iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VI **Month of publication:** June 2025

DOI: https://doi.org/10.22214/ijraset.2025.72225

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VI June 2025- Available at www.ijraset.com

An Overview on the Performance of Reasoning Agents in Large Language Models

Nathan Man[1], Harish Senthilkumar[2], Ryan Li[3], Samarth Prajapati[4], Pranav Shankar[5], Ethan Shen[6], Tanusha Tamijet[7]
[1, 2, 3, 4]*Valley Christian High School, BASIS Independent Silicon Valley, BASIS Independent Fremont, Archbishop Mitty High School*
[5, 6, 7]*Leigh High School, Cupertino High School, Monte Vista High School*

Abstract: ***The recent rise of Large Language Models (LLMs), which are able to generate human-like text, has put a large amount of attention onto AI and its potential uses. However, most LLMs are limited to a one-dimensional/leftto-right method of decision-making that can impede their performance in tasks that require accurate foresight and reference to previous decisions to execute. We hypothesize that various types of LLM reasoning agents have different strengths and weaknesses that allow for applications for different strategic use cases. In our research, we hope to determine the specific use cases and strengths of various reasoning agents, which will allow for the creation of LLMs tailored towards certain tasks with the use of such agents. With the help of reasoning agents, such as symbolic, arithmetic, and chain-of-thought reasoning, LLMs adopt a greater understanding of the context given to them and use a multi-step approach to adequately solve problems. Existing challenges in evaluating reasoning agents within LLMs include issues such as dataset biases and the potential brittleness of the model. These challenges, combined with the ethical concerns surrounding the reasoning agents such as their susceptibility to amplifying biases within a response, offer a rich research area. Using a quantitative analysis of several reasoning agents within a controlled environment, we apply diverse multi-modal and iterative reasoning techniques. Through this analysis, we explore the strengths and weaknesses of these reasoning techniques, resulting in a better understanding of the reasoning capabilities to be applied to real- world scenarios and products.***

Keywords: ***large language models, decision-making, reasoning agents, dataset biases, quantitative analysis, iterative reasoning techniques***

I. INTRODUCTION

Reasoning Agents for LLMs stem from diverse backgrounds. They each have their strengths and weaknesses, tailored towards obtaining optimal performance to solve a certain set of tasks. Many reasoning agents have already shown exceptional capability in yielding high performance results. Surpassing standard prompting in terms of solve rate (%) on three different data sets, Chain-of-thought (CoT) prompting is a reasoning agent that is steadfast in performing arithmetic, commonsense, and symbolic reasoning tasks. In essence, it prompts a strategy where the model is encouraged to solve problems by breaking the reasoning process up into a series of intermediate steps. This allows the model to perform better on complex tasks such as multi-step arithmetic, logic puzzles, and common sense reasoning because it shows its process of getting there. Another approach includes zero-shot reasoning, which adds phrases such as “Let’s think step by step” to the prompt itself. The upside of zero-shot reasoning is that when it’s used, a model can produce structured thought processes when guided with subtle cues which makes it a powerful reasoning agent though it may not reach the same performance on more complex tasks. Moreover, it’s easy to implement due to the fact that it doesn’t require any additional data or fine-tuning. LLMs can think on a step by step basis to perform complex reasoning tasks on a much simpler set of steps. Much like zero- shot reasoning, Tree-of-Thought (ToT) prompting that enhances an LLMs capability for complex reasoning through generating and evaluating multiple reasoning paths instead of a single, linear chain. This approach allows for the model to reason deeper with more flexibility, which is especially crucial when solving problems with multiple viable solution paths. Unfortunately, it is computationally more expensive due to evaluating multiple reasoning agents. With reasoning agents like those mentioned, LLMs have the potential to solve all sorts of complex problems and questions. If designed efficiently and for a bigger purpose, reasoning agents can greatly foster the performance of LLMS, and form a breakthrough in the development of the machine learning field. Their unlimited potential and revolutionary nature hold key in deciphering reasoning tasks faster and with higher precision.

In this paper, we evaluated multiple models’ performances across different reasoning datasets after implementing different reasoning agents. The four models tested were Llamma 2, Llamma 3, ChatGPT 3.5, and Gemini 1.0 Pro.

The first reasoning agent used was zero-shot Chain-of-thought (CoT), a hybrid of Zero-Shot and Chain-of-Thought, in which the model is prompted to approach problems in a procedural and step by step manner, which can result in an increase in accuracy, especially for arithmetic reasoning tasks. It leverages the benefits of CoT without requiring examples to train making it a lightweight yet effective reasoning agent.

Another reasoning agent we tested was Automatic Chain-of-thought (Auto-CoT). Similar to regular CoT, Auto-CoT approaches problems in a step by step manner. However, the dataset of questions fed through the model are clustered beforehand into groups, with the model generating its own example reasoning process for different types of questions. This can result in an overall better performance as the model can take different more appropriate approaches to solve different types of problems more accurately.

The third reasoning agent tested was Self Consistency. We leveraged a few shot Self Consistency reasoning agent in our tests, meaning five example questions and answers were given to the model for reference. Self Consistency builds upon typical Chain of Thought reasoning by generating multiple reasoning paths with the most common solution being chosen as the final prediction. This helps reduce randomness in generation and ensures a more reliable final answer, particularly when CoT outputs can vary slightly which reduces incorrect outputs. In order to work, the model needs to be prompted multiple times.

II. METHODOLOGY

1) Data Preparation: Gather multiple datasets relevant to natural language understanding tasks. These datasets may include text classification, question-answering, text generation, and others. Ensure the datasets are diverse and representative of real-world applications.
2) Baseline Testing on LLM: Run each dataset on the selected LLM(ChatGPT 3.5 - Turbo) to establish a baseline performance in terms of accuracy. Record the results for each task.
3) Incorporation of Reasoning Agents: Integrate the reasoning agent Automatic Chain of Thought Ensure that these agents are appropriately fine-tuned and compatible with the LLM architecture.
4) Testing with Reasoning Agents: Run all datasets on the LLM with reasoning agent integrated. For each dataset, compare the performance of the LLM with reasoning agents to the baseline performance without reasoning agents. Record the results for each task.
5) Rinse and Repeat: Re-run the tests with other reasoning agents, including reAct, and Self Consistency in Chain of Thought, redo the previous 2 steps for each reasoning agent, and record results accordingly
6) Data Analysis: Analyze the collected data to compare the performance of the LLM with reasoning agents and without reasoning agents. Calculate accuracy metrics for each task and reasoning agent combination.
7) Comparison of Results: Compare the accuracy of the LLM with reasoning agents (Auto Cot, ReAct, Self-Consistency in Chain of Thought) to the accuracy without reasoning agents. Determine the magnitude of impact that each reasoning agent had on the LLM, and in which area each reasoning agent would be useful for.
8) Conclusion: Summarize the findings and draw conclusions regarding the effectiveness of reasoning agents on Large Language Models for various natural language understanding tasks.

III. RESULTS

		LLAMA-2	Gemini 1.0	Llama 3	Llama 3.1 8b	Total Questions
GSM8K	Questions Correct	307	293	1010	327	1319
	Percent Correct	23.28	22.21	76.57	24.79	1
	0-shot cot	137	1001	1001	1043	1319
	%	10.39	75.89	75.89	79.08	1
	autocot	382	1064	948	1028	1319
	%	28.96	85.67	71.87	77.94	1
	self consistency	417	1065	1117	1177	1319
	%	31.61	80.74	84.69	89.23	1
	RAG	4		663		1319
	%	0.3		50.27		1
CSQA	Questions Correct	321	956	787	837	1319
	%	24.34	72.48	59.67	63.46	1
	0-shot cot	397	954	821	891	1319
	%	30.1	72.33	62.24	67.63	1
	autocot	510	805	847		1319
	%	38.67	61.03	64.22		1
	self consistency	425	1021	950		1319
	%	32.22	77.41	72.02		1
	RAG	289		471		1319
	%	21.91		35.71		1

Fig. 1 Data on Accuracy of Models

The data collected from running the GSM8K dataset on grade school math word problems on GPT-3.5 with CoT reasoning can be utilized to quantify the efficacy of CoT reasoning. Using CoT reasoning methods, GPT-3.5 correctly outputted 5472 correct answers out of 7274 grade school math word problems processed from the GSM8K dataset, yielding an accuracy of 75.2(%). This is in comparison to the base GPT-3.5 model, which has shown to give an accuracy of only 16.8(%). The implementation of step-by-step reasoning chains through CoT reasoning is shown to significantly increase the effectiveness of the GPT-3.5 model. Given the efficacy of CoT reasoning, future development of reasoning agents can incorporate CoT methodology to further improve upon the performance of GPT-3.5 and potentially other models. As depicted by the results, every single reasoning agent changed the accuracy of the large language models (LLMs). The base accuracy when using the GSM8K dataset was 23.28% on Llama-2, 22.21% on Gemini 1.0, 76.57% on Llama-3, and 24.79% on Llama 3.1 8b. When using 0-shot cot as the reasoning agent on each model while still using the GSM8K dataset , the accuracy of Llama-2 decreased to 10.39%, the accuracy of Gemini 1.0 increased to 75.89%, the accuracy of Llama-3 decreased to 75.89%, and the accuracy of Llama 3.1 8b increased to 79.08%. When using autocot as the reasoning agent on each model while still using the GSM8K dataset , the accuracy of Llama-2 increased to 28.96%, the accuracy of Gemini 1.0 increased to 80.67%, the accuracy of Llama-3 decreased to 71.87%, and the accuracy of Llama 3.1 8b increased to 77.94%. When using self consistency as the reasoning agent on each model while still using the GSM8K dataset , the accuracy of Llama-2 increased to 31.61%, the accuracy of Gemini 1.0 increased to 80.74%, the accuracy of Llama-3 increased to 84.69%, and the accuracy of Llama 3.1 8b increased to 89.23%. Finally, when using RAG as the reasoning agent on each model while still using the GSM8K dataset, the accuracy of Llama-2 decreased to 0.3% while the accuracy of Llama-3 decreased to 50.27%. The base accuracy when using the CSQA dataset was 24.34% on Llama-2, 72.48% on Gemini 1.0, 59.67% on Llama-3, and 63.46% on Llama 3.1 8b. When using 0-shot cot as the reasoning agent on each model while still using the CSQA dataset, the accuracy of Llama-2 increased to 30.10%, the accuracy of Gemini 1.0 decreased to 72.33%, the accuracy of Llama-3 increased to 64.24%, and the accuracy of Llama 3.1 8b increased to 67.63%. When using autocot as the reasoning agent on each model while still using the CSQA dataset, the accuracy of Llama-2 increased to 88.67%, the accuracy of Gemini 1.0 decreased to 61.03%, and the accuracy of Llama-3 increased to 64.22%. Lastly, when using RAG as the reasoning agent on each model while still using the CSQA dataset, the accuracy of Llama-2 decreased to 21.91% and the accuracy of Llama-3 decreased to 35.71%.

IV. CONCLUSION

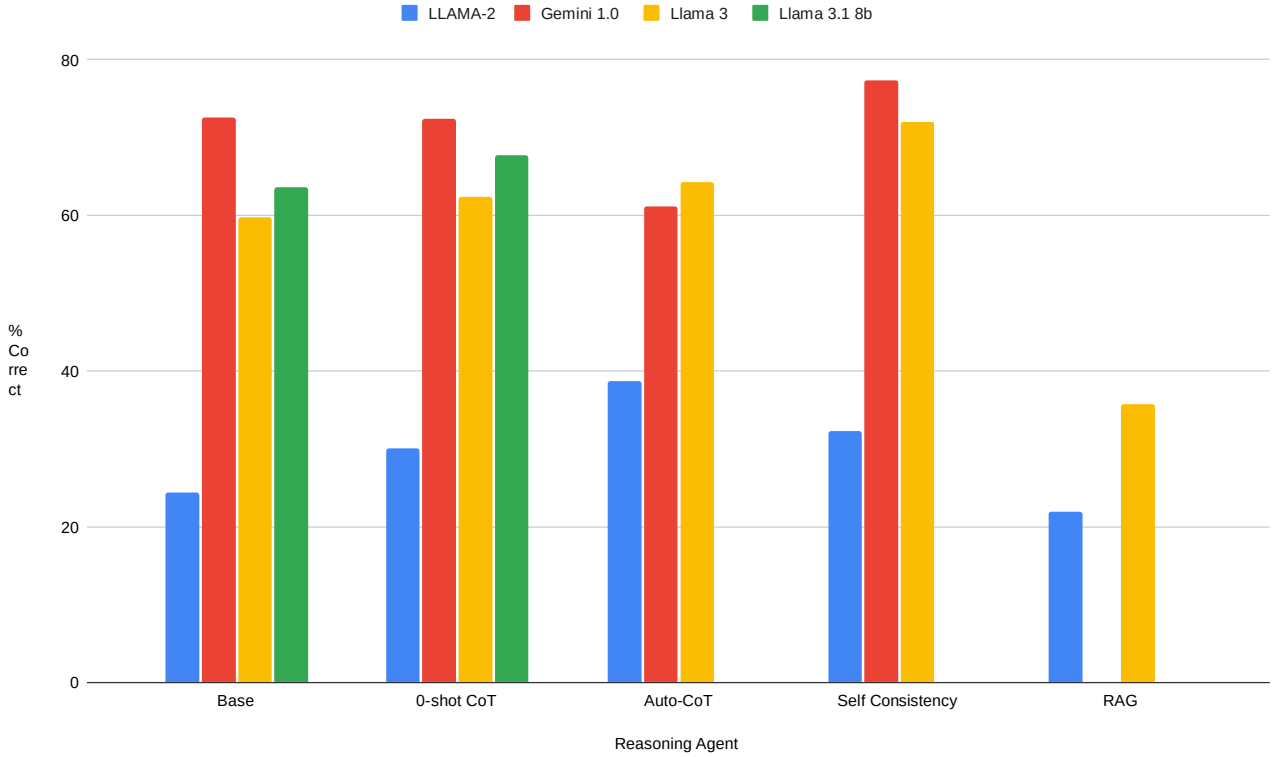


Fig. 2 Graphical Data on CSQA

Fig. 3 Graphical Data on GSM8K

Across all tested models - Llama-2, Llama-3, Gemini 1.0, Gemini 1.5, and Llama 3 8B - structured reasoning methods AutoCoT and Self-Consistency consistently improved accuracy over the base models. Self-Consistency improved the performance of Llama-2, Llama-3, Gemini 1.0, and Llama 3 8B for the GSM8K dataset and improved the performance of Llama-2, Gemini 1.0, and Llama-3. With the significant increase in accuracy across both the datasets, Self-Consistency demonstrates the ability for structured mathematical reasoning as well as common sense based, open-ended reasoning, suggesting strong reasoning capabilities and adaptability across diverse problem types. Autocot performed quite well on the GSM8K dataset as Llama-2 saw a sizable improvement in accuracy while Gemini 1.0 and Llama 3.1 8b saw a massive boost in accuracy. Only in Llama 3 did autocot see a dip in performance, thus indicating that across most LLMs, autocot improves the model's structured mathematical reasoning. On the CSQA dataset, autocot improves the accuracy of Llama3 slightly, improves the accuracy of Llama-2 by a drastic margin, and sees a sizable dip in Gemini 1.0 which indicates that in general, autocot boosts the ability of an LLM for open ended reasoning. Autocot's propensity for improving the mathematical and open-ended reasoning makes it reliable for diverse problem types on most LLMs. On the other hand, 0-shot cot saw an inconsistent performance on the GSM8K dataset depending on the LLM.With a sizable decrease in accuracy in Llama-2, negligible decrease in accuracy in Llama 3, and massive increases in accuracy in Gemini 1.0 and Llama 3.1 8b, 0-shot cot shows inconsistencies in its ability to apply structured, step-by-step mathematical reasoning, potentially due to limitations in problem decomposition or numerical manipulation. Conversely, with the CSQA dataset, 0-shot cot performed much better seeing a very negligible dip in performance with Gemini 1.0 and a sizable increase in accuracy with Llama-2, Llama-3, Llama 3.1 8b. This exhibits 0-shot cot improves commonsense reasoning in most models, though one model may have inherent limitations. Lastly, RAG performed poorly regardless of the dataset as for GSM8K and CSQA, Llama-2 and Llama-3 saw sizable decreases in accuracy. This suggests that RAG's approach is ineffective across both mathematical and commonsense reasoning tasks, potentially due to a lack of generalization of RAG.

V. DISCUSSION AND FUTURE WORKS

As we tested several different models, there continues to be newer and more improved versions of Large Language Models(LLMs) continue to be made. Something to explore further is how these more recent versions of Llama, Chat GPT, Gemini, and more compare to past versions of models and whether or not reasoning agents may affect them differently than their counterparts. Furthermore, certain reasoning agents such as Retrieval Augmented Generation (RAG) and Automatic Chain of Thought (AutoCOT) performed poorly or gave mixed results when run on different LLMs.

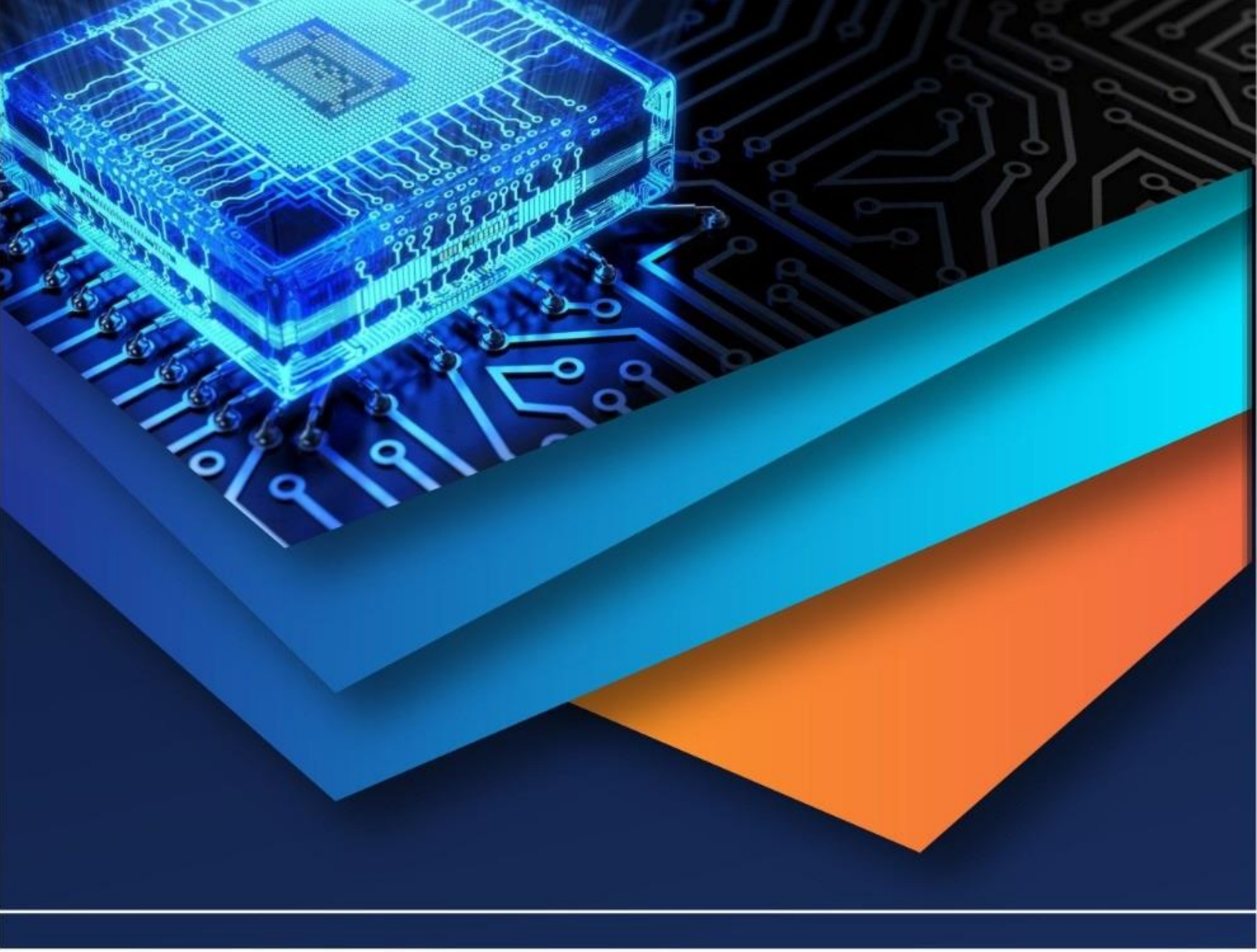
We could explore the reasons for these differences and whether a different approach, such as hybridization of reasoning agents where several reasoning agents are used collaboratively to improve performance, could improve accuracy. In this study, the only measurement we took was the accuracy of each model with an associated reasoning agent, but we could have compared more to get a more accurate representation of the performance of the models. Reasoning agents act as different thought processes for models to understand, and with these different thought processes come in varying steps to the final answer. We could further analyze exactly what step-by-step solutions work best for which models. Our research gives a fundamental understanding of the benefits and limitations of reasoning agents with Large Language Models, and this could be a step towards researching more complex structures and systems of reasoning agents that have the potential to further improve the accuracy of a model. In addition, another potential direction for future work is fine-tuning LLMs with specific reasoning strategies or on datasets that incorporate various reasoning agent patterns. This could potentially allow model behavior to more closely match the stepby-step reasoning sequences, which would improve consistency and performance across tasks. Fine-tuning could also allow models to better integrate hybrid reasoning approaches. Our research provides us with an initial overview of the pros and cons of reasoning agents with LLMs, and this could be one step towards exploring more complex structures and frameworks of reasoning agents that have the potential to improve the accuracy of the model.

VI. ACKNOWLEDGMENT

The authors would like to thank the Aspiring Scholars Directed Research Program (ASDRP) for their support and express their gratitude to Dr. Phil Mui, and Mr. Suresh Subramaniam for their mentorship and guidance throughout this project.

REFERENCES

[1] Yao, Shunyu, et al. “Tree of Thoughts: Deliberate Problem Solving with Large Language Models.” 17 May 2023 (2305.10601.pdf (arxiv.org))
[2] Wei, Jason, et al. “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models” 28 January 2022 (2201.11903.pdf (arxiv.org))
[3] Wang, Xuezhi, et al. “Self-Consistency Improves Chain of Thought Reasoning in Language Models” 7 March 2023 (2203.11171.pdf (arxiv.org))
[4] Yao, Shunyu, et al. “ReAct: Synergizing Reasoning and Acting in Language Models” 6 October 2023 (2210.03629.pdf (arxiv.org))
[5] Mialon, Gregoire, et al. “Augmented Language Models: a Survey” 15 February 2023
[6] (2302.07842.pdf (arxiv.org))
[7] Huang, Jie, et al. “Towards Reasoning in Large Language Models: A Survey” 20 December 2022 (2212.10403.pdf (arxiv.org))
[8] Hao, Shibo, et al. “Reasoning with Language Model is Planning with World Model” 24
[9] May 2023 (2305.14992.pdf (arxiv.org))



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY