CORE-EVO: Evolutionary Search for Enhancing Commonsense Reasoning in Large Language Models

Anonymous ACL submission

Abstract

The test-time scaling techniques demonstrate a potential direction to improve the reasoning abilities of LLMs by searching the reasoning space with a score function. Although the test-time scaling methods have been widely studied for math-reasoning tasks, the inference scaling capabilities of LLMs for commonsense reasoning remain largely underexplored. In this work, we examine the scalability of inference scaling techniques for commonsense reasoning by using a pretrained entailment verifier model as the score function. We also propose a new inference scaling method, called CORE-EVO, which integrates the evolutionary search algorithm with LLMs. The CORE-EVO is capable of tackling the shortcomings of the best-of-N and self-consistency, searching for the reasoning path in the local optima in reasoning spaces, by performing evolutionary operations and population refinement based on the entailment verification score. The experimental results on CommonsenseQA, PIQA, and SocialIQA benchmarks show that our method is able to scale inference compute more effectively than other test-time scaling techniques at high inference scales. Notably, our method outperforms the best-of-N and self-consistency by a significant gap, about 4% and 3% respectively, in terms of average performance using the Llama3.1-8B language model.

1 Introduction

Recent advances in large language models (LLMs) have significantly enhanced performance on commonsense reasoning tasks through increased model size, extensive pretraining (Grattafiori et al., 2024; Yang et al., 2024a), fine-tuning on high-quality instruction data (PENG et al., 2025; Ho et al., 2023; Kang et al., 2023), and prompt-tuning techniques (Wei et al., 2022b; Madaan et al., 2023; Zhang et al., 2023). Orthogonal to these approaches, an emerging research direction is to leverage the test-time scaling compute of LLMs (Wang et al., 2023; Snell et al., 2025) to enhance reasoning capabilities without additional training. This direction builds on the premise that LLMs, having been pretrained on massive web-scale data, may already possess the internal knowledge required for complex common-sense inference. However, the inference scaling capabilities of LLMs for commonsense reasoning remain largely underexplored.

Current inference scaling techniques, such as best-of-N sampling (Gui et al., 2024) and self-consistency (Wang et al., 2023), aim to improve the reasoning capabilities of pretrained LLMs by identifying consistent and high-quality reasoning paths. While effective, these methods share a key limitation: their search strategies often converge to local optima, as the sampled candidates tend to be similar to one another. This constraint may hinder the ability of pretrained LLMs to scale effectively in discovering truly optimal reasoning paths.

To address these limitations, we propose a novel algorithm, called COmmensense REasoning EVOlution (CORE-EVO), which incorporates evolutionary search (Lee et al., 2025) to move beyond local optima and efficiently explore optimal reasoning paths. The core idea of our method is illustrated in Figure 1. CORE-EVO begins by initializing a population of reasoning paths sampled from an LLM, and iteratively applies mutation or crossover operations to evolve the population over a fixed number of rounds. We leverage the self-refinement capabilities of LLMs to perform mutations, while prompt-based strategies are used to implement crossovers. After each evolutionary step, new reasoning paths are assessed using a pretrained entailment verifier (Sanyal et al., 2024) and incorporated into the population. The population is then refined to retain higher-quality candidates. Finally, the evolved reasoning paths from the last round are aggregated to produce a consistent inference.

In contrast to best-of-N and self-consistency methods, which typically search for reasoning

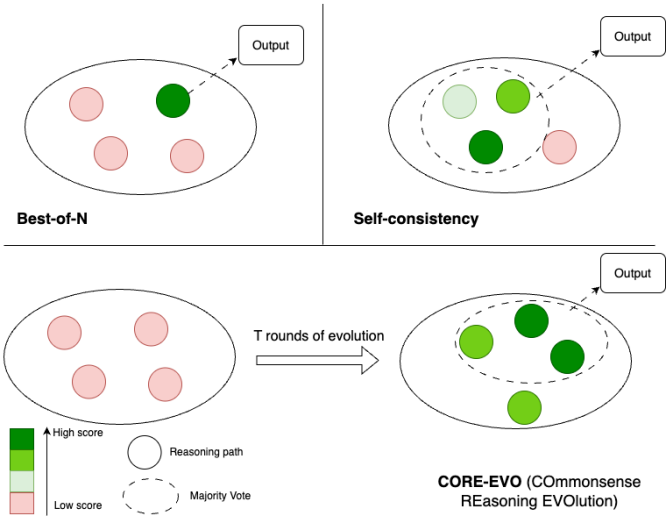


Figure 1: A comparison between the CORE-EVO with other inference scaling techniques, Best-of-N and Self-consistency. Each circle represents a sampled reasoning path. The big oval represents a population of reasoning paths. The dashed oval demonstrates the majority vote procedure on a population.

paths near a local optimum in the reasoning space, CORE-EVO addresses the challenge of discovering solutions beyond local optima by iteratively evolving a population of reasoning paths. This enables more effective exploration of high-scoring regions in the reasoning space. Additionally, our method differs from the evolutionary search approach in (Lee et al., 2025) by tailoring the crossover, mutation, and fitness functions specifically for commonsense reasoning.

We empirically evaluate our method against competitive inference scaling techniques for pretrained LLMs on a range of commonsense reasoning tasks, covering both physical and social domains. Our experiments show that CORE-EVO produces more diverse reasoning paths than conventional methods, facilitating the discovery of more accurate solutions using *the same number of sampled paths*. The results on the CommonsenseQA, PIQA, and SocialIQA benchmarks also demonstrate that our method significantly outperforms both best-of-N and self-consistency across two LLM backbones, the Llama3.1-8B and the Qwen1.5-7B, at different inference scaling levels. These findings highlight the superior capability of CORE-EVO in effectively exploring reasoning spaces for commonsense tasks.

2 Method

We introduce the CORE-EVO, a reasoning framework based on scaling test-time inference to address commonsense reasoning tasks. The CORE-EVO is developed according to the evolutionary algorithm that is inspired by the biological evolutionary process. The main idea is to leverage the power of LLMs to perform evolutionary operations, and then iteratively refine the population of reasoning paths based on their fitness scores. Our framework is a general test-time scaling method that is able to adapt to any pretrained LLMs. The overall framework is illustrated in the Algorithm. 1.

As shown in the Algorithm. 1, the proposed framework makes use of a pretrained LLM to first sample multiple reasoning paths $R = \{r_i\}_{i=1}^{N}$, called initial population, for a given query $\mathcal{Q}$ and compute corresponding fitness scores for sampled reasoning paths. Then, a subsequence of the evolution operations is performed to enhance the initial population. For each evolutionary round, we grow the population by adding a new reasoning trace by performing selection and then operating mutation or crossover. At the end of the evolutionary round, we refine the population by the procedure as follows. Given a list of reasoning paths $R = \{r_i\}_{i=1}^{N+1}$ and their corresponding scores $S = \{s_i\}_{i=1}^{N+1}$, we define a permutation $\pi : \{1, 2, \ldots, N+1\} \rightarrow \{1, 2, \ldots, N+1\}$ such that $s_{\pi(1)} \geq s_{\pi(2)} \geq \ldots \geq s_{\pi(N+1)}$. The top-$N$ elements are then selected as $R_k = \{r_{\pi(i)} \mid i \in \{1, 2, \ldots, N\}\}$. Finally, we aggregate over the final population of reasoning paths to achieve the answer

$$\text{maj-vote}(A) = \arg\max_{\mathcal{A} \in A} \sum_{i=1}^{N} \mathbb{1}(\mathcal{A}_i = \mathcal{A}) \quad (1)$$

Algorithm 1 Evolutionary search for commonsense reasoning

Require: A pretrained LLM $f(\cdot)$, population size N, a score function $g(\cdot)$, evolution rounds T, an input query $\mathcal{Q}$, and prompts $\mathcal{T}_{init}, \mathcal{T}_{mutation}, \mathcal{T}_{crossover}$.

Initialize population: $R_0 \leftarrow \{r_i\}_{i=1}^{N} \sim f(\mathcal{Q}, \mathcal{T}_{init}, N), S_0 \leftarrow \{g(r_i)\}_{i=0}^{N}$ ▷ Sample N reasoning paths
for $t = 1$ to T do **do**
  $\epsilon \sim \mathcal{U}(0, 1)$
  if $\epsilon \leq 0.3$ **then**
    Selection: $r_1 \sim R_{t-1}$ ▷ Sample a reasoning path to perform mutation
    Mutation: $r_{new} \leftarrow f(\mathcal{Q}, \mathcal{T}_{mutation}, r_1)$ ▷ Mutation based on prompting LLMs
  else
    Selection: $r_1, r_2 \sim R_{t-1}$ ▷ Sample two reasoning paths to perform crossover
    Crossover: $r_{new} \leftarrow f(\mathcal{Q}, \mathcal{T}_{crossover}, r_1, r_2)$ ▷ Crossover based on prompting LLMs
  end if
  Update population: $R_t \leftarrow \{R_{t-1}, r_{new}\}, S_t \leftarrow \{S_{t-1}, g(r_{new})\}$
  Refine population: get top-N reasoning path base on score $R_t \leftarrow$ top-N(R_t) ▷ Refine population based on scores
end for
return the answer based on population R_T, $\mathcal{A} \leftarrow \arg\max_{\mathcal{A} \in A} \sum_{i=1}^{N} \mathbb{1}(\mathcal{A}_i = \mathcal{A})$

2.1 Population Initialization

We prompt a LLM $f(\cdot)$ with the query $\mathcal{Q}$ along with the initialization instruction, consists of a task instruction and a few demonstrated examples, $\mathcal{T}_{init}$. The LLM then generates multiple potential reasoning paths by using sampling techniques, such as either nucleus or top-k sampling methods.

$$R = f(\mathcal{Q}, \mathcal{T}_{init}, N) \tag{2}$$

, where N is the number of individuals in a population. The instruction $\mathcal{T}$ is a manual instruction to guide the language model to generate supporting information to draw a plausible explanation (Liu et al., 2022) for the input query $\mathcal{Q}$. We also demonstrate a few detailed reasoning paths in the instruction to steer the language model following the desired template. Our CoT is slightly different from the standard CoT (Wei et al., 2022b) by defining well-structured reasoning chains for commonsense reasoning tasks. The reasoning chains are described as follows:

- **Step 1:** Retrieve relevant knowledge regarding the query, such as the object or context being asked in the query.
- **Step 2:** Gather retrieved knowledge from previous steps and given options to simulate a realistic situation.
- **Step 3:** Conduct a reasoning based on the simulated situation from Step 2 and draw a conclusion.

Well-structured reasoning chains are critical to impacting the reasoning abilities of LLMs (Yao et al., 2023; Jin et al., 2024). (Liu et al., 2022) shows that the generated information from a pretrained LLM is able to enhance commonsense reasoning capabilities. Therefore, the rationale for the first three steps is to generate relevant knowledge to assist language models in drawing correct and plausible reasoning in the fourth step. We curate and unify the design of the reasoning chain for commonsense reasoning tasks to disentangle the performance gain of prompt design. Furthermore, the well-structured reasoning chains are more verifiable by using a specialized language model that is crucial in our framework to evaluate the fitness score of each reasoning path. We utilize the above reasoning chain structure for all of our experiments.

2.2 Selection Process

The reasoning path selected in the evolutionary round t is sampled from the population of the previous round R_{t-1} with the initial population at round $t = 0$. In round t, given the population is N of reasoning paths $R_{t-1} = \{r_i\}_{i=1}^{N}$ with the corresponding score $S_{t-1} = \{s_i\}_{i=0}^{N}$, there are two selection processes that could be used to sample reasoning paths to perform evolution as follows

- **Tournament selection:** There are k individuals $R^k \subset R_{t-1}$ randomly selected from the population along with their score $S^k \subset S_{t-1}$, and then the fittest individual is chosen from the k sampled individuals using the score values $r = \arg\max_{r \in R^k} S^k$.
- **Rottle-wheel selection:** This method assigns selection probability proportional to fitness $p(i) = \frac{s_i}{\sum_{s_j \in S_{t-1}} s_j}$, where $p(i)$ is the probability of selecting individual i, f_i is its fitness, and N is the population size. Then, a reasoning path r is sampled from R_{t-1} based on the distribution $P = \{p(i)\}_{i=1}^{N}$.

We conduct an ablation study in the experiment section to select the best selection technique for

our framework.

2.3 Evolutionary Operations

There are prior works (Meyerson et al., 2024; Lehman et al., 2023) that demonstrate the capability of LLMs to perform evolutionary operations for a wide range of language generation tasks. Inspired by these previous works, we prompt LLMs with detailed instructions to perform evolutionary operations, which are mutation and crossover operations. The prompts for evolutionary operations can be found in the Appendix. B

Mutation operation. Given a parent reasoning trace r_1, we operate a mutation by first prompting the language model with the parent reasoning trace along with the query $\mathcal{Q}$ and the reflection instruction $\mathcal{T}_{reflect}$ to acquire feedback

$$\mathcal{F} = f(\mathcal{T}_{reflect}, \mathcal{Q}, r_1) \quad (3)$$

. Then, the feedback $\mathcal{F}$ is concatenated with the mutation instruction $\mathcal{T}_{mutaion}$ to refine the parent reasoning trace

$$r_{new} = f(\mathcal{T}_{mutation}, \mathcal{F}, r_1,) \quad (4)$$

. The feedback $\mathcal{F}$ plays a crucial role in the mutation operation to refine the parent reasoning trace for getting a more correct and plausible reasoning path (Madaan et al., 2023).

Crossover operation. Given a pair of parent reasoning paths (r_1, r_2), we prompt the language model with a pair of parents along with the crossover instruction $\mathcal{T}_{crossover}$, which instructs to randomly mix reasoning steps between two parents, to generate a new reasoning path

$$r_{new} = f(\mathcal{Q}, \mathcal{T}_{reflect}, r_1, r_2) \quad (5)$$

2.4 Fitness function

To measure the quality of reasoning paths, we leverage the pretrained entailment verifier described in (Sanyal et al., 2024) as the fitness function. The entailment verifier-based language model is the Flan-T5-xxl language model (Chung et al., 2024) which is fintuned on a large amount of natural language inference tasks. The pretrained entailment verifier is capable of measuring the entailment score $s(p, h)$ between a premise p and a hypothesis h. Given a query $\mathcal{Q}$ and a reasoning path $r = \{r^1, \ldots, r^m\}$ of m reasoning steps, the quality of the reasoning path is computed as follows

$$g(r, \mathcal{Q}) = \sum_{i=1}^{m} s(\mathcal{Q}||r^{<i}, r^i), \text{where } r^0 = \emptyset \quad (6)$$

. The entailment score of the reasoning chain with respect to the query reflects the quality of the reasoning chain to conclude with a plausible and correct answer.

3 Experiments

We design experiments to demonstrate the effectiveness of our proposed framework in enhancing the commonsense reasoning abilities of LLMs on three popular benchmarks. Then, we demonstrate the superior scalability of our methodology by comparing our approach with existing inference scaling techniques. Finally, we perform a comprehensive ablation study to analyze the contribution of each component to the design choice of our method.

3.1 Experimental Setup

Benchmarks and baselines. We evaluate the performance of our framework and baselines on three commonsense reasoning benchmarks: PIQA (Bisk et al., 2019), CommonsenseQA (Talmor et al., 2019), and Social IQA (Sap et al., 2019). We use accuracy on these three benchmarks as the evaluation metric. We compare our method against the following baselines: Chain-of-though (CoT) (Wei et al., 2022b), Best-of-N, Self-Refine (Madaan et al., 2023), and Self-consistency (Wang et al., 2023).

Language models and sampling scheme. We study the effectiveness of our method on two LLMs: Llama3.1-8B (Grattafiori et al., 2024) and Qwen1.5-7B (Bai et al., 2023) models. We also use the Llama 3 family model, consisting of Llama3.2-1B, Llama3.2-3B, and Llama3.1-8B models, to examine the influence of model parameters on our method. To generate diverse reasoning paths, we apply temperature sampling with temp=0.8 truncated at the top-k (k=50) and top-p (p=0.7). The sampling hyperpameters are applied to all aforementioned large language models in all experiments. We conduct an ablation study for searching sampling hyperparameters in the appendix. A

3.2 Overall Performance

To validate the efficacy of our method, we evaluate our method against baseline techniques on three benchmarks: PIQA, CommonsenseQA, and Social IQA. We configure our method with the number of population $N = 10$ and the number of evolutionary rounds $T = 20$ for the experimental results shown in Table. 1. We run all experiments three times and report the average results for all methods.

	Method	PIQA	CommonsenseQA	SocialIQA	Average
Llama3.1-8B	CoT	79.37	77.12	72.38	76.29
	Best-of-N	74.64	80.34	75.23	76.73
	Self-consistency	78.72	79.87	74.76	77.83
	Self-refine (4-iterations)	76.92	78.82	74.21	76.65
	Ours	**84.65**	**81.98**	**76.25**	**80.96**
Qwen1.5-7B	CoT	73.62	73.44	62.53	69.86
	Best-of-N	79.44	75.94	63.66	73.01
	Self-consistency	79.95	79.59	73.59	77.71
	Self-refine (4-iterations)	75.26	74.21	62.94	70.80
	Ours	**80.99**	**81.67**	**74.64**	**79.1**

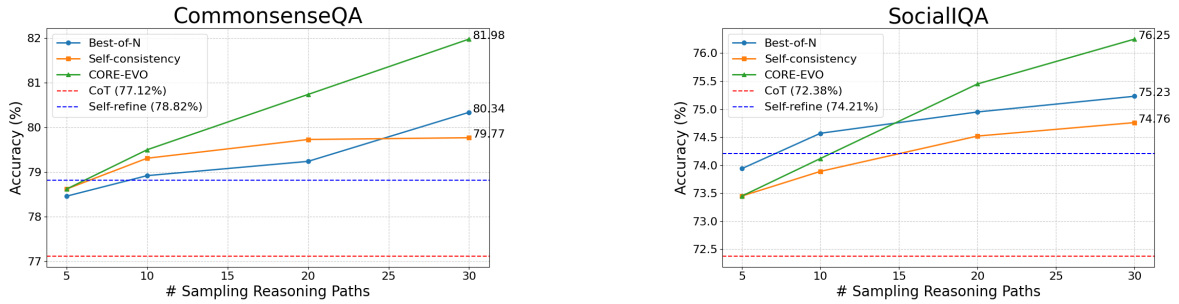
Table 1: The commonsense reasoning accuracy of our method compared to baseline techniques on three benchmarks. The result of CoT method is reported with the greedy decoding, and other method results are reported with the sampling technique described in Section. 3.1. The number of generated reasoning paths of Best-of-N and self-consistency is on par with the number of reasoning paths of our method for a fair comparison. All experiments are run three times, and the average results are reported.

	CommonsenseQA
Question	If there is a place that is hot and arid, what could it be? A) Bland B) Lifeless C) Sandy D) Neutral E) Freezing
Best-of-N	**Step 1:** A hot and arid place could be a desert, which is characterized by its extreme heat and lack of rainfall, resulting in arid conditions. The question asks for a description of a place that is hot and arid, implying a geographical location with specific climate conditions. **Step 2:** For example, the Sahara Desert in North Africa is a hot and arid region, known for its sandy dunes and extreme temperatures. **Step 3:** Deserts are often associated with hot and arid conditions, making them a fitting description for such a place. C
Self-consistency	**Step 1:** The object is a place that is hot and arid, which suggests a region with extreme temperatures and low humidity, often characterized by vast expanses of sand and limited vegetation. The question asks for a possible description of such a place. **Step 2:** For example, a desert like the Sahara is hot and arid, and it can be described as sandy due to its vast dunes of sand. **Step 3:** A place that is hot and arid is often associated with sandy landscapes, which are typical of many deserts. C
CORE-EVO	**Step 1:** A hot and arid place could be a desert, characterized by high temperatures, low humidity, and limited vegetation. The question asks what type of place would have these characteristics. **Step 2:** For example, a desert like the Sahara Desert is known for its hot and arid conditions, with vast expanses of sandy dunes and limited plant life. **Step 3:** Deserts are often associated with extreme heat and aridity, which matches the description provided in the question, and are also often described as lifeless due to the harsh conditions and limited vegetation. B

Table 2: This qualitative example where the CORE-EVO provides a more accurate reasoning path over baselines. The reasoning paths are selected based on the highest score in the population for each method. The issues with the reasoning path are highlighted in red. The correct answer and accurate explanation are highlighted in blue.

The Table. 1 shows the overall performance of our CORE-EVO algorithm compared to other inference scaling techniques on three standard commonsense reasoning benchmarks. We use the aforementioned setting for the CORE-EVO algorithm to generate 30 reasoning paths, 10 initial traces and 20 evolved traces, for each question. We also sample the same number of reasoning paths for the Best-of-N and self-consistency methods to have a fair comparison. Regarding the self-refinement method, we are able to perform a maximum of 4 iterations for refinement due to the computational constraints. The self-refine method demonstrates a modest performance gain with the CoT method across the benchmarks, while both Best-of-N and self-consistency techniques demonstrate substantial performance improvements on average. Our approach combines both the self-fine method and the self-consistency method under the lens of the evolutionary search algorithm. Therefore, it outperforms the best scaling technique baseline, the self-consistency method, by a large margin in average accuracy. In particular, our algorithm increases the accuracy of the reasoning by $\sim 3\%$ compared to the self-consistency method for the Llama3.1-8B model across benchmarks.

The Table. 3 illustrates the diversity of generated reasoning paths of baseline methods and our



(a) (b)

Figure 2: The inference scaling experiment on CommonsenseQA and SocialIQA benchmark for Llama3.1-8B model. At the scale of 5-reasoning paths, the CORE-EVO generates a population of 5 reasoning paths without performing evolution on the population, thereby it resembles self-consistency at that scale. The setting for scale of 10, 20, and 30-reasoning paths of CORE-EVO are $\{N = 5, T = 5\}$, $\{N = 10, T = 10\}$, and $\{N = 10, T = 20\}$, respectively. All experiments are run three times, and the average results are reported.

Method	CommonsenseQA $Var(M)$	SocialIQA $Var(M)$
Best-of-N	0.00305	0.0011
Self-consistency	0.00302	0.0011
CORE-EVO	0.00346	0.0019

Table 3: The experiment studies the diversity of generated reasoning paths $N = 30$. The reported metric $Var(M)$ is the variance of the cosine similarity matrix M between reasoning paths using the pretrained sentence embedding model (Reimers and Gurevych, 2019).

method with a population size of $N = 30$. Our method is capable of generating a diverse set of reasoning paths, thus its variance on the cosine similarity matrix between reasoning paths is higher than the baselines. We also demonstrate qualitative examples in Table. 9 to show the capabilities of searching for a more plausible and accurate reasoning path of our method. We conduct qualitative analysis in Appendix. C.

3.3 Inference Scaling

We conduct the inference scaling experiments to demonstrate the scalability of our algorithm compared to other inference scaling methods on the CommonsenseQA and Social IQA benchmarks for the Llama3.1-8B model. We perform experiments with respect to a varying number of sampled reasoning paths (5, 10, 20, 30) and report the average accuracy for 3 runs. The sampling scheme is described in 3.1 to sample a diverse set of reasoning paths on different scales. Since the CORE-EVO has two parameters, population N and evolutionary round T, which impact the number of sampling reasoning paths, therefore we adjust them to match the experimental setup. The setting for CORE-EVO are $\{N = 5, T = 0\}$, $\{N = 5, T = 5\}$, $\{N = 10, T = 10\}$, and $\{N = 10, T = 20\}$ on the scale of 5, 10, 20, and 30 sampling reasoning paths, respectively.

As shown in Figure. 2, our algorithm scales the inference compute more effectively than both the best-of-N and self-consistency methods on two benchmarks for the Llama3.1-8B model. The performance of the self-consistency method quickly saturates after sampling 20 reasoning paths, while the best-of-N shows a favorable increase in performance in terms of inference scaling. At the small scale, the CORE-EVO shows a marginal improvement compared to self-consistency, and it is even surpassed by the best-of-N method on the SocialIQA benchmark. However, at higher scales, 20 and 30-reasoning paths, our method illustrates a significant gap in terms of performance compared with both best-of-N and self-consistency methods.

Model	CoT	CORE-REVO	Δ
Llama3.1-8B	77.12	81.98	↑ 4.86
Llama3.2-3B	72.71	79.25	↑ 6.54
Llama3.2-1B	27.27	44.48	↑ 17.21

Table 4: The experiment on the inference scaling ability of different model sizes for the performance of CORE-EVO on the CommonsenseQA benchmark.

The results in Table. 4 illustrates how inference performance scales across models of varying sizes, from 1 billion to 8 billion parameters. We observe that performance improvements are inversely proportional to model size, indicating our approach is more effective for smaller language models. Notably, significant gains are achieved

with smaller models, approximately 7% improvement for Llama3.2-3B and an impressive 17% for Llama3.2-1B.

3.4 Ablation Studies

Method	CSQA	SocialIQA	Average
Standard CoT	75.54	71.23	73.38
Ours CoT	77.12	72.38	74.75

Table 5: The experiment on the impact of different CoT techniques on CommonsenseQA(CSQA) and SocialIQA benchmarks using Llama3.1-8B.

Impact of Chain-of-thought techniques. We study the impact of different CoT prompting techniques. We compare our CoT technique, described in 2.1, with the standard CoT method, which basically prompts the LLMs with a simple instruction "*Let's think step by step*". As shown in Table. 5, our CoT technique outperforms the simple CoT method by about 1.5% average accuracy on two benchmarks. These results demonstrate that the structure CoT and the relevant generated knowledge are crucial for commonsense reasoning tasks. This experimental result aligns with previous studies (Liu et al., 2022) that have shown that generated knowledge is able to enhance the reasoning abilities of LLMs.

	CSQA	SIQA	Average
CORE-EVO			
w/ rottle-wheel	81.98	76.25	**79.11(0.0)**
w/ tournament	80.78	75.14	77.96(-1.15)
w/o mutation	79.93	74.16	77.04(-2.07)
w/o crossover	80.85	74.94	77.89(-1.22)

Table 6: The ablation studies for the contribution of selection methods and evolutionary operation to the CORE-EVO accuracy on CommonsenseQA (CSQA) and SocialIQA (SIQA) benchmarks using Llama3.1-8B.

Impact of evolutionary operations and selection methods. We study the contribution of each evolutionary operation and the selection methods to the CORE-EVO performance. As shown in Table. 6, the rottle-wheel selection method outperforms the tournament selection method by 1.15% average accuracy on the CommonsenseQA and Social IQA benchmarks. Thus, we select the rottle-wheel as the selection method for our proposed method. There is a significant drop, approximately 2%, in average accuracy when using only the crossover operation. This result shows that the mutation operation plays a crucial role in our method. There is a similar observation with using only the mutation operation, thus the trade-off between mutation and crossover is critical for the CORE-EVO. The Figure 5 demonstrates a comprehensive study of the mutation-crossover tradeoff.

	CSQA	SIQA	Average
CORE-EVO	81.98	76.25	**79.11(0.0)**
w/ quantile-α (Gupta et al., 2024)	80.13	75.11	77.61(-1.50)
w/ logprob	79.89	74.82	77.35(-1.76)

Table 7: The ablation studies on the impact of different fitness functions on the CORE-EVO accuracy on CommonsenseQA (CSQA) and SocialIQA (SIQA) benchmarks using Llama3.1-8B.

Impact of fitness function. We study the impact of different fitness functions on our method. We utilize the quantile-α function (Gupta et al., 2024), the normalized logprob function, and the entailment verifier function described in section. 2.4 as the fitness function. Both the quantile-α and the normalized logprob function are used to measure the confidence level of reasoning paths of language models, while our fitness function measures the overall entailment score of reasoning paths. The results in Table. 7 shows that our fitness function is more well-suited for the evolutionary search algorithm. We also conduct an ablation study regarding the ranking ability of those fitness function, the results are shown in Table. 8

4 Related Work

4.1 Evolutionary Algorithm for LLMs

Recent research has investigated the integration of large language models with evolutionary algorithms for optimization tasks. This includes work on numerical optimization (Liu et al., 2025; Brahmachary et al., 2025) and combinatorial optimization (Ye et al., 2024). Furthermore, evolutionary search techniques have been applied to prompt optimization to enhance performance on downstream tasks (Yuan et al., 2024; Fernando et al., 2024; Guo et al., 2024). Notably, (Yuan et al., 2024) proposed an agent-based method, called EvoAgent, by leveraging the power of LLMs to address several challenging NLP tasks. Another line of work is to leverage LLMs as evolutionary operators (Meyerson et al., 2024; Lehman et al., 2023), facilitating the integration of evolutionary algorithms with powerful language models for text generation applications.

The recent work (Lee et al., 2025) represents the most closely related research to our approach. (Lee et al., 2025) introduces a novel approach to combine LLMs with evolutionary search algorithms to tackle natural language planning tasks. In contrast, our work aims to overcome the commonsense reasoning shortcomings of LLMs by introducing a specialized score function tailored for commonsense reasoning tasks.

4.2 Reasoning with LLMs

The advent of Large Language Models has successfully tackled numerous complex NLP challenges, including reasoning capabilities that approach human-level performance. Researchers primarily employ two methodologies to harness LLMs for reasoning tasks: finetuning-based and prompt-based techniques.

The finetuning-based approaches are pioneered by (Sanh et al., 2022; Wei et al., 2022a; Thoppilan et al., 2022) through fine-tuning on diverse NLP tasks and data, yielding impressive zero-shot performance on unseen tasks. The Orca (Mukherjee et al., 2023) and its advanced version (Mitra et al., 2023) demonstrated that smaller-scale LLMs are able to achieve sophisticated reasoning abilities through extensive instruction-based fine-tuning. Another line of work (Ho et al., 2023; Magister et al., 2023; Hsieh et al., 2023) is generating supervision data from larger and more capable language models to distill reasoning abilities to smaller models for efficiency. To improve the generalization ability of out-of-domain, (PENG et al., 2025) proposed the ReGenesis framework to self-synthesize reasoning paths for post fine-tuning LLMs.

As regards the prompt-based method, chain-of-thought (CoT) (Wei et al., 2022b) is the first method to facilitate the reasoning of LLMs by a step-by-step thinking process. (Kojima et al., 2022) introduced zero-shot CoT to eliminate the need for curated high-quality reasoning paths for models fine-tuning. Building on this foundation, (Zhang et al., 2023) developed AutoCoT, which autonomously generates few-shot demonstrations with a CoT reasoning structure to tackle the scalable issue of manual CoT methods. Additionally, several prior works (Agarwal et al., 2024; Yang et al., 2024b) employ in-context learning techniques to enhance the reasoning capabilities of pretrained language models.

A new emerging direction is to scale the inference time to improve the thinking abilities of language models. The Self-Refine framework proposed by (Madaan et al., 2023) uses an iterative process that improves reasoning by refining previous reasoning paths. A simple yet effective method (Wang et al., 2023) is to generate a diverse set of reasoning paths and then marginalize over them to draw a conclusion. To enhance the strategic planning ability of LLMs, previous works (Yao et al., 2023; Hao et al., 2023) have explored the integration of tree-based structures with the thinking process to enhance the strategic planning ability of language models by systematically organizing their reasoning processes. The recent study (Snell et al., 2025) provides analysis and evidence indicating that scaling inference compute is more effective than scaling training compute, offering a novel approach for developing more capable reasoning agents.

5 Conclusion

We propose the COmmensense REasoning EVOlution method which is a general inference scaling method to advance commonsense reasoning capabilities of LLMs. Our method leverages the self-feedback ability and prompting techniques of LLMs to implement the mutation and crossover operations for integrating the evolutionary search for commonsense reasoning tasks. Our method is capable of addressing the limitations of the best-of-N and self-consistency method, searching reasoning paths in the local optima in reasoning spaces, by iteratively performing evolutionary operations and population refinement based on the pretrained entailment verifier model. Our experimental results demonstrate that the CORE-EVO outperforms inference scaling baselines by a significant margin in terms of accuracy on three popular benchmarks across two LLM backbones, the Llama3.1-8B and Qwen1.5-7B.

Limitations

We study the inference scaling ability of pretrained LLMs for commonsense reasoning tasks. The research goal is to advance the reasoning ability of pretrained LLMs without requiring fine-tuning. Below are the limitations of our work:

- Our method relies heavily on hand-crafted prompts for different tasks, and the performance of our method is bound by the performance of base language models.

- We are not able to systematically study the inference scaling ability of very large language models, which have more than 8 billion parameters, on commonsense reasoning tasks due to time and computational constraints.

- Our method requires generating a greater number of tokens compared to alternative techniques, resulting in lower token efficiency.

Ethics Statement

We recognize that our proposed methodology may produce potentially misleading reasoning paths as a result of inherent biases present in the underlying language models. Although generating biased reasoning is not the intended purpose of our approach, we acknowledge this limitation. This risk can be mitigated through two ways: careful design of input prompts, or thorough evaluation and selection of language models with reduced bias profiles. We remain committed to responsible development and application of our method while acknowledging these inherent challenges.

References

Rishabh Agarwal, Avi Singh, Lei M Zhang, Bernd Bohnet, Luis Rosias, Stephanie C.Y. Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, John D Co-Reyes, Eric Chu, Feryal Behbahani, Aleksandra Faust, and Hugo Larochelle. 2024. Many-shot in-context learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. Piqa: Reasoning about physical commonsense in natural language. In *AAAI Conference on Artificial Intelligence*.

Shuvayan Brahmachary, Subodh M Joshi, Aniruddha Panda, Kaushik Koneripalli, Arun Kumar Sagotra, Harshil Patel, Ankush Sharma, Ameya D Jagtap, and Kaushic Kalyanaraman. 2025. Large language model-based evolutionary optimizer: Reasoning with elitism. *Neurocomputing*, 622:129272.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. 2024. Promptbreeder: self-referential self-improvement via prompt evolution. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Lin Gui, Cristina Gârbacea, and Victor Veitch. 2024. Bonbon alignment for large language models and the sweetness of best-of-n sampling. *arXiv preprint arXiv:2406.00832*.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2024. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. In *The Twelfth International Conference on Learning Representations*.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. 2024. Language model cascades: Token-level uncertainty and beyond. In *The Twelfth International Conference on Learning Representations*.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8154–8173, Singapore. Association for Computational Linguistics.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14852–14882, Toronto, Canada. Association for Computational Linguistics.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8003–8017, Toronto, Canada. Association for Computational Linguistics.

Mingyu Jin, Qinkai Yu, Dong Shu, Haiyan Zhao, Wenyue Hua, Yanda Meng, Yongfeng Zhang, and Mengnan Du. 2024. The impact of reasoning step length on large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1830–1842, Bangkok, Thailand. Association for Computational Linguistics.

Minki Kang, Seanie Lee, Jinheon Baek, Kenji Kawaguchi, and Sung Ju Hwang. 2023. Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks. *Advances in Neural Information Processing Systems*, 36:48573–48602.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*.

Kuang-Huei Lee, Ian Fischer, Yueh-Hua Wu, Dave Marwood, Shumeet Baluja, Dale Schuurmans, and Xinyun Chen. 2025. Evolving deeper llm thinking. *arXiv preprint arXiv:2501.09891*.

Joel Lehman, Jonathan Gordon, Shawn Jain, Kamal Ndousse, Cathy Yeh, and Kenneth O Stanley. 2023. Evolution through large models. In *Handbook of evolutionary machine learning*, pages 331–366. Springer.

Fei Liu, Xi Lin, Shunyu Yao, Zhenkun Wang, Xialiang Tong, Mingxuan Yuan, and Qingfu Zhang. 2025. Large language model for multiobjective evolutionary optimization. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 178–191. Springer.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, Toronto, Canada. Association for Computational Linguistics.

Elliot Meyerson, Mark J. Nelson, Herbie Bradley, Adam Gaier, Arash Moradi, Amy K. Hoover, and Joel Lehman. 2024. Language model crossover: Variation through few-shot prompting. *ACM Trans. Evol. Learn. Optim.*, 4(4).

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Codas, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, and 1 others. 2023. Orca 2: Teaching small language models how to reason. *arXiv preprint arXiv:2311.11045*.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of gpt-4. *arXiv preprint arXiv:2306.02707*.

XIANGYU PENG, Congying Xia, Xinyi Yang, Caiming Xiong, Chien-Sheng Wu, and Chen Xing. 2025. Regenesis: LLMs can grow into reasoning generalists via self-improvement. In *The Thirteenth International Conference on Learning Representations*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, and 21 others. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Soumya Sanyal, Tianyi Xiao, Jiacheng Liu, Wenya Wang, and Xiang Ren. 2024. Are machines better at complex reasoning? unveiling human-machine inference gaps in entailment verification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10361–10386, Bangkok, Thailand. Association for Computational Linguistics.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. Social IQa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, Hong Kong, China. Association for Computational Linguistics.

Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, and 1 others. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022a. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Ling Yang, Zhaochen Yu, Tianjun Zhang, Shiyi Cao, Minkai Xu, Wentao Zhang, Joseph E. Gonzalez, and Bin CUI. 2024b. Buffer of thoughts: Thought-augmented reasoning with large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Haoran Ye, Jiarui Wang, Zhiguang Cao, Federico Berto, Chuanbo Hua, Haeyeon Kim, Jinkyoo Park, and Guojie Song. 2024. Reevo: Large language models as hyper-heuristics with reflective evolution. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Siyu Yuan, Kaitao Song, Jiangjie Chen, Xu Tan, Dongsheng Li, and Deqing Yang. 2024. Evoagent: Towards automatic multi-agent generation via evolutionary algorithms. In *NeurIPS 2024 Workshop on Open-World Agents*.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2023. Automatic chain of thought prompting in large language models. In *The Eleventh International Conference on Learning Representations*.

A Additional Ablation Study

A.1 Additional ablation studies

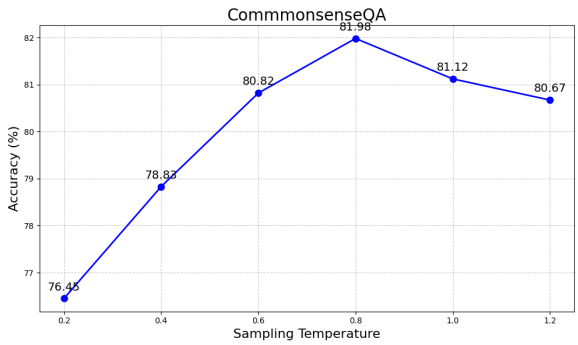


Figure 3: The experiment for choosing the sampling temperature to initialize population with the setting $\{N = 10, T = 20\}$ for the CORE-EVO on the CommonsenseQA benchmark for Llama3.1-8B model

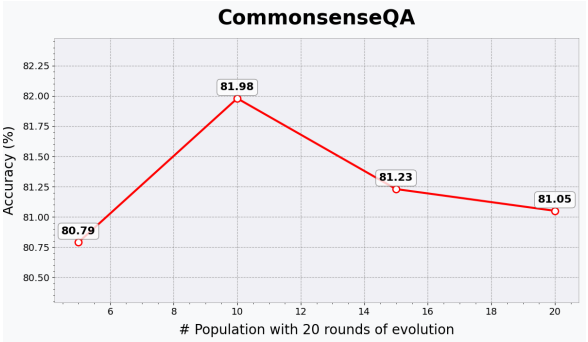


Figure 4: The experiment for choosing the number of initial population for a fixed number of evolutionary rounds $T = 20$ on the CommonsenseQA benchmark for Llama3.1-8B model

The Figure. 3 illustrates the hyperparameter selection for the sampling temperature parameter for our method. The temperature at temp $= 0.8$ achieves the highest performance, therefore, all our experiments are conducted using the temperature of 0.8. The ablation study for choosing the number of the initial population for our method is demonstrated in Figure. 4. The number of population $N = 10$ achieves the highest performance on the CommonsenseQA benchmark, thus, it is used for experiments throughout this work.

In Table. 8, we examine the effectiveness of different fitness/rankin functions on the CommonsenseQA benchmark with Llama3.1-8B. We searched for optimal reasoning paths using the best-of-N approach with different fitness functions applied. Our findings show that the pretrained verifier function delivers the highest average accuracy

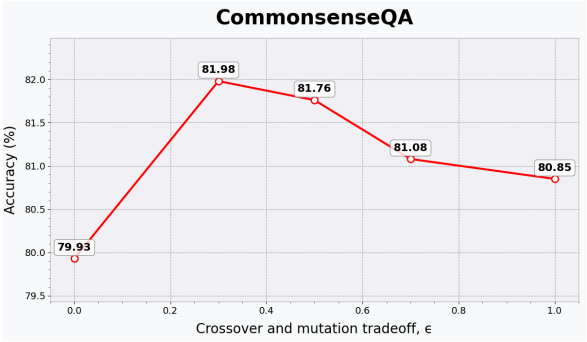


Figure 5: The experiment for mutation-crossover tradeoff on the CommonsenseQA benchmark for Llama3.1-8B model

fitness function	CSQA	SIQA	Average
pretrained verifier	80.34	75.23	**77.79**
quantile-α	78.54	74.61	76.57
logprob	77.17	73.86	75.51

Table 8: The ablation study on the effectiveness of different fitness functions for Llama3.1-8B. The inference scaling method is best-of-N with different fitness/ranking functions to search for the best solution.

when compared to both the normalized logprob and the quantile-α methods.

A.2 Computation

We do not train any new model in this paper, and we instead propose a new inference scaling technique. Inference is conducted on A100 GPUs and costs about 300 GPU hours in total. Our method is implemented with PyTorch and the Huggingface Transformers library.

B Prompt

B.1 CommonsenseQA prompts

CommonsenseQA CoT Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its relevant knowledge in detail. Also, determine the context in the given question in detail.
Step 2. Randomly choose an option to simulate an example using information from previous steps. Try to be as specific as possible.
Step 3. Conclude the answer based on information from step 2. Answer by choosing the most correct option. Strictly output a single character.

Please response your answer by using JSON format: <'step i'>: <content>..

<In-context examples>

Question: {input-question}
Answer:

CommonsenseQA Crossover Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its relevant knowledge in detail. Also, determine the context in the given question in detail.
Step 2. Randomly choose an option to simulate an example using information from previous steps. Try to be as specific as possible.
Step 3. Conclude the answer based on information from step 2. Answer by choosing the most correct option. Strictly output a single character.

Please response your answer by using JSON format: <'step i'>: <content>..

You will be given two reasoning paths. Randomly mix the first three steps between two reasoning paths, but keep the same step orders. Please conclude the reasoning in step 4 based on the first three steps. Please respond the answer by using JSON format without explanation: <'step i'>: <content>.

Question: {input-question}
The first reasoning path: {path-1}
The second reasoning path: {path-2}
The new reasoning path:

CommonsenseQA Reflection Prompt

Instruction: You are an advanced reasoning agent that can improve based on self refection. You will be given a previous reasoning trial in which you were given access to relevant context and a question to answer. Please provide feedback on the previous reasoning path. The feedback indicates missing concepts. Provide a concise and high-level feedback for the previous reasoning path.

Context: {input-question}
Previous trial: {previous reasoning path}
Reflection:

CommonsenseQA Mutation Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its relevant knowledge in detail. Also, determine the context in the given question in detail.
Step 2. Randomly choose an option to simulate an example using information from previous steps. Try to be as specific as possible.
Step 3. Conclude the answer based on information from step 2. Answer by choosing the most correct option. Strictly output a single character.

Generate the new reasoning path based on the reflection from the previous reasoning path. Please refine the previous path and correct any false claims. Please respond the answer by using JSON format without explanation: <'step i'>: <content>.

Question: {input-question}
Previous reasoning path: {previous reasoning path}
Reflection: {feedback}
The new reasoning path:

B.2 PIQA prompts

PIQA CoT Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its physical properties in the question. Also, determine the differences in the situation between the two options.
Step 2. Simulate an example based on the previous step's information.
Step 3. Conclude the answer information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Please respond with the answer by using JSON format: <'step i'>: <content>.

<In-context examples>

Question: {input-question}
Answer:

PIQA Crossover Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its physical properties in the question. Also, determine the differences in the situation between the two options.
Step 2. Simulate an example based on the previous step's information.
Step 3. Conclude the answer information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Please respond with the answer by using JSON format: <'step i'>: <content>.

You will be given two reasoning paths. Randomly mix the first three steps between two reasoning paths, but keep the same step orders.

Please conclude the reasoning in step 4 based on the first three steps.

Question: {input-question}
The first reasoning path: {path-1}
The second reasoning path: {path-2}
The new reasoning path:

PIQA Reflection Prompt

Instruction: You are an advanced reasoning agent that can improve based on self refection. You will be given a previous reasoning trial for a physical situation in which you were given access to relevant context and a question to answer. Please provide feedback on the previous reasoning path. Provide a concise and high-level feedback for the previous reasoning path.

Context: {input-question}
Previous trial: {previous reasoning path}
Reflection:

PIQA Mutation Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Determine the object and its physical properties in the question. Also, determine the differences in the situation between the two options.
Step 2. Simulate an example based on the previous step's information.
Step 3. Conclude the answer information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Generate the new reasoning path based on the reflection from previous reasoning path. Please refine the previous path and correct any false claims. Please conclude the reasoning in step 4 based on the first three steps.

Please respond with the answer by using JSON format without explanation: <'step i'>: <content>.

Question: {input-question}
Previous reasoning path: {previous reasoning path}
Reflection: {feedback}
The new reasoning path:

B.3 SIQA prompts

SIQA CoT Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Analyze the context and extract factual knowledge based on the context. Also, analyze the given question.
Step 2. Randomly choose an option to simulate a situation using information from the previous step.
Step 3. Conclude the answer based on information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Please respond with the answer by using JSON format: <'step i'>: <content>.

<In-context examples>

Question: {input-question}
Answer:

SIQA Crossover Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Analyze the context and extract factual knowledge based on the context. Also, analyze the given question.
Step 2. Randomly choose an option to simulate a situation using information from the previous step.
Step 3. Conclude the answer based on information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Please respond with the answer by using JSON format: <'step i'>: <content>.

You will be given two reasoning paths. Randomly mix the first three steps between two reasoning paths, but keep the same step orders.

Please conclude the reasoning in step 4 based on the first three steps.

Question: {input-question}
The first reasoning path: {path-1}
The second reasoning path: {path-2}
The new reasoning path:

SIQA Reflection Prompt

Instruction: You are an advanced reasoning agent that can improve based on self refection. You will be given a previous reasoning trial for social interaction in which you were given access to relevant context and a question to answer. Please provide feedback on the previous reasoning path. Provide a concise and high-level feedback for the previous reasoning path.

Context: {input-question}
Previous trial: {previous reasoning path}
Reflection:

SIQA Mutation Prompt

Instruction: You are a helpful assistant. You are able to understand commonsense knowledge and answer related commonsense questions perfectly.
You should reason your answer step by step concisely. Here are the reasoning steps:

Step 1. Analyze the context and extract factual knowledge based on the context. Also, analyze the given question.
Step 2. Randomly choose an option to simulate a situation using information from the previous step.
Step 3. Conclude the answer based on information from step 1 and 2. Answer by choosing the most correct option. Strictly output a single character.

Generate the new reasoning path based on the reflection from previous reasoning path. Please refine the previous path and correct any false claims. Please conclude the reasoning in step 4 based on the first three steps.

Please respond with the answer by using JSON format without explanation: <'step i'>: <content>.

Question: {input-question}
Previous reasoning path: {previous reasoning path}
Reflection: {feedback}
The new reasoning path:

C Qualitative Examples Analysis

We perform a qualitative evaluation on a sample of 50 questions from the CommonsenseQA benchmark to evaluate reasoning quality across different approaches. Our findings demonstrate that CORE-EVO generates significantly more nuanced and comprehensive reasoning paths compared to both best-of-N and self-consistency baselines. Furthermore, CORE-EVO incorporates supplementary evidence and contextual details that facilitate more robust conclusion formation. The following examples illustrate the qualitative distinctions between CORE-EVO and baseline methods, self-consistency and best-of-N. Figure 10 demonstrates that while all methods correctly identify the answer, baseline approaches directly select the UK without sufficient justification. In contrast, our method follows a more sophisticated and thorough reasoning process, explicitly considering Ireland as a potential candidate before ultimately determining Great Britain as the correct answer.

	CommonsenseQA
Question	James was looking for a good place to buy farmland. Where might he look? A) Midwest B) Countryside C) Estate D) Farming area E) Illinois
Best-of-N	**Step 1:** Farmland is typically located in areas with a suitable climate and geography for agriculture, often in rural or countryside settings. The question asks where James might look for a good place to buy farmland, indicating a search for a specific geographic location. **Step 2:** For example, the Midwest region in the United States is known for its fertile soil and favorable climate for farming, making it an ideal location for buying farmland. **Step 3:** Farming areas or regions with suitable conditions for agriculture are likely where James would find suitable farmland for purchase. D
Self-consistency	**Step 1:** The object is farmland, which is a piece of land used for agriculture, typically for crops or livestock. The question asks where James might look for a good place to buy farmland, suggesting a location with suitable conditions for farming. **Step 2:** For example, the midwest region in the United States is known for its fertile soil and suitable climate for farming, making it a prime location for buying farmland. **Step 3:** Farming areas or regions with suitable climate and soil conditions are ideal for buying farmland. D
CORE-EVO	**Step 1:** Farmland is typically located in areas suitable for agriculture, often in rural regions with fertile soil and adequate water supply. James was looking for a good place to buy farmland, indicating he is seeking a location with suitable conditions for agriculture. **Step 2:** For example, The Midwest region in the United States is known for its fertile plains and is often considered an ideal place to buy farmland due to its suitable climate and soil conditions. **Step 3:** The Midwest region offers suitable conditions for agriculture, including fertile soil and adequate water supply, making it a prime location for buying farmland. A

Table 9: This qualitative example where the CORE-EVO provides a more accurate reasoning path over baselines. The reasoning paths are selected based on the highest score in the population for each method. The issues with the reasoning path are highlighted in red. The correct answer and accurate explanation are highlighted in blue.

	CommonsenseQA
Question	What island country is ferret popular? A) Own home B) North Carolina C) Great britain D) Hutch E) Outdoors
Best-of-N	**Step 1:** Ferrets are popular pets in many countries, but they are often associated with the United Kingdom, where they have been kept as companions for centuries. The question asks which island country ferrets are popular in, indicating a specific geographic location. **Step 2:** For example, in the United Kingdom, particularly in England, ferrets are a common household pet and are often kept in domestic settings. **Step 3:** The popularity of ferrets as pets in the UK is well-documented, and they are often associated with British culture. C
Self-consistency	**Step 1:** The object is a ferret, a domesticated mammal known for its playful and curious nature, often kept as pets. The question asks in which island country ferrets are popular pets. **Step 2:** For example, in the United Kingdom, particularly in England, ferrets are relatively popular pets, and their care and housing are well-documented. **Step 3:** The UK, including England, has a history of keeping ferrets as pets, and they are relatively well-known in the country. C
CORE-EVO	**Step 1:** The object is a ferret, a domesticated mammal known for its playful and curious nature, often kept as pets. The question asks in which island country ferrets are popular, specifying a country that is both an island and has a notable ferret population. **Step 2:** For example, while Ireland is an island country with a known ferret population, it is not the correct answer choice. **Step 3:** Instead, the answer choice C, Great Britain, is a more accurate match, as ferrets are popular pets in the UK, including in England, Scotland, and Wales, which are part of Great Britain. Great Britain meets the criteria of being an island country and has a known ferret population, making it a suitable answer, and it is also the correct answer choice. C

Table 10: This qualitative example where the CORE-EVO provides a more accurate reasoning path over baselines. The reasoning paths are selected based on the highest score in the population for each method. The issues with the reasoning path are highlighted in red. The correct answer and accurate explanation are highlighted in blue.