begin{cases} (x_i, r_i, a_i) & \text{if } L(x_i, r_i, a_i) < L(x_i, r_i) \\ (x_i, r_i) & \text{else others} \end{cases} \tag{2}$$

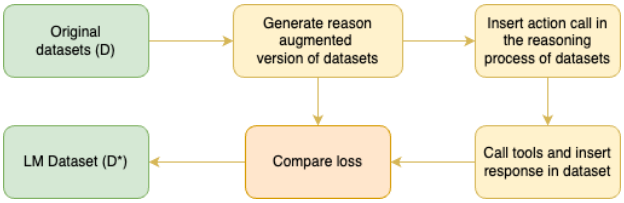

Fig. 2: Steps to generate data based on original dataset and transform with reasoning model and action calling.

B. Model finetuning

The experiments will be performed with a base model LlaMA-3.1-8B and LlaMA-3.1-70B [5]. For the efficiency of the optimization process, we approach the **QLoRA** (Quantized Low-Rank Adapter) [10]. Finetuning very large models is prohibitively expensive and recently quantization method reduce the memory footprint of LLMs. QLoRA [10] is a parameter-efficient fine-tuning method that enables large language models (LLMs) to be fine-tuned on consumer hardware. It leverages:

- 4-bit quantization of the base model,
- frozen base weights,
- trainable low-rank adapters (LoRA).

TABLE I: Architectural components for Llama 3.1–8B and Llama 3.1–70B models.

Component	Llama 3.1–8B	Llama 3.1–70B
Number of Layers	32	80
Hidden Size	4096	8192
Feedforward Size (MLP)	∼14336	∼22016
Attention Heads	32	64
GQA Heads	—	8
Activation Function	GEGLU	GEGLU
Normalization	RMSNorm	RMSNorm
Positional Encoding	RoPE	RoPE
Context Length	8K	8K+ (scalable)
Parameters	∼8 Billion	∼70 Billion

III. Experiments

Instruction fine-tuning techniques will be used to perform the optimization process, using a standard language modeling objective. This technique is a process where models are fine-tuned on a dataset of tasks with human-provided directives or instructions. This method aims to enhance the model's ability to follow natural language commands and improve its performance on a diverse range of tasks.

Prompt:
Below is an instruction that describes a task. Write a response that appropriately completes the request. Keep in count if input is provided.
`Input: ...`
`Instruction: ...`
`Reason: ...`
`Response: ...`

Fig. 3: Instruction fine-tuning prompt template.

An maximum sequence length of 2048 tokens will be configured and an batch size of 16 samples with gradient accumulation equal to 4 and 60 epochs. LoRA is applied to attention projection layers and feedforward layers.

For evaluation, GSM8K, HotpotQA, and TriviaQA [13] were used, only the evaluation parts of each.

TABLE II: Using the web search and calculator tool for most of examples, Llama 3.1-8b-react and Llama 3.1-70b-react models clearly outperforms baselines of the same size in the evaluation of 10% from the dataset.

Model	GSM8K	HotpotQA	TriviaQA
Llama 3.1-8b-Instruct	57.5	40.1	46
Llama 3.1-70b-Instruct	70.5	67.1	70
Llama 3.1-8b-react (ours)	**67.2**	**65.8**	**59**
Llama 3.1-70b-react (ours)	**75.2**	**72.8**	**78**

IV. Conclusions and Future Development

In this research, we show that fine-tuning a large model to reason with function call can improve performance over factual learning or math solving. The dataset, consisting of 5000 samples with action calls in the reasoning process, was

obtained by generating text with the DeepSeek-R1 reasoning model. Further research could focus on developing more sophisticated reasoning models that can seamlessly integrate tool calls during the decision-making process.

- **Diverse Tool Integration:** Adding domain-specific APIs (e.g., weather, maps, finance).
- **Scalable Dataset Generation:** Automating synthetic generation for broader domains.
- **Preference-based Reinforcement Learning:** Applying fine-tuning methods such as DPO or GRPO.

REFERENCES

[1] Y. Li, S. Bubeck, R. Eldan, A. Del, G. Suriya, G. Yin, and T. Lee, ") phi-1.5 (1.3b) phi-1.5 (1.3b) phi-1.5 (1.3b) phi-1.5 (1.3b) phi-1.5-web (1.3b) phi-1.5-web (1.3b) phi-1.5-web (1.3b) phi-1.5-web (1.3b) phi-1.5-web (1.3b) falcon-rw-1.3b falcon-rw-1," 2023.

[2] H. Touvron, L. Martin, and K. Stone, "Llama 2: Open foundation and fine-tuned chat models," 7 2023. [Online]. Available: http://arxiv.org/abs/2307.09288

[3] G. Team, "Gemma 2: Improving open language models at a practical size," 7 2024. [Online]. Available: http://arxiv.org/abs/2408.00118

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 5 2020. [Online]. Available: http://arxiv.org/abs/2005.14165

[5] A. Grattafiori and A. Dubey, "The llama 3 herd of models," 7 2024. [Online]. Available: http://arxiv.org/abs/2407.21783

[6] DeepSeek-AI, "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning," 1 2025. [Online]. Available: http://arxiv.org/abs/2501.12948

[7] W. Chen and Z. Li, "Octopus v2: On-device language model for super agent," 4 2024. [Online]. Available: http://arxiv.org/abs/2404.01744

[8] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman, "Training verifiers to solve math word problems," 10 2021. [Online]. Available: http://arxiv.org/abs/2110.14168

[9] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," 9 2018. [Online]. Available: http://arxiv.org/abs/1809.09600

[10] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," 5 2023. [Online]. Available: http://arxiv.org/abs/2305.14314

[11] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," 10 2022. [Online]. Available: http://arxiv.org/abs/2210.03629

[12] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," 2 2023. [Online]. Available: http://arxiv.org/abs/2302.04761

[13] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension," 5 2017. [Online]. Available: http://arxiv.org/abs/1705.03551