Thought Space Explorer: Navigating and Expanding Thought Space for Large Language Model Reasoning

Anonymous ACL submission

Abstract

Recent advances in large language models (LLMs) have demonstrated their potential in handling complex reasoning tasks, which are usually achieved by constructing a thought chain to guide the model to solve the problem with multi-step thinking. However, existing methods often remain confined to previously explored solution spaces and thus overlook the critical blind spot within LLMs' cognitive range. To address these issues, we design the Thought Space Explorer (TSE), a novel framework to expand and optimize thought structures to guide LLMs to explore their blind spots of thinking. By generating new reasoning steps and branches based on the original thought structure with various designed strategies, TSE broadens the thought space and alleviates the impact of blind spots for LLM reasoning. Experimental results on multiple levels of reasoning tasks demonstrate the efficacy of TSE. We also conduct extensive analysis to understand how structured and expansive thought can contribute to unleashing the potential of LLM reasoning capabilities.

1 Introduction

Recent advances in large language models (LLMs) have shown great potential in solving complex tasks with reasoning capabilities (Huang and Chang, 2022; Achiam et al., 2023; Patterson et al., 2022) by guiding the LLMs to reasoning logically solve the complex task step-by-step. A common practice is to design the Chain-of-Thought (CoT) (Kojima et al., 2022) to boost reasoning capabilities by evolving the thinking from single outputs to intermediate steps reasoning.

Existing studies (Wang et al., 2022; Yao et al., 2024; Zhang et al., 2024c; Besta et al., 2024) attempt to develop various thought structures with multiple chains or branches of thought on top of CoT to arouse the LLM reasoning ability. Compared with direct output and CoT, the core advantage of thought structures enables models to explore the solution space of a task from local to global (Hao et al., 2023). For example, in Figure 1, thought structures may initiate exploration from two distinct points *"specialty"* and *"industry"*. Such exploration allows LLMs to generate diverse paths to solutions and thus enhances reasoning capacity. Moreover, the diverse structures can enable models to perform forward and backward evaluations within the explored thought space to seek the optimal solution, i.e., a more effective reasoning thought.

A series of studies are conducted to optimize thought structures with various aspects, including generating parallel thought (Wang et al., 2022), constructing tree structure reasoning topology on top of CoT (Yao et al., 2024), fine-tuning the LLMs with direct preference optimization (DPO) to align thought steps of CoT with task performance (Zhang et al., 2024b), etc. The key idea of this work is to compare multiple responses or extending existing chains (e.g. *"Coffee over Drone"* or *"Coffee industry → Coffee bottle industry"*) to obtain a better thought chain. However, these approaches lack exploration of unknown solution spaces, which might lead to consistent oversight of LLMs' cognitive blind spots. Merely generating more chains does not enable LLMs to conceive of content previously unthought of. This is because the over-generated chains tend to repeat previous thought patterns and would lead to two main issues: (1) The absence of feasible solutions, when feasible solutions are in blind spot space, repeatedly filtering or extending existing paths may converge to a local optimum (e.g., exploring only from a *coffee* perspective). (2) Insufficient diversity. Especially for open-ended questions, existing methods have limited impact on exploring the thought space, and excessive extension or filtering might even reduce the diversity of responses (e.g., discarding feasible solutions or creating redundancy through repetitive thinking).

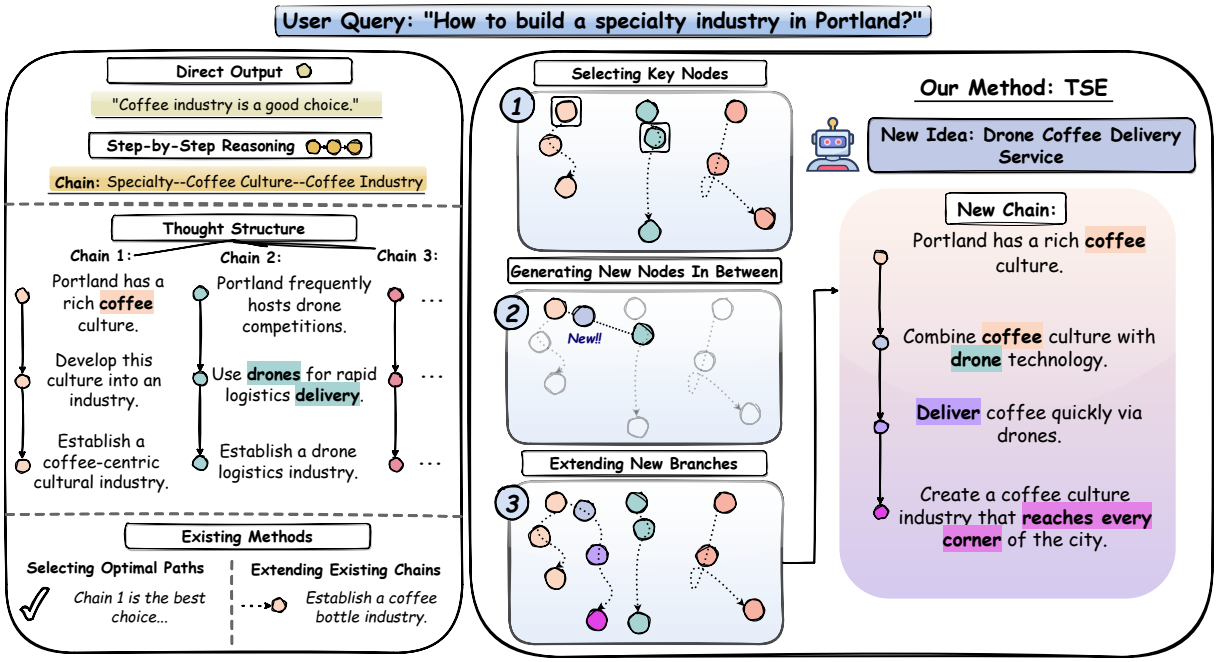

Figure 1: Thought structure optimization through TSE. On the left side, we showcase traditional thought structures and optimization methods, where the LLM's generation may limited by its thought pattern. On the right side, we show how TSE method expands thought structure through a three-step generation of branches. Our method guides the LLM to explore the blind spots between previous thought paths.

To address these issues, we propose the **T**hought **S**pace **E**xplorer (TSE), a novel framework designed to expand and optimize thought structures. The TSE starts from already explored thought paths and guides the model to explore hidden solution spaces because the existing thought structures often already contain feasible solutions or crucial information pointing towards such solutions. To enhance efficiency and precision, further exploration of the model starts from thought nodes within explored solutions, which ensures that the reasoning process is not a blind exploration but a deeper inquiry based on verified insights.

Concretely, to identify key points of information from existing thoughts, we first quantify the contribution of each thought node to the conclusion during the model's reasoning process to select key thought nodes (e.g., in Chain 2, the details about *"drones and delivery"* serve as key information leading toward *"logistics industry"*). Considering the visibility of parameters in LLMs, we select key nodes from two perspectives, relative gradients and semantic relationships. Further, based on these key nodes, the model generates new thought nodes and proceeds with deeper reasoning in new directions from "original node" to "new nodes" and facilitates the exploration of the solution space by the thought structure. Finally, depending on whether the reasoning tasks require a singular or comprehensive conclusion, we proceed with collaborative reasoning across the entire thought structure to generate the output. We evaluate the effectiveness of TSE on three reasoning tasks and the results show that TSE significantly improves the performance of thought structures compared with existing methods. We further analyze the effectiveness of each component to comprehensively understand their functionality in each stage.

Our contributions are summarized as follows: (1) We propose TSE reasoning framework to expand thought structures for exploring solution spaces to alleviate the impact of blind spots for LLM reasoning. (2) We investigate various strategies to prioritize and refine the thought structure by identifying the importance of the node in the thought structure. Different strategies are designed to adapt to various settings. (3) Experimental results on three specific reasoning tasks indicate the effectiveness of our TSE compared with the existing reasoning methods without exploring thought structure.

2 Related Work

2.1 LLM Reasoning Structures

The most straightforward method to address reasoning tasks is to generate a conclusion through one-step thinking. However, the LLM might overlook essential intermediate steps and generate incoherent logic or incorrect conclusions (Chu et al.,

2023). The advent of CoT (Wei et al., 2022; Wang and Zhou, 2024) optimizes the reasoning step by connecting distinct thoughts into a coherent sequence (Li et al., 2024). Although CoT can improve transparency and coherence, its singular structure limits its capability to handle more complex logical relationships (Jin and Lu, 2024; Sprague et al., 2024). To this end, some studies develop structured reasoning methods, such as self-consistent CoT and Tree-of Thought. (Wang et al., 2022; Zhang et al., 2024c; Yao et al., 2024; Liu et al., 2024; Mo and Xin, 2024; Zhang et al., 2024a). These sophisticated thought structures enhance the consistency and systematic nature of reasoning by expanding the ability of the model to manage diverse logical relationships (Xia et al., 2024; Stechly et al., 2024). Thought structures offer distinct advantages by maintaining coherence and depth while increasing the diversity and flexibility of reasoning paths(Liang et al., 2024). However, the reasoning chains within these structures are highly repetitive and thus reduce generation and selection efficiency.

2.2 Thought Structure Optimizations

To further enhance the capabilities of thought structures, recent researches focus on two main optimization strategies. The first is selecting optimal paths within the structures (Feng et al., 2023; Long, 2023; Hao et al., 2023; Shinn et al., 2023; Jung et al., 2022). By choosing optimal paths, the model filters the irrelevant and low-quality branches and globally searches for correct or optimal solutions, thus enhancing reasoning efficiency and accuracy. The second is expanding reasoning depth and breadth (Zhu et al., 2022; Besta et al., 2024; Gao et al., 2024; Zhang et al., 2024a,b; Hou et al., 2024). By deepening and widening the thought structure, the model can explore a broader array of possibilities and perspectives, thus improving its understanding and capability to solve complex issues. However, these methods might be limited to previously explored spaces and directions, thus failing to adequately investigate the blind spots within the thought space of the model. Different from them, we aim to expand the depth and breadth of thought structures by actively exploring the cognitive blind spots of the model in thought space.

3 Methodology

To expand and optimize thought structures for reasoning tasks, we introduce TSE, a self-expanding and exploring method to enable LLMs to proactively address deficiencies in reasoning processes and explore new reasoning directions with limited generation. We implement the TSE method through several steps: **(1) Key Node Selection**, in which we identify the most influential nodes and generate new nodes based on the crucial information they contain; **(2) Connection and Expansion**, in which we systematically connect selected key nodes and expands them into new branches to explore new reasoning directions; and **(3) Collaborative Reasoning**, in which we address deficiencies in the model's ability to synthesize and integrate diverse reasoning paths in different directions.

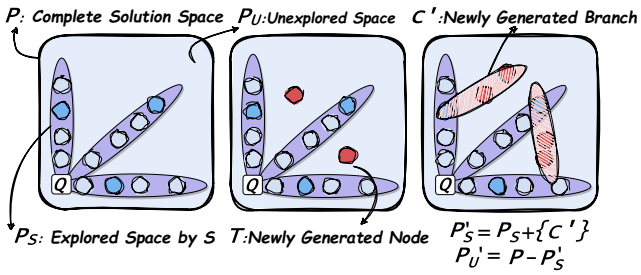

Figure 2: Solution space exploration by TSE. By generating new branches of solutions, the explored space of solution expands.

3.1 Problem Formulation

Given a specific reasoning task $\mathcal{Q}$, a large language model (LLM) $\mathcal{L}$ is expected to generate a thought structure $\mathcal{S}$ composed of directed links connecting sentence-level thought nodes. The set of all possible thought nodes, e.g., sentences or reasoning steps, is denoted as $\mathcal{T}$, where T_{ij} denotes the j-th thought point in the i-th chain. Then the thought point $\mathcal{T}$ and the connections between two nodes compromise the set of vertices $\mathbf{V}$ and the directed edges $\mathbf{E}$ in the thought structure $\mathcal{S}$. Consequently, a thought chain C_i is an ordered sequence of thought nodes:

$$\mathbf{V} = \bigcup_{i=1}^{N} \bigcup_{j=1}^{K_i} \{T_{ij}\}, K_i = |C_i|, \quad (1)$$

$$\mathbf{E} = \bigcup_{i=1}^{N} \bigcup_{j=1}^{K_i-1} \{(T_{ij}, T_{i(j+1)})\}, \quad (2)$$

$$\mathcal{S} = (\mathbf{V}, \mathbf{E}), \quad C_i = \langle T_{i1}, T_{i2}, \ldots, T_{iK_i} \rangle \quad (3)$$

For a specific task $\mathcal{Q}$, the complete reasoning solution space $\mathcal{P}$ encompasses all possible valid reasoning paths (thought chains). As shown in Figure 2, the space that has been explored by the

generated thought structure $\mathcal{S}$ and the unexplored implicit thought space are denoted as $\mathcal{P}_{\mathcal{S}}$ and $\mathcal{P}_{\mathcal{U}}$, where $\mathcal{P}_{\mathcal{S}} \cup \mathcal{P}_{\mathcal{U}} = \mathcal{P}$. Our objective is to generate new branches C' and expand $\mathcal{S}$ to cover as many as possible $\mathcal{P}_{\mathcal{U}}$ to increase the likelihood of finding viable solutions, and the optimization goal is defined as Eq. 4, where J is the reasoning performance metric and $\mathcal{S}'$ is the expanded thought structure.

$$\max_{\mathcal{S}'} J(\mathcal{S}', \mathcal{Q}) \tag{4}$$

3.2 Key Node Selection

The key nodes refer to those significantly impacting the solution path in the solution space $\mathcal{P}$, i.e., containing the known crucial information required to solve the problem. We aim to select key nodes within $\mathcal{S}$, a structure composed of multiple parallel thought chains with the same length. Then, the thought chains $\mathcal{S}$ can be expanded based on the selected key nodes from two aspects: i) exploring the most promising areas for solution paths, and ii) reducing the error propagation by extra analysis and checking on the key nodes, which are often the source of the possible errors. To support the different availability of the backbone models, we propose two methods, gradient-based and self-prompting selection to identify the key nodes for those with accessible gradients and black-box ones, respectively.

3.2.1 Gradient-based Selection

The Gradient-based selection is applicable when the internal structure and the gradient information of model $\mathcal{L}$ is accessible. The representation of each thought node T_{ij} in the generated thought chains is obtained by the model $\mathcal{L}$ as:

$$\mathcal{L} : T_{ij} \rightarrow \mathbf{v}_{ij} \in \mathbb{R}^d \tag{5}$$

Then, we aim to analyze the gradient importance of nodes $\mathbf{T}_{ij}$ relative to the conclusion node T_{iK_i} within a chain. The representation of the conclusion node v_{iK_i} is mapped to the output space as the model's prediction at the end of the chain:

$$\hat{y}_i = f(\mathbf{v}_{iK_i}) \tag{6}$$

where the function f represents a mapping from the representation space of the conclusion node $\mathbf{v}_{iK_i}$ to the output space, and the prediction $\hat{y}_i$ is usually a textual conclusion or decision to task $\mathcal{Q}$.

The self-information loss L_i is a common practice to evaluate the model's confidence in its predictions (Wang and Feng, 2021), where higher confidence corresponds to lower loss values. Thus, we calculate the partial derivative of the loss $\mathbf{g}_{ij}$ with respect to each node's representation $\mathbf{v}_{ij}$ and the Euclidean norm of its gradient G_{ij} to measure the importance of the nodes. Then, a normalization is applied to determine the relative importance I_{ij} of each node for a consistent and comparative analysis of node importance across different chains within the whole thought structure:

$$L_i = -\log P(\hat{y}_i \mid \mathbf{v}_{iK_i}), \tag{7}$$

$$\mathbf{g}_{ij} = \frac{\partial L_i}{\partial \mathbf{v}_{ij}}, \quad G_{ij} = \|\mathbf{g}_{ij}\|_2, \tag{8}$$

$$I_{ij} = \frac{G_{ij}}{\sum_{k=1}^{K_i} G_{ik}} \tag{9}$$

In this way, we can identify the key nodes that have the highest impact on the model's predictions. Finally, a set of key nodes $\mathcal{T}_{key} = \{T_{i_{key}}\}_{i=1}^{N}$ is obtained with the highest relative gradient.

3.2.2 Self-prompting Selection

When the gradient is not accessible, e.g., a black-box model, the self-prompt selection is an alternative for key node identification. Under this setting, we leverage LLM's natual language understanding capabilities to analyze and prioritize key nodes in the thought chains. Although the inner workings of the model are opaque, we infer critical areas within the network's structure by constructing specific prompts based on semantic and logical relationships. Specifically, model $\mathcal{L}$ ranks the importance of nodes within chain C_i for the key node as:

$$\mathcal{T}_{key} = \arg \max_{T_{ij} \in C_i} \text{Rank } \mathcal{L}(T_{ij} \mid \mathcal{Q}) \tag{10}$$

3.3 Connection and Expansion

After obtaining selected key nodes, the next step is to adopt them as conditional information to generate new thought nodes. Then we generate the new nodes based on the key nodes by integrating them into the thought structure and expanding it to form new branches. For each new node, we select two key nodes from $\mathcal{T}_{key}$, denoted as $T_{i_{key}}$ and $T_{l_{key}}$. With each pair of selected key nodes, the model generates a new thought node T_{il}^1 as

$$T_{il}^1 = \mathcal{L}(T_{i_{key}}, T_{l_{key}}), i, l \in [1, N], i \neq l \tag{11}$$

For the new node, to decide which key node to follow, we need to select $T_{i_{key}}$ or $T_{l_{key}}$ to serve as the connection point T_{il}^1 for extending the new

thought branch, which is supposed to maintain logical coherence while exploring previously unknown regions of the thought space. Therefore, we select the conditional nodes with higher semantic relevance with new nodes and contribute more to reasoning as connection nodes. Building on the previous stage, we propose two connection node selection methods: the relative gradient method and the semantic relevance method.

3.3.1 Relative Gradient Selection

The relative gradient selection is applied to take the node between $T_{i_{key}}$ and $T_{l_{key}}$ with the larger relative gradient as the connection point by comparing the importance indices $I_{i_{key}}$ and $I_{l_{key}}$:

$$C' = \langle \arg \max_{T_{key} \in \{T_{i_{key}}, T_{l_{key}}\}} (I_{key}), T_{il}^1 \rangle \quad (12)$$

The C' denotes the new branch initiated from the key node with the higher importance index. Starting from T_{il}^1, the model $\mathcal{L}$ continues reasoning guided by the information from the new nodes, and since the new nodes introduce fresh perspectives of existing information, the new branch is likely to go and explore previously unconsidered directions with step-by-step reasoning. The process of extending the branch C' continues until a specified depth is reached. The language model $\mathcal{L}$ recursively generates subsequent nodes, using the strongest connection node as the context for generating new nodes:

$$C' \rightarrow \mathcal{L}(C') = \langle C', T_{il}^2, \ldots, T_{il}^K \rangle, \quad (13)$$

$$K = \begin{cases} K_i & \text{if } I_{i_{key}}, > I_{l_{key}} \\ K_l & \text{otherwise.} \end{cases} \quad (14)$$

3.3.2 Semantic Relevance Selection

Semantic relevance selection is applicable after both key node selection methods. Here, the connection node is selected by the model $\mathcal{L}$ based on semantic relationships between nodes. Then the new branch C' continues to extend recursively until a specified depth is reached, with $\mathcal{L}$ generating subsequent nodes based on the strongest semantic relationship:

$$C' \rightarrow \mathcal{L}(C') = \left\langle T_{\text{selected}}, T_{il}^1, T_{il}^2, \ldots, T_{il}^K \right\rangle \quad (15)$$

where $T_{selected}$ is the key node chosen by $\mathcal{L}$, and K denotes the ending depth of the thought chain.

3.4 Collaborative Reasoning

Give the task $\mathcal{Q}$ and its unseen complete solution space $\mathcal{P}$, the model needs to generate new thought branches based on the original thought structure $\mathcal{S}$. During this process, each new branch C' expands the explored thought space $\mathcal{P}_{\mathcal{S}} \in \mathcal{P}$ by mining potential solutions based on the established structure:

$$\mathcal{P}'_{\mathcal{S}} \leftarrow \mathcal{P}_{\mathcal{S}} \cup \{C'\}, \quad \mathcal{P}'_{\mathcal{U}} \leftarrow \mathcal{P} - \mathcal{P}'_{\mathcal{S}} \quad (16)$$

Thus the refined structure $\mathcal{S}'$ with more solutions compared with $\mathcal{S}$ achieves a larger explored thought space $|\mathcal{P}'_{\mathcal{S}}| \geq |\mathcal{P}_{\mathcal{S}}|$. Based on $\mathcal{S}'$, we collaborate between original and new thought paths to obtain a unified reasoning conclusion.

3.4.1 Collaborative Weighted Summation

This strategy applies for gradient-available LLMs. We encompass all thought chains in refined thought structure $\mathcal{S}'$. First, we use gradient information to recalculate and select key nodes of each chain. For each key node $T_{i_{key}}$, we calculate a weight $w_{i_{key}}$ based on its impact on the solution:

$$w_{i_{key}} = \frac{\exp(-L_{i_{key}})}{\sum_{k=1}^{K'_i} \exp(-L_{ik})} \quad (17)$$

where $L_{i_{key}}$ represents the self-information loss at node $T_{i_{key}}$, which reflects the model's confidence and potential errors in its prediction at that node. The contribution of each key node, denoted as $v_{i_{key}}$, is the value vector that quantifies the influence of the node on the overall reasoning process. This vector can include various factors such as the relevance of the node's content to the task or the accuracy of its inference. Then for reasoning task $\mathcal{Q}$, we compute the collaborative reasoning score by summing the contributions of all key nodes across all chains:

$$C(\mathcal{Q}) = \sum_{i=1}^{N} \sum_{j=1}^{K_i} w_{i_{key}} \cdot v_{ij} \quad (18)$$

This process integrates the weighted contributions for a reasoning score that involves the individual node's direct impact and its significance within the context of the entire chain. The final decision D for the task $\mathcal{Q}$ is selected based on the highest collaborative reasoning score:

$$D = \arg \max_{q \in \mathcal{Q}} C(q) \quad (19)$$

3.4.2 LM-as-a-Judge

This strategy is suitable for reasoning tasks that require detailed interpretation and judgment, applicable to both gradient-available models and black-box models. In this strategy, the model $\mathcal{L}$ acts as a judge to score each thought chain's output based on its assessment of reasoning coherence, prediction confidence, or relevance to the task. Based on the evaluation, the model employs a voting mechanism to determine the final output:

$$\text{Score}_i = \text{Evaluate}_{\mathcal{L}}(C_q) \tag{20}$$

$$D = \arg\max_{q \in \mathcal{Q}} C(q) \tag{21}$$

In this way, we implement distinct collaboration strategies that enhance the model's reasoning capabilities by leveraging targeted evaluations and adaptive integration.

4 Experiments

We conduct experiments on three reasoning tasks that require mathematical, hierarchical, and comprehensive reasoning abilities to evaluate the effectiveness of our TSE method.

4.1 Experimental Settings

Supplemental Details. We conduct the experiments utilizing the `GPT-4o-mini` (OpenAI, 2024) and `Llama-3.1-8B-Instruct` (Hugging Face, 2024). Unless otherwise specified, we generate five parallel thought chains with a depth of 5 for each question and use this structure as the basis for new node generation. The temperature for all models is set to the default value of 0.7, with a maximum token limit of 50. All tasks are performed on an NVIDIA 4090 GPU.

Baselines. We apply our method to the simplest multi-chain thought structure and compare the results with several baseline methods, including CoT, Vanilla CoT-SC, ToT, and RATT. This comparison aims to illustrate how our method enhances the thought structure compared to existing approaches. Detailed information is provided in Appendix A.1.

Task Description. We evaluate TSE and the baseline methods on three reasoning datasets with specific tasks: (1) *Game of 24*, a mathematical challenge whose objective is to use the four basic arithmetic operations to make four given numbers equal 24. The task requires the language models to combine multiple operations to achieve a target outcome, which can evaluate arithmetic reasoning and logical problem-solving capabilities. (2) *Mini Crosswords*, a game of 5×5 mini crosswords and each input includes the 5 horizontal and 5 vertical clues. To solve this task, the model requires deeper exploration and strategic integration of linguistic clues, allowing us to understand how effectively the evaluated model can expand traditional solution paths and uncover new insights within a complex search space; (3) *Creative Writing*, a task to construct a coherent passage with four paragraphs, each ending with one of four given sentences.This task compels LLMs to generate imaginative text withlogically sound, and contextually rich.

4.2 Overall Performance

The overall performance is reported in Table 1. We can see our TSE consistently outperforms compared baseline methods across different tasks on most of the metrics. The effectiveness of TSE is attributed to expanding thought structures and exploring solution space to contribute to different aspects of reasoning. Then, we analyze each task separately as follows.

Game of 24. The evaluation of Game of 24 is shown in Table 1, where the TSE-refined CoT-SC method significantly outperforms other thought structures, including vanilla CoT-SC, with an improvement of 58.56%, ToT by 40.50%, and RATT by 79.04%. Additionally, our enhancement of the basic CoT-SC structure achieves accuracy on GPT-4o-mini that matches the performance of the more complex ToT ($b = 5$) implemented on GPT-4 (Yao et al., 2024). These results highlight the substantial improvements in reasoning capabilities brought about by our method.

Mini Crossword. In our task setup, we evaluate performance based on two metrics: the proportion of correct letters (out of 25 per game) and the proportion of successfully solved games. As shown in Table 1, the CoT-SC with TSE achieves an impressive accuracy rate of 82.4%, significantly outperforming vanilla CoT-SC (by 30.2%), surpassing ToT by 6.9%. However, for the Game metric, RATT performs slightly better than our method, as it can leverage external knowledge through RAG. Specifically, vanilla CoT-SC suffers from limited exploration of thought paths, which reduces its ability to consistently generate accurate answers. By refining CoT-SC with our method, we enhance its depth and diversity in exploring possible solutions, leading to more accurate and coherent outputs. These results further demonstrate the substan-

Task	Game of 24	Mini Crossword		Creative Writing				
	Success Rate (%)	Success by Letter (%)	Success by Game (%)	Soundness	Innovation	Coherence	Expression	Overall
CoT	13.3	38.4	1.8	5.26	5.24	5.03	4.95	5.12
CoT-SC	46.7	52.2	7.3	5.41	5.17	5.17	5.16	5.23
ToT	52.7	75.5	18.7	5.90	5.36	5.40	5.43	5.52
RATT	41.3	79.2	**25.5**	6.02	6.22	5.69	**5.67**	5.90
TSE	**74.0**	**82.4**	24.1	**6.19**	**6.55**	**5.74**	5.64	**6.03**

Table 1: Overall performance on three reasoning tasks based on GPT-4o-mini. **Bold** and underline indicate the best and the second-best results.

tial improvements our method brings to enhance language models' reasoning capabilities.

Creative Writing. As shown in Table 1, the evaluation criteria included *Soundness*, *Innovation*, *Content Coherence of Reasoning*, and *Clarity of Expression*, applying consistently across all evaluations. We rate each metric on a scale from 0 to 10 in 0.5-point increments, and calculate the overall performance score as the average of these four dimensions. For detailed figure, refer to Appendix A.4.

4.3 Ablation Study

4.3.1 Key Node Selection

To validate the importance of key node selection, we compare (1) the original CoT-SC, (2) the random selection of one node per chain for generation, and (3) our key node selection method. As shown in Table 3, our method significantly improves on *Game of 24* compared to original and random approaches. In *Creative Writing*, particularly for innovation, our method also demonstrates notable enhancements, proving its effectiveness in selecting nodes for generation. The superior performance of our key node selection method is primarily due to its ability to identify nodes containing crucial information for problem-solving. The model only utilizes the most influential information and thus avoids redundancy. In this way, our method enhances the model's efficiency and creativity.

Game of 24		Creative Writing	
Method	**Success Rate (%)**	**Overall**	**Coherence**
Key Node Selection			
Original	5.33	4.68	4.85
Random	3.95	4.54	4.51
TSE	17.11	5.09	5.05
Connection Node Selection			
Random	9.52	4.82	4.85
Layer-based	6.12	4.84	4.88
TSE	16.32	4.91	4.94
Collaborative Reasoning			
Majority-Vote	46.7	5.81	5.87
Random Sampling	27.3	5.90	5.89
New Chains Only	64.0	5.95	5.87
TSE	74.0	6.03	5.90

Table 2: Impact of node selection methods.

4.3.2 Connection Node Selection

We then test the necessity of connection node selection by comparing (1) randomly selecting key nodes to connect, (2) selecting the key nodes based on lower inference layers, and (3) our method. As shown in Table 3, our method significantly superior random selection and layer-based selection on overall reasoning performance, particularly in coherence. This is because random selection lacks specificity and cannot guarantee that the chosen nodes will effectively support the content of new nodes. Although layer-based selection considers the structural hierarchy of information, it does not necessarily reflect the actual importance or applicability of the information. Our method analyzes relationships and reasoning paths between nodes, thus maintaining and strengthening the information flow's coherence and depth.

4.3.3 Collaborative Reasoning

In this experiment, we continue to test the importance of collaboration method in TSE. We compare four collaboration methods: (1) aggregating answers through majority voting, (2) randomly sampling partial thought chains, (3) using only the outcomes from newly generated chains, and (4) implementing the TSE collaborative method. As shown in Table 3, TSE consistently outperforms the other methods in both tasks. Meanwhile, using only new chains also achieves a high success rate in *Game of 24*, demonstrating their potential to offer innovative solutions. However, since these new chains lack integration with original chain information, their performance is still weaker than that of TSE's comprehensive collaboration methods. In *Creative Writing*, using only new chains proved better than majority voting. This indicates that while new chains can provide innovative content, they may fall short in coherence and depth without sufficient integration with the original chains.

4.4 Strategy Analysis

4.4.1 Impact of Key Node Selection Methods

In this experiment, we compare the impact of our two key node selection methods on model reasoning performance. Here we contrast the gradient-based and self-prompting selection on Llama's CoT-SC structures. As shown in Figure 3, both methods perform similarly in task *Game of 24*, while in *Creative Writing*, the self-prompting method shows superior results. This suggests that both methods are capable of selecting appropriate key nodes. Meanwhile, the self-prompting methods guide the model to review known information as well as consider how to connect different thought fragments. As both methods select the key node of the chain, self-promoting explores blind spots that the model might overlook in the usual autoregressive generation process. By delving deeper into these spaces, the model generates more reliable information from the selected nodes and provides more creative content when generating new nodes. Thus, for tasks requiring high creativity and diverse thinking, a potential strategy is to prompt the model to break free from conventional patterns and generate innovative insights.

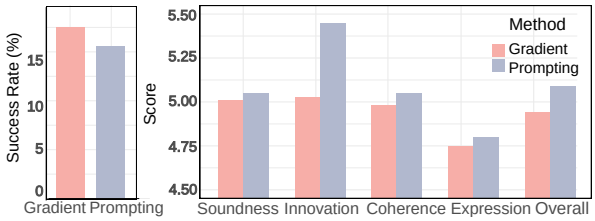

Figure 3: Performance of key node selection methods.

4.4.2 Comparison of Connection Node Selection Methods

In this experiment, we compare the impact of two different methods for selecting connection nodes for newly generated nodes on reasoning performance. We evaluate the gradient comparison and semantic relevance methods on Llama-3.1-8B. As shown in Table 3, selecting nodes based on semantic relevance leads to better performance on *Game of 24* reasoning and content coherence of reasoning in *Creative Writing*. The difference possibly arises from two main reasons: (1) Nodes with larger relative gradients do not necessarily have a stronger influence on the content when generating new nodes. (2) Compared to gradient-based selection, the semantic relevance method more effectively captures the actual content and meaning relationships between nodes, leading to more suitable connection points for new nodes. This demonstrates that considering semantic relevance when selecting connection points provides more effective support for reasoning performance.

	Game of 24	Creative Writing	
Method	**Success Rate (%)**	**Overall**	**Coherence**
Connection Node Selection Methods			
Gradient	13.61	4.88	4.86
Relevance	16.32	4.91	4.94
Collaborative Reasoning Methods			
CWS	12.93	4.97	5.01
LM-as-a-Judge	15.65	4.94	4.95

Table 3: Comparison of connection node selection and collaborative reasoning strategy.

4.4.3 Comparison of Collaborative Reasoning Methods

In this experiment, we compare the performance of CWS and LM-as-a-Judge. As shown in Table 3, in task *Game of 24*, which requires dense logic and has well-defined objectives, the LM-as-a-Judge method accurately and quickly identifies the correct answers by directly evaluating each thought chain and selecting the optimal solution. This method relies on the model's ability to assess individual outputs, allowing for rapid selection of the best solution—especially useful in tasks that require precise calculations and quick responses. On the other hand, in task *Creative Writing*, which requires innovation and diversity, CWS evaluates and weights the contributions of each thought chain. The final output represents a blend of multiple perspectives and enhances the richness and depth of the narrative significantly. We further analyze the thought chain efficacy and provide a case study to illustrate the effectiveness of TSE generation and collaboration, referring to Appendix A.3.

5 Conclusion

In this study, we introduce TSE, a novel approach to enhance the reasoning structures of LLMs. TSE generates new thought branches based on existing thought paths to explore previously overlooked solutions. The generated new reasoning nodes and chains are incorporated into thought structures to explore diverse reasoning directions in terms of a reasoning task. Our experiments across multiple reasoning datasets demonstrate the effectiveness of the TSE. The detailed analysis reveals the utilization of each component in TSE during the integration of diverse thought processes.

Limitations

We develop a framework with multiple strategies to enhance thought structures for LLMs, where the expansion of thought structures through self-generated nodes might lead to over-fitting to existing patterns as the process works without integrating external knowledge. This may limit the TSE method's ability to explore more diverse thought patterns. Besides, the experiments are conducted on only two language models, which may not provide a comprehensive view of the TSE's generalization capability across LLMs with varying sizes and pre-training processes. Moreover, the TSE is mainly evaluated on specific tasks and these tasks cannot fully reflect the complexities of real-world scenarios where reasoning tasks can be variable and with more complex and external solution spaces. We left these potential explorations as future work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17682–17690.

Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2023. A survey of chain of thought reasoning: Advances, frontiers and future. *arXiv preprint arXiv:2309.15402*.

Xidong Feng, Ziyu Wan, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. 2023. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*.

Zitian Gao, Boye Niu, Xuzheng He, Haotian Xu, Hongzhang Liu, Aiwei Liu, Xuming Hu, and Lijie Wen. 2024. Interpretable contrastive monte carlo tree search reasoning. *arXiv preprint arXiv:2410.01707*.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*.

Guiyang Hou, Wenqi Zhang, Yongliang Shen, Linjuan Wu, and Weiming Lu. 2024. Timetom: Temporal space is the key to unlocking the door of large language models' theory-of-mind. *arXiv preprint arXiv:2407.01455*.

Jie Huang and Kevin Chen-Chuan Chang. 2022. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*.

Hugging Face. 2024. Llama-3.1-8B-Instruct.

Ziqi Jin and Wei Lu. 2024. Self-harmonized chain of thought. *arXiv preprint arXiv:2409.04057*.

Jaehun Jung, Lianhui Qin, Sean Welleck, Faeze Brahman, Chandra Bhagavatula, Ronan Le Bras, and Yejin Choi. 2022. Maieutic prompting: Logically consistent reasoning with recursive explanations. *arXiv preprint arXiv:2205.11822*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. 2024. Chain of thought empowers transformers to solve inherently serial problems. *arXiv preprint arXiv:2402.12875*.

Xun Liang, Shichao Song, Zifan Zheng, Hanyu Wang, Qingchen Yu, Xunkai Li, Rong-Hua Li, Feiyu Xiong, and Zhiyu Li. 2024. Internal consistency and self-feedback in large language models: A survey. *arXiv preprint arXiv:2407.14507*.

Tongxuan Liu, Wenjiang Xu, Weizhe Huang, Xingyu Wang, Jiaxing Wang, Hailong Yang, and Jing Li. 2024. Logic-of-thought: Injecting logic into contexts for full reasoning in large language models. *arXiv preprint arXiv:2409.17539*.

Jieyi Long. 2023. Large language model guided tree-of-thought. *arXiv preprint arXiv:2305.08291*.

Shentong Mo and Miao Xin. 2024. Tree of uncertain thoughts reasoning for large language models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12742–12746. IEEE.

OpenAI. 2023. Gpt-4. `https://openai.com/index/gpt-4-research/`.

OpenAI. 2024. GPT-4o-mini: Advancing Cost-Efficient Intelligence. Accessed: 2024-10-05.

David Patterson, Joseph Gonzalez, Urs Hölzle, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R So, Maud Texier, and Jeff Dean. 2022. The carbon footprint of machine learning training will plateau, then shrink. *Computer*, 55(7):18–28.

Noah Shinn, Beck Labash, and Ashwin Gopinath. 2023. Reflexion: an autonomous agent with dynamic memory and self-reflection. *arXiv preprint arXiv:2303.11366*, 2(5):9.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2024. To cot or not to cot? chain-of-thought helps mainly on math and symbolic reasoning. *arXiv preprint arXiv:2409.12183*.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. 2024. Chain of thoughtlessness: An analysis of cot in planning. *arXiv preprint arXiv:2405.04776*.

Weikuan Wang and Ao Feng. 2021. Self-information loss compensation learning for machine-generated text detection. *Mathematical Problems in Engineering*, 2021(1):6669468.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Xuezhi Wang and Denny Zhou. 2024. Chain-of-thought reasoning without prompting. *arXiv preprint arXiv:2402.10200*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yu Xia, Rui Wang, Xu Liu, Mingyan Li, Tong Yu, Xiang Chen, Julian McAuley, and Shuai Li. 2024. Beyond chain-of-thought: A survey of chain-of-x paradigms for llms. *arXiv preprint arXiv:2404.15676*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Jinghan Zhang, Xiting Wang, Weijieying Ren, Lu Jiang, Dongjie Wang, and Kunpeng Liu. 2024a. Ratt: Athought structure for coherent and correct llmreasoning. *arXiv preprint arXiv:2406.02746*.

Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. 2024b. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *arXiv preprint arXiv:2406.09136*.

Yifan Zhang, Yang Yuan, and Andrew Chi-Chih Yao. 2024c. On the diagram of thought. *arXiv preprint arXiv:2409.10038*.

Xinyu Zhu, Junjie Wang, Lin Zhang, Yuxiang Zhang, Yongfeng Huang, Ruyi Gan, Jiaxing Zhang, and Yujiu Yang. 2022. Solving math word problems via cooperative reasoning induced language models. *arXiv preprint arXiv:2210.16257*.

A Experimental Settings

A.1 Baselines

We apply our method to the simplest multi-chain thought structure and compare the results with several baseline methods, including CoT, Vanilla CoT-SC, ToT, and RATT. This comparison aims to illustrate how our method enhances the thought structure compared to existing approaches. For a consistent evaluation, we standardize the depth to five across all methods. Specifically, we set the number of chains to $k = 5$ and generate $b = 5$ candidates at each step for CoT-SC and ToT, respectively. For RATT, the model generates and integrates five candidate results at each decision point and uses `Wikipedia`[1] as the external resource.

A.2 Task Description

We evaluate TSE and the baseline methods on three reasoning datasets with specific tasks: (1) *Game of 24*, a mathematical challenge whose objective is to use the four basic arithmetic operations to make four given numbers equal 24. The task requires the language models to combine multiple operations to achieve a target outcome, which can evaluate arithmetic reasoning and logical problem-solving capabilities. (2) *Mini Crosswords*, a game of 5×5 mini crosswords and each input includes the 5 horizontal and 5 vertical clues. The expected output is a completed 55 crossword board containing 25 letters. To solve this task, the model requires deeper exploration and strategic integration of linguistic clues, allowing us to understand how effectively the evaluated model can expand traditional solution paths and uncover new insights within a complex search space; (3) *Creative Writing*, a task to construct a coherent passage with four paragraphs, each ending with one of four given sentences.This task compels LLMs to generate imaginative text withlogically sound, and contextually rich. To ensure a rigorous and comprehensive assessment, we conduct half of the evaluations using GPT-4 (OpenAI, 2023), while the other half are performed by a panel of expert human annotators. In this task, we select 100 open-ended questions from several prompts listed on *Reedsy. com*[2] as our input.

[1] https://en.wikipedia.org

[2] https://blog.reedsy.com/creative-writing-prompts/#:~:text=When%20the%20idea%20to%20start%20a%20weekly%20newsletter

A.3 Thought Chain Efficacy

In this experiment, we conduct a detailed comparison between the success rates of original thought chains and new chains generated by our method in the *Game of 24* task. Experimental results indicate that for each problem, the success rate of having at least one correct answer in the original CoT-SC is approximately 50.7%, while the new chains generated by our method achieve a success rate of 64.0%. Moreover, despite improving accuracy, the overlap of problems successfully solved by both the old and new chains is only 35.3%. This low level of overlap suggests that our method effectively explores new areas that the original thought structures do not address and brings a significant increase in the overall success rate.

A.4 Creative Writing

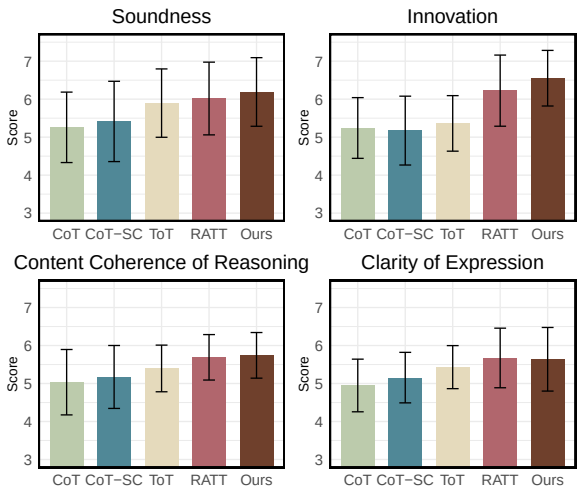

Figure 4: Performance of different methods on GPT-4o-mini for task Creative Writing.

B Detailed Information of Case Study

B.1 Case Analysis

As our approach significantly enhances the reasoning capabilities of thought structures while maintaining minimal computational cost, it effectively addresses gaps within existing structures. Figure 5 provides a specific example of this enhancement by comparing the original thought chains with the new chains generated by our method. In the provided case study, we see the reasoning prompt *"Explain why it is important for children to learn mathematics"*. The original thought chains (Thought Chain 1 and Thought Chain 2) are linear and somewhat limited in scope, focusing and repeating on founda-

tional aspects like *"Critical Thinking", "Academic Success"* and *"Career Opportunities"*. However, the new chain generated by our method explores additional dimensions, such as ''*Enhancement of Cognitive and Memory Skills"* and *"Preparation for the Digital Age"*. These new perspectives enrich the argument and provide a more comprehensive understanding of the importance of mathematics. For the detailed information of this case, refer to Appendix B.

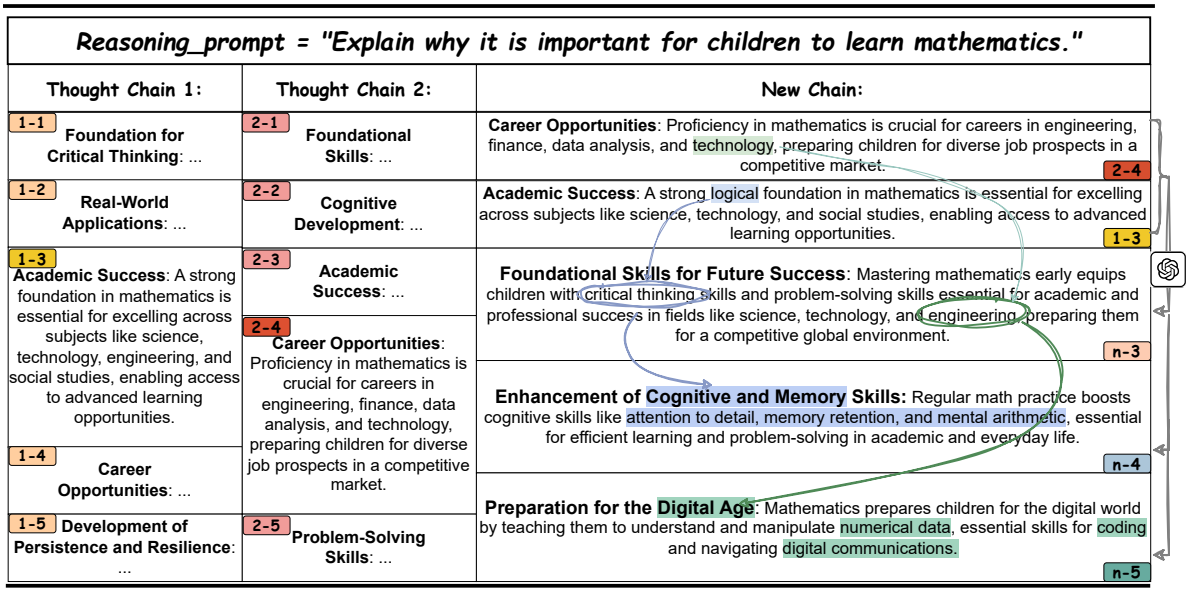

Figure 5: Case study of comparison of original and new chains.

Thought Chain 1	Thought Chain 2	New Chain
Foundation for Critical Thinking: Mathematics fosters critical and logical thinking, enhancing problem-solving and analytical skills necessary for daily life.	**Foundational Skills:** Mathematics equips children with essential daily skills like counting, measuring, and problem-solving, crucial for tasks such as budgeting, shopping, and time management.	**Career Opportunities:** Proficiency in mathematics is crucial for careers in engineering, finance, data analysis, and technology, preparing children for diverse job prospects in a competitive market.
Real-World Applications: Mathematics is crucial for managing daily activities like finances, cooking, and scheduling, empowering children with the confidence to handle them effectively.	**Cognitive Development:** Math enhances logical reasoning, critical thinking, and problem-solving abilities, benefiting other subjects and real-life situations.	**Academic Success:** A strong foundation in mathematics is essential for excelling across subjects like science, technology, engineering, and social studies, enabling access to advanced learning opportunities.
Academic Success: A strong mathematics foundation is essential for excelling in subjects like science, technology, engineering, and social studies, paving the way for advanced educational opportunities.	**Academic Success:** Proficiency in mathematics underpins success in STEM fields, crucial for today's job market, and opens doors to advanced educational opportunities.	**Foundational Skills for Future Success:** Mastering mathematics early equips children with critical thinking and problem-solving skills essential for academic and professional success in fields like science, technology, and engineering, preparing them for a competitive global environment.
Career Opportunities: Proficiency in mathematics is key for careers in engineering, finance, technology, medicine, and data science, equipping children for diverse job prospects in a competitive market.	**Career Opportunities:** Proficiency in mathematics is crucial for careers in engineering, finance, data analysis, and technology, preparing children for a broad range of opportunities in a competitive job market.	**Enhancement of Cognitive and Memory Skills:** Regular math practice boosts cognitive skills like attention to detail, memory retention, and mental arithmetic, essential for efficient learning and problem-solving in academic and everyday life.
Development of Persistence and Resilience: Mathematics teaches children systematic problem-solving, a skill invaluable in academics and beyond.	**Problem-Solving Skills:** Mathematics teaches children systematic problem-solving, a skill invaluable in academics and beyond.	**Preparation for the Digital Age:** Mathematics prepares children for the digital world by teaching them to understand and manipulate numerical data, essential skills for coding and navigating technological advancements.

Table 4: Reasoning Prompt: Explain why it is important for children to learn mathematics.