Reproduction Project: ”Advancing Spatial Reasoning in Large Language Models: An In-Depth Evaluation and Enhancement Using the StepGame Benchmark”

Edward Sonny Bakane
matrícula: 0124015814-27D

June 2024

Abstract

In this work we are reproducing work which focuses on improving large language models (LLMS) GPT and stepGame in spatial reasoning. A state of the art solution was presented. We reproduce the work to check if the results that were presented are meaningful. The solution is able to reason with the provided English state provide location or position letter or objects in a tree structure (Figure 5).

Keywords: Artificial intelligence, AI, large language models, ChatGPT-3.5 Turbo, atGPT4 wa, LLMs, reason

1 Introduction

Our work is to reproduce the article by Li et al [1], which is entitle: 'Advancing Spatial Reasoning in Large Language Models: An In-Depth Evaluation and Enhancement Using the StepGame Benchmark'

The current work researched on a solution to improve chatGPT cognition capability to be able to reason and interact with humans in spatial communication. Also, to comprehend the surrounding with the effort to ensure human user immersion when communicating with the bots, perhaps during the time of emergency. Several technology were tested against chatGPT3 to chatGPT4. ChatGPT4 was able to stable it performance at 90%. This remarkable performance was a result of the current work being able to rectifying errors which were previous ignore.

The proposed solution is composed of the sentence to relation mapping with logic based spatial reasoning. The combination of StepGame, chatGPT and large language models (LLMs) fused together to come up with a brilliant solution is said to be game changer for artificial intelligence (AI) in spatial reasoning.

2 Methodology

2.1 Methods

Solution for the Corrected Benchmark According to the Li et al [1] claimed that the logic-based is error free. The approach uses the template to address sentence to relation mapping through semantic parsing using closely related statements. ASP reasoner is paramount for reasoning behavior in this work.

1. Sentence-to-Relation Mapping. In this work, presented description r is check with the resources currently available in the template. The template is equipped with resources such as this $o_{0\_}v_{\_o_1}$. To extract the location and verify that o_i, has not been used anywhere, $k-hop$ is used. When the initial object is $(i=0)$, it set to o_0, while the middle object is set to $(i \geq 1)$. The prompt is then triggered to look for o_i.

2. The prompt mechanism assists large language models (LLMs) to carry out the mapping procedure on the spatial relations description.

3. The computation of the coordinate is done by initializing the o_0 to $(0,0)$ as well as spatial relation is instantiated to an offset to process the location of the object using the equation "$O_{i+1} = O_i + offset(r^j) = (x_{oi}, y_{oi}) + (x_v, y_v) = (x_{o_{(i+1)}}, y_{o_{i+1}})$".

Tree-of-Thoughts (ToT) Prompting In this work, algorithm 1 assumes the responsibility of improving the reasoning chain building process. As it does, it grant LLMs to pursue other avenues. In a case where obstructing mapping may arise while performing lookup for the relation and object, LLMs can be triggered to handle the problem. One of the interesting functions of the algorithm 1, is that it triggers LLMs to setup the tree structure from a default state ensuring that clearance for current state is complete.

- Thought Generation: LLMs are permitted to initiate the proposal of the j thought generation prompt while focusing in utilizing all the unused relations to enumerate all potential expansions of the chain making used of the relations that are not currently used with the experiment. Fig. 1 represents the algorithm for the thought generation approach illustration:

Algorithm 1: Our ToT Approach

Require: LLM, input x

$S_0 \leftarrow Init(x)$
$i \leftarrow 1$
while no $s_f \in S_{i-1}$ has arrived at o_t **do**
  $S_i^{'} \leftarrow \{s{\cdot}c | c \in G(s,j) \wedge ChainExtn(c) \wedge s \in S_{i-1}\}$
  if $S_i^{'} = \emptyset$ **then return** failure
  $S_i \leftarrow select(b, \{\langle s, y\rangle | s \in S_i^{'} \wedge y = \Sigma_1^n \sigma(V(s))\})$
  $i = i + 1$
end while
return $Link(s_f)$

Figure 1: Thought generation approach illustration

In this experiment, when $j = 2$, it instructs the LLMs to produce the time two.

- State Evaluation: Li et al [1], state that their approach consist of classification methodology which make use of the designed value prompt that attentively checks if the chain is able to reach the target. The chain expansion may have reach the target or not. It reached, perfect but if not yet reached target the assumption is that it is likely complicated to more especially if there are unused relations. Furthermore, It has been established that prompt directs the LLM through sequential step by step to explore recently produced states $s \in S$. This employs the LLM stochastic procedure that excludes the zero temperature for increasing the reliability scoring. Some output such '*sure*', '*likely*' and '*impossible*' and transformed into numerical values through a function $Q()$ in order to permit the selection procedure with all the new created states.

- Search Algorithm: In this research, breadth first search is used with the depth of the tree limited to 10 ($depth = 10$) and the with restricted to '$b = 3$'. During the training, with $b = 3$ the procedure terminates immediately when the connecting chain reaches the targeted object.

According to Li et al [1], their approach builds chain from O_0 all the way to O_0. Their spatial relation amongst such objects are computed based on the old CoT prompting technique using the c^{map} as well as c^{calcu}.

2.2 Experimental Design

2.2.1 Model Settings

This research utilizes the Azure OpenAI service for chatGPT-3.5-Turbo, GPT-3 and GPT-4 API access. For more reliable concentrated as well as deterministic results, the have initialized the temperature to 0 in the CoT experiment. The ToT experiments set the the comperature at 0.7 default to produce different thought proposals.

Different Test Subsets: According to Li et al [1], they have used 30 or 100 test examples for a complete set of 10,000 for every k^{th} value. However, they have acknowledged that this technique is pron to irregularities that may introduce biases.

Moreover, this experiment explores the effects of these number of test examples. Their main focus is to verify the impact of limited number of test examples on the accuracy of the results. They conducted tests on a clean test set which was separated for ($k \in [1.10]$) which resulted in covering a range of task complexities. So, they performed test 30, 100 and 1000 test examples check the effect number of the experiment examples on the evaluation.

Different Few-Shot Sets: Three different few shot were fired that triggered sets to evaluate how input examples influence prompts:

- $clean5shot(1,3,5,7,10)$: They created a single prompt composed of five examples, thus include 1-hop, 3=hop, 5-hop, 5-hop and 10-hop reasoning.
- $clean10shot$: Then a formulation of a prompt which utilizes 10 examples, everyone taken from a certain k-hop from a clean set.
- $clean5shotseparate$: It builds a prompt for everyone k-hop reasoning task, using the previous 5 examples from corresponding k-hop training set as few shot examples.

3 Reproduction

3.1 Experimental Results

3.1.1 Evaluation Results

Influence of Scale of Test Examples Li et al [1] stated that their results were based on the clean 10shot. When evaluating the expanded test set which was made up of 1000 examples, the model demonstrated a uniform decline in performance as k raise from 1 to 10. The assumption is that the results may be due to the increment of complexity as the number of hops go up. however, test sets involving 100 and 30 examples have shown results which are less consistence. The author assumed that smaller small sizes provided may have a pivotal role in these fluctuation. Hence, the believe larger datasets may provide a more reliable test bed.

Influence of Prompting Examples: Li et al [1] explain that subplot demonstrates that the decision of choice of prompting strategy may affect the model's capability to handle tasks of different complexity differently. It is further stated that as the number of hops significantly go up there is a notable decline in accuracy. These 3 techniques produce close results. Even though differences are there on specific hop levels, there is no notable single technique that performs better than the rest. Refer to the table: 2.1

According to the authors, clean 5shot (1,3,5,7,10) has shown to outperform clean 10shot (1 10) in every hop level. Their results interpretation suggest that it is essential to pick examples from a wider spectrum of hop levels instead of relying on each hop level.

	$left/right$	$above/below$	$lower_left/upper_right$	$lower_right/upper_left$
total	44	53	50	53
text-curie-001	11	41	30	37
text-davinci-003	0	0	0	2
gpt-3.5-turbo	2	2	3	1

Table 1: The relation extraction performance of GPT. The numbers in rows 2-4 are incorrect predictions numbers

Influence of Models: This research deduced comparison between GPT-3, turbo and Davinci models. The possible reason may, it lacks in reading understanding, part of speech, as well as extracting relation tasks which may be as a results of its smaller model size.

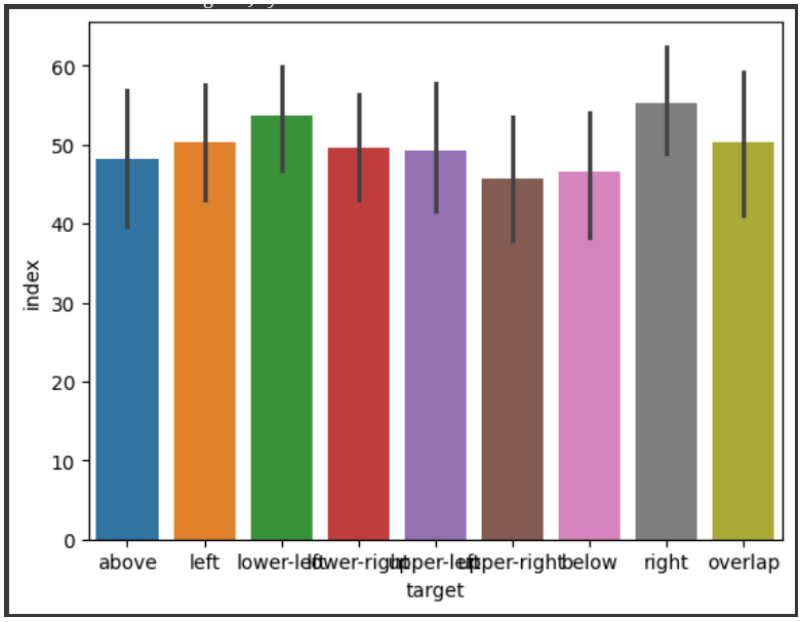

Figure 2: GPT-3.5 Spatial Targets Success

However, in stepGame spatial reasoning task needs the understanding of sequential connections as well as drawing of the deductions from them. Even though Davinci outperforms turbo, the difference is notably minimal in terms of performs. The more the increment in complexity levels they seem to broadly meet each other halfway. However the reproduction results illustrates that GPT-3.5 turbo and Davinci results are inseparable, both managing to reach the hightest of 61 each as shown in Figure 2 and 3.

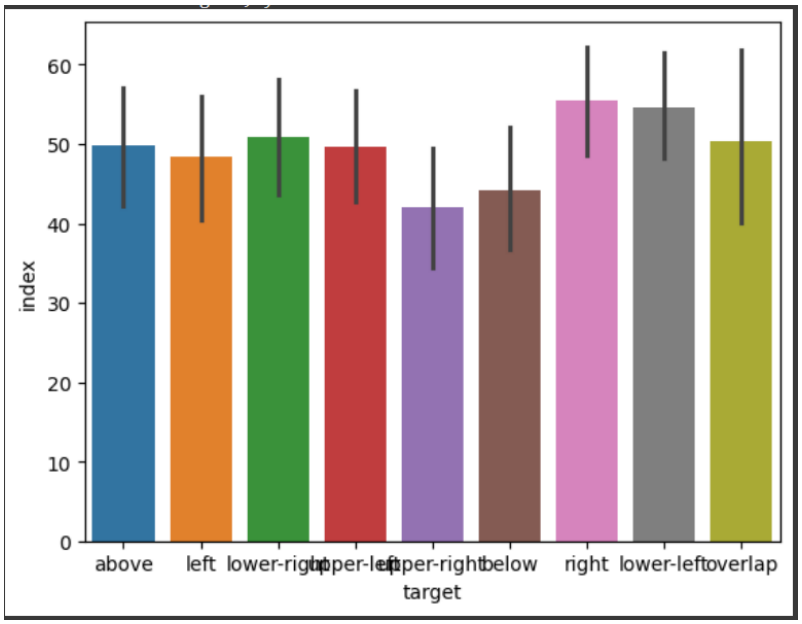

Figure 3: Davinci Spatial Targets Success

Despite, GPT-4 being the most advance, is struggling in spatial recognition. Figure 4 is showing the GPT-4 scoring the lowest results. Its highest achieved score is 17.5.

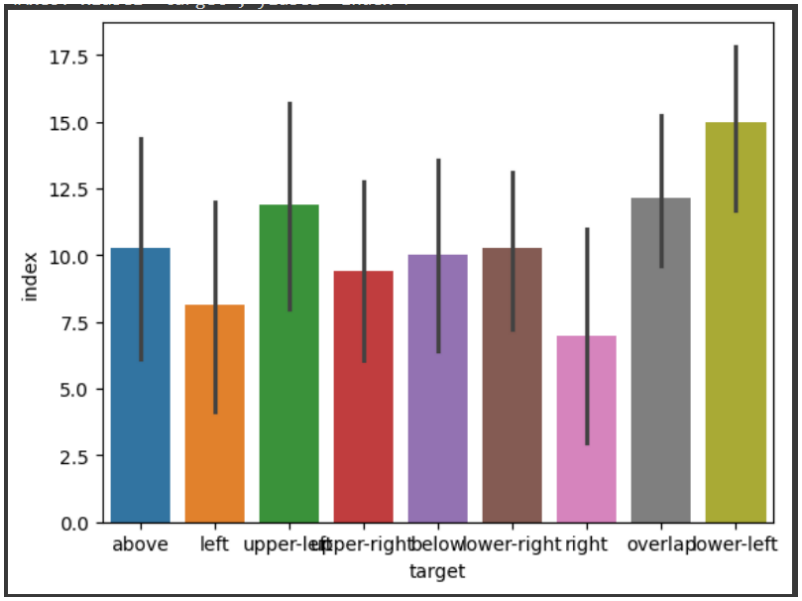

Figure 4: GPT-4 Spatial Targets Success

3.1.2 Results of the Improved Methods

Resolution for the Benchmark: According to the author, the results of mapping and reasoning illustrate the accuracy in scores and high scores represent better performance. Their improvement have resulted into achieving correct

spatial relation mapping as well as multihop spatial reasoning with no errors discovered. Illustration is shown in Figure 5. It able to describe the location of I from E, position of E from W, and L from Y respectively.

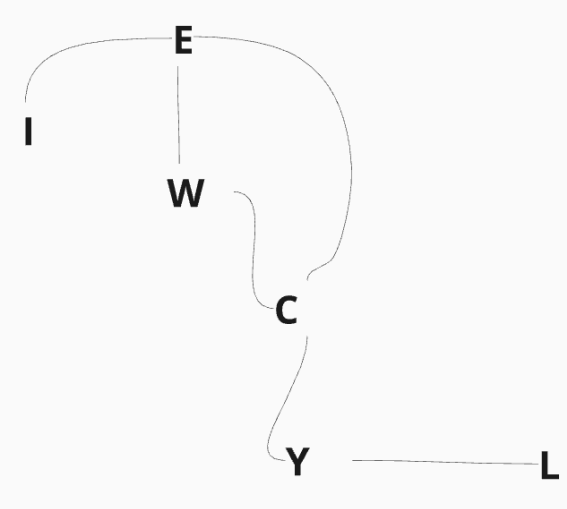

Figure 5: Spatial Objects positions diagram

GPT for Relation Extraction + ASP for Reasoning: When analysing GPT performance on the relation extraction subtask, they noted that Curie has highest number of incorrect prediction in all relations while Davinci and Turbo are doing well.

CoT and ToT: The authors stated that GPT-4 and tree-of-thoughts (ToT) achieved better result of a test set which consist of 20 instances. Davinci and GPT-3.5 Turbo used a larger test set with 100 examples. The results for both the base and CoT methods were achieved through 5$shot separate$ from the clean set.

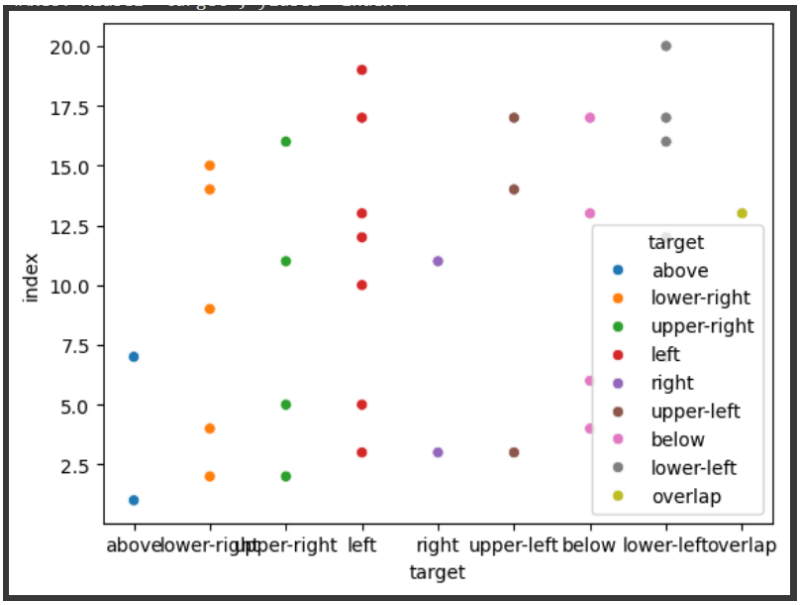

Figure 6: CoT_GPT-4 Spatial Targets Success

The results of ToT_CoT in this section are of GPT-4 linkage chain. Afterwards Turbo, Davinci and GPT-4 were hooked together for CoT reasoning behind the built connections. GPT-4's dominance proven to be outperform its competitors across all problem set. However, GPT-4 struggles as level of complexity is increased.

CoT and ToT approach have contributed to a better performance of the GPT-4 model for complex tasks which run from $k = 2tok = 10$. According to Li et al [1], their presented ToT and CoT amplified the robustness of performance of the Davinci and GPT-4 especially with hops that are larger. Turbo model does not do very well. This maybe due to this work's prompts which requires knowledge of the coordinates and relations.

With all the computed confidence interval with *CI* 95% for the *target* attribute for spatial recognition in the stepGame *right* in the experiment scored the highest confidence interval of (23.89, 31.11). This may mean that in the experiment *right* was the interpreted with multihops. Also, $lower - right$ is further recognised.

Activity	Confidence Interval with 95%
left	(11.72, 18.94)
upper-right	(7.39, 14.61)
lower-right	(21.79, 29.01)
above	(6.89, 14.11)
lower-left	(16.89, 24.11)
right	(23.89, 31.11)
below	(12.39, 19.61)
overlap	(4.39, 11.61)
upper-left	(19.72, 26.94)

Table 2: CoT and ToT approach CI with 95%

4 Conclusions

This research presented a more improved approach GPT-4 and Davinci for reasoning and mapping spatial relations. In this paper, Turbo did not perform well compared to the newly devised approach. The reproduction results that the state of the art solution is able to provide spatial reasoning. We performed the confidence interval with 95% for the target attribute "left" it has it proven that indeed the state of the art solution meets the requirements of spatial reasoning when provided with a set state.

References

[1] Fangjun Li, David C Hogg, and Anthony G Cohn. Advancing spatial reasoning in large language models: An in-depth evaluation and enhancement using the stepgame benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18500–18507, 2024.