Self-DPO: Improving Reasoning Across Domains with Small Models via Self-Generated Preferences

Anonymous ACL submission

Abstract

Large Language Models (LLMs) demonstrate strong performance on structured reasoning tasks like mathematics and coding, yet this success often fails to translate to specialized domains. Bridging this gap typically relies on knowledge distillation from computationally-intensive teacher models, creating significant barriers to accessible research. To overcome these barriers, we propose Self-DPO, a novel framework that enables smaller models to improve their domain reasoning autonomously within a teacher-free paradigm. Self-DPO empowers a model to generate its own reasoning pathways. It then employs a lightweight mechanism to assess the quality of individual reasoning steps, generating a contribution vector for each path that captures its step-by-step logical flow. Preference pairs for Direct Preference Optimization (DPO) are then constructed by selecting a concise, correct path as the chosen response, and crucially, identifying a path that is maximally divergent in its reasoning trajectory, quantified using Dynamic Time Warping distance between contribution vectors, as the rejected response. This process-oriented contrast provides a more nuanced learning signal than simple outcome-based filtering. We validate Self-DPO across diverse domains, demonstrating significant performance improvements and achieving results comparable to distillation-based methods, but without any reliance on larger teacher models. This establishes a practical and resource-efficient pathway for enhancing reasoning in specialized models.

1 Introduction

Large Language Models (LLMs) have achieved state-of-the-art performance on complex reasoning tasks by leveraging techniques like Chain-of-Thought prompting (Wei et al., 2022; Kojima et al., 2022). While successful in structured domains like mathematics and code generation (Jaech et al., 2024; Lewkowycz et al., 2022; Chen et al., 2021), this success often fails to translate to specialized fields such as medicine and law, where performance gains remain limited (Hendrycks et al., 2021; Guha et al., 2023; Singhal et al., 2023). Therefore, developing methods to effectively enhance the reasoning capabilities of LLMs for these critical domains is a pressing research priority.

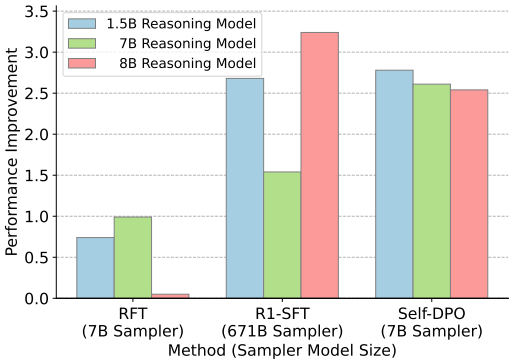


Figure 1: Fine-tuning method comparison by performance improvement. Our method, Self-DPO using its own 7B parameters for sampling, shows strong performance comparable to distillation from a 671B model.

Enhancing domain-specific reasoning in LLMs is primarily constrained by training data acquisition. Supervised Fine-Tuning (SFT) often utilizes either expert annotations (which are costly and difficult to scale) or synthetic data. A dominant approach for generating such data is knowledge distillation, where a smaller student model learns from a significantly larger teacher model (Team, 2025; An et al., 2023). While effective, this paradigm creates a dependency on proprietary or computationally-intensive teacher models. An alternative, self-sampling (Wang et al., 2023; Min et al., 2024), avoids this dependency but faces a fundamental tension: its quality is inherently capped by the base model's own limited capabilities (Wang et al., 2024), constraining the potential for self-improvement. Rule-based Reinforcement Learning (RL), such as Group Relative Policy Optimization

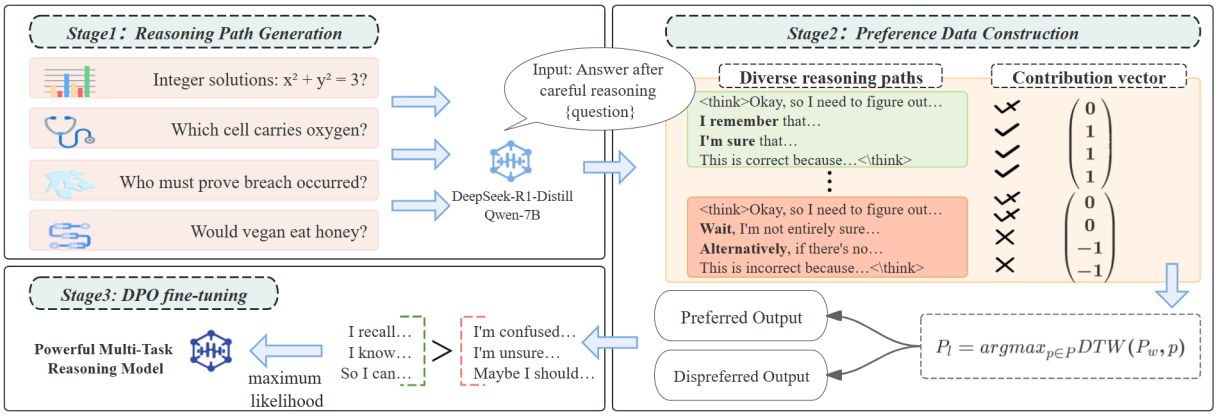


Figure 2: Overview of the proposed Self-DPO framework. **Stage 1**: The base LLM generates multiple candidate reasoning paths for diverse input questions via self-sampling. **Stage 2**: Each path is evaluated by segmenting it into steps and assigning a contribution score per step using NLI, yielding a contribution vector. A preferred path (P_w, correct and concise) and a dispreferred path (P_l, maximally divergent based on DTW distance between contribution vectors) are selected. **Stage 3**: The preference pairs (P_w, P_l) are used to fine-tune the base LLM via DPO, resulting in a model with improved multi-task reasoning capabilities.

(GRPO) (Shao et al., 2024), offers an alternative via reward functions, but typically incurs high computational costs, suffers from training instability (Casper et al., 2023), and struggles with effective reward function design for complex reasoning.

Motivated by the substantial costs and limitations of existing methods, we explore whether capable language models can autonomously enhance their domain-specific reasoning without relying on external datasets or larger teacher models. To this end, we propose Self-DPO, a novel framework for efficient and self-sufficient domain reasoning improvement. Self-DPO operates by having the model initially self-sample multiple reasoning pathways for domain problems. Crucially, it then assesses the contribution of individual steps within these pathways to guide the selection of preference pairs. Concise and correct reasoning sequences serve as positive examples (chosen), while less effective or erroneous pathways are designated as negative examples (rejected), informed by this contribution analysis. This automatically constructed preference dataset directly fuels Direct Preference Optimization (DPO)(Rafailov et al., 2023). By optimizing on these self-generated preferences, the model learns to favor effective reasoning pathways while avoiding flawed ones. This process establishes a self-sufficient improvement loop, enhancing domain-specific capabilities efficiently without reliance on costly external data or knowledge distillation from larger models.

We evaluated Self-DPO on challenging domain-specific reasoning tasks. Our primary baseline was SFT using synthetic data distilled from a significantly larger model. This setup allows for a direct comparison between our autonomous, teacher-free method and the resource-intensive distillation paradigm. Empirical results demonstrate that Self-DPO achieves performance gains comparable to SFT using costly data distilled from a significantly larger model, validating it as a powerful yet economical method for domain reasoning enhancement. Our contributions are as follows:

- We propose Self-DPO, a novel framework that enables LLMs to autonomously improve domain-specific reasoning in a teacher-free setting, using only self-generated data.
- We introduce a process-oriented method for automatically constructing preference data, which assesses the step-by-step logical contribution within reasoning trajectories, rather than relying only on final outcomes.
- Our experiments demonstrate that Self-DPO achieves reasoning performance comparable to distillation-based SFT, establishing it as a resource-efficient and accessible alternative for domain adaptation.

2 Related Work

2.1 Reasoning Models

With the recent advancements in reasoning models (Jaech et al., 2024; DeepSeek-AI et al., 2025), many works (Team, 2025; Labs, 2025; Min et al., 2024) have utilized the approach of employing the GRPO method (Shao et al., 2024) combined with distilled data for SFT to train models with long

reasoning capabilities. To address the issue of redundant reasoning in long-reasoning models, numerous studies (Muennighoff et al., 2025; Aggarwal and Welleck, 2025; Wang et al., 2025) also explore test-time scaling along with reasoning path optimization. In contrast to prior works primarily focusing on optimization in mathematics and coding, this paper investigates a self-sampling synthetic data training approach to enhance reasoning abilities across multiple domains, while reducing domain adaptation costs by avoiding distillation from large models.

2.2 Self-improvement for Reasoning Models

Several studies have investigated the use of smaller language models to autonomously generate synthetic data for enhancing reasoning capabilities. For instance, STaR (Zelikman et al., 2022) iteratively improves complex reasoning by augmenting limited reasoning examples with extensive non-reasoning data. RFT (Yuan et al., 2023) employs rejection sampling, refining the fine-tuning dataset by selecting the highest-quality responses from multiple model generations. LEMA (An et al., 2023) focuses on learning from errors, collecting incorrect reasoning paths and utilizing GPT-4 to generate corrective feedback. However, Huang et al. (2024) highlighted the inherent difficulty for LLMs to self-correct reasoning errors without external guidance. Our approach distinguishes itself by simplifying the synthesis of preference pairs for DPO. We leverage both correct and flawed reasoning paths generated by the small model itself, thereby circumventing the need for explicit path correction while improving reasoning performance across diverse domains. While Wang et al. (2024) also utilizes a self-sampling DPO strategy, their preference data construction relies solely on the correctness of the final answer. In contrast, our method implements a more granular filtering mechanism based on an analysis of the reasoning process itself.

3 Methodology

In this section, we describe our approach to enhancing the reasoning capabilities of low-parameter language models by utilizing self-sampled reasoning paths to automatically construct a preference dataset for DPO fine-tuning. Algorithm 1 provides a high-level overview.

Algorithm 1 Overview of Self-Sampling DPO

Require: Base LLM M_{ref}, QA dataset Q, self-sampling count n, NLI model M_{NLI}

Initialize preference dataset $\mathcal{D} \leftarrow \emptyset$
for all problem $q \in Q$ **do**
  Generate n paths $\{P_1, ..., P_n\} \leftarrow M_{ref}(q)$
  Compute contribution vectors $\{\mathbf{v}_1, ..., \mathbf{v}_n\}$ via segmentation and NLI scoring
  Select preferred path P_w: shortest correct path in $\{P_1, ..., P_n\}$
  Select dispreferred path P_l: path P_k maximizing $\text{DTW}(\mathbf{v}_w, \mathbf{v}_k)$ for $P_k \neq P_w$
  Add (q, P_w, P_l) to $\mathcal{D}$
end for
Fine-tune M_{ref} on $\mathcal{D}$ to obtain M_θ
return M_θ

3.1 Reasoning Path Generation and Evaluation

Generating Diverse Solutions The methodology initiates by employing the base low-parameter language model (M_{ref}) to explore potential solution pathways for a given problem q. We employ a self-sampling strategy, setting a relatively high temperature during generation. This encourages M_{ref} to produce n distinct reasoning paths $\{P_1, P_2, ..., P_n\}$ for the input q, capturing varied reasoning steps and conclusions.

Path Segmentation Each generated path P_i undergoes a structured evaluation process designed to assess its deductive quality and relevance with respect to the ground truth answer A^+. The initial phase of this evaluation involves semantic segmentation: each path P_i is parsed into a sequence of k_i coherent reasoning units or steps, denoted as $P_i = \{s_{i,1}, s_{i,2}, ..., s_{i,k_i}\}$. This segmentation facilitates a more granular analysis of the inherent reasoning trajectory within each path.

Contribution Evaluation Following segmentation, the contribution of each segment $s_{i,j}$ towards the final answer is evaluated. A pre-trained Natural Language Inference (NLI) model, specifically DeBERTa-v2-XXL-MNLI (He et al., 2021) is employed for this purpose. For each segment $s_{i,j}$, two hypotheses are formulated: one corresponding to the correct answer A^+ and the other to a known incorrect answer A^-. The NLI model ascertains the entailment relationship between the segment $s_{i,j}$ (premise) and these hypotheses. Based on the

entailment outcome, a contribution score $c_{i,j}$ is assigned as follows:

$$c_{i,j} = \begin{cases} +1 & \text{if } s_{i,j} \Rightarrow A^+ \\ -1 & \text{if } s_{i,j} \Rightarrow A^- \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

This scoring quantifies the contribution of each reasoning step towards the correct (A^+) or an incorrect (A^-) conclusion. These segment-level scores are subsequently aggregated sequentially into a contribution vector $\mathbf{v}_i = [c_{i,1}, c_{i,2}, ..., c_{i,k_i}]$ associated with each reasoning path P_i. Each vector $\mathbf{v}_i$ thus encapsulates the step-wise reasoning profile of its corresponding path.

3.2 Preference Data Construction via Path Dissimilarity

The evaluated reasoning paths $\{P_1, ..., P_n\}$ and associated contribution vectors $\{\mathbf{v}_1, ..., \mathbf{v}_n\}$ serve as the foundation for constructing the preference dataset required by DPO. Specifically, for each input problem q, we identify a preferred response P_w and a corresponding dispreferred response P_l to form training instances (q, P_w, P_l).

The preferred response P_w is identified from the set of generated paths that yield the correct answer A^+. Among these valid candidates, we select the path with the minimum number of semantic segments ($\min k_i$) as P_w. This heuristic selects for reasoning paths that are both correct and maximally concise, prioritizing reasoning efficiency. The contribution vector associated with the selected P_w is subsequently denoted as $\mathbf{v}_w$.

Having identified the preferred response P_w and its contribution vector $\mathbf{v}_w$, the corresponding dispreferred response P_l is then selected to maximize divergence from P_w's reasoning profile, providing a challenging negative sample for DPO. Measuring the divergence between reasoning paths presents challenges: the corresponding contribution vectors $\mathbf{v}_w$ and $\mathbf{v}_k$ often have different lengths, precluding direct vector comparison. Moreover, the sequence of contribution scores $(+1, -1, 0)$ reflects the step-by-step unfolding of the reasoning process, requiring a distance metric sensitive to this sequential pattern. To address these challenges, we quantify the divergence using the Dynamic Time Warping (DTW) distance (Berndt and Clifford, 1994) between the contribution vector $\mathbf{v}_w = (w_1, ..., w_m)$ and those of all other candidate paths $\mathbf{v}_k = (k_1, ..., k_n)$ ($P_k \neq P_w$). DTW accommodates sequences of differing lengths and finds the optimal alignment by constructing an $m \times n$ accumulated cost matrix $\mathbf{C}$, where each element $C(i, j)$ represents the minimum cost alignment between the prefixes $\mathbf{v}_w[1..i]$ and $\mathbf{v}_k[1..j]$. The matrix is computed recursively:

$$c(i,j) = d(w_i, k_j) + \min\{c(i-1,j), c(i,j-1), c(i-1,j-1)\} \quad (2)$$

where $d(w_i, k_j) = |w_i - k_j|$ is the local distance metric (absolute difference) between individual contribution scores, and appropriate boundary conditions are applied. The overall DTW distance is given by $DTW(\mathbf{v}_w, \mathbf{v}_k) = C(m, n)$. The path P_k yielding the maximum DTW distance to $\mathbf{v}_w$ is designated as P_l:

$$P_l = \arg\max_{P_k \neq P_w} DTW(\mathbf{v}_w, \mathbf{v}_k) \quad (3)$$

This selection strategy ensures that P_l represents a maximally divergent reasoning trajectory from the preferred path P_w, based on the contribution vector analysis. Applying this process to each input problem q yields a preference triplet (q, P_w, P_l), which forms a training instance for the DPO fine-tuning stage. This procedure generates the final preference dataset $\mathcal{D} = \{(q^{(i)}, P_w^{(i)}, P_l^{(i)})\}_{i=1}^N$, comprising preferences synthesized directly from the model's self-generated exploratory outputs.

3.3 DPO Fine-tuning with Self-Generated Preferences

The final stage enhances the base model M_{ref} by fine-tuning it on the synthesized preference dataset $\mathcal{D}$. We employ DPO (Rafailov et al., 2023), a technique particularly suited for this task as it directly leverages preference pairs (q, P_w, P_l) without requiring an auxiliary reward model. Unlike standard supervised fine-tuning, which typically relies solely on positive examples, DPO's formulation allows the model to learn from the contrast between preferred (P_w) and dispreferred (P_l) reasoning paths identified in the preceding stages. This contrastive signal guides the model M_θ not only to emulate effective and concise reasoning strategies but also to penalize and avoid divergent or erroneous trajectories. The optimization objective is to maximize the relative likelihood of preferred paths over dispreferred ones, formalized by the loss function:

$$\mathcal{L}_{DPO}(M_\theta; M_{ref}) = -\mathbb{E}_{(w,l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\,\Delta r_\theta^{w,l}\right)\right] \quad (4)$$

where $\Delta r_\theta^{w,l} = r_\theta(q, P_w) - r_\theta(q, P_l)$, using the shorthand $r_\theta(q, P) = \log \frac{M_\theta(P|q)}{M_{ref}(P|q)}$ for the log-probability ratio of path P under models M_θ and M_{ref}. β represents a hyperparameter regulating the extent of deviation from M_{ref}; and σ signifies the logistic function.

The DPO fine-tuning process directly integrates the nuanced preferences captured by our path evaluation and dissimilarity metrics into the model parameters. By optimizing against the loss in Eq. (4), the model M_θ learns to favor reasoning paths exhibiting desirable structural properties like correctness and conciseness (as represented by P_w), while simultaneously learning to suppress the generation of less effective or divergent reasoning patterns characteristic of P_l. The resultant model M_θ is anticipated to exhibit improved reasoning fidelity and efficiency across diverse application domains.

4 Experimental Setup

4.1 Datasets

We evaluated our method on a curated benchmark suite covering four diverse domains: mathematics, medicine, law, and commonsense reasoning.

Mathematics Mathematical reasoning proficiency is assessed using the GPQA-Diamond benchmark (Rein et al., 2023), specifically its challenging Diamond subset, which tests graduate-level expertise. Additionally, we include two relevant tasks from MMLU (Hendrycks et al., 2021): college mathematics and formal logic.

Medicine Performance in the medical domain is evaluated using four specialized subsets from MMLU (Hendrycks et al., 2021): human aging, nutrition, professional medicine, and medical genetics. These tasks probe knowledge and reasoning specific to medical contexts.

Law Law reasoning ability is benchmarked using the hearsay task from the LegalBench (Guha et al., 2023). This task requires models to classify whether provided legal evidence as hearsay based on its definition.

Commonsense Reasoning General multi-step reasoning involving implicit assumptions and complex inference chains is evaluated using the StrategyQA dataset (Geva et al., 2021). This dataset consists of yes/no questions requiring models to formulate and justify a reasoning strategy.

4.2 Baselines

Base We employ three models derived from DeepSeek-R1 distillation as M_{ref}. These include two variants based on the Qwen architecture (1.5B and 7B parameters) and one based on the Llama architecture (8B parameters). Their performance without fine-tuning serves as a reference.

Rejection Fine-Tuning As an alternative self-sampling baseline, we implement Rejection Fine-Tuning (RFT) (Yuan et al., 2023). This method involves M_{ref} generating multiple reasoning paths, selecting only those that yield correct final answers, and then performing standard SFT exclusively on these correct paths.

Outcome-DPO To isolate the benefits of our process-oriented preference signal, we implement an outcome-oriented DPO baseline based on the strategy of Wang et al. (2024). For this baseline, preference pairs are constructed using a simple outcome-based criterion: a reasoning path with a correct final answer is chosen, while one with an incorrect answer is rejected.

Reasoning Distillation(R1-SFT) We fine-tune M_{ref} via SFT on a synthetic dataset generated by the larger Deepseek-R1 teacher model (DeepSeek-AI et al., 2025). This baseline contrasts our self-improvement method with direct knowledge transfer from a larger model.

4.3 Implementation Details

Preference Dataset Construction The preference dataset $\mathcal{D}$ for DPO is constructed by applying our self-sampling and preference labeling methodology (Sections 3.1-3.2) to domain-specific training data. For Mathematics, we use problems from the AQUA-RAT training set (Ling et al., 2017). For Medicine, problems are sourced from the Elderly Care, Genetics, Nutrition, and Medicine subsets of Health Benchmarks (Science, 2025). For Legal reasoning, a training split is manually derived from the LegalBench hearsay task (Guha et al., 2023). For Implicit Reasoning, we utilize the StrategyQA training set (Geva et al., 2021). In total, the constructed training dataset comprises 3168 examples across these four domains.

Fine-tuning Setup All models are fine-tuned using the Llama-Factory framework (Zheng et al., 2024) on 8 NVIDIA RTX 3090 GPUs. We employ Low-Rank Adaptation (LoRA) (Hu et al., 2022)

Table 1: Main experimental results (Acc) comparing our Self-DPO method with baselines across three low-parameter backbone models. Performance, including average accuracy and average rank, is evaluated on benchmarks spanning Mathematics (GPQA-Diamond and MMLU), Medicine (MMLU), Law (LegalBench), and Reasoning (StrategyQA). Best results per model backbone are marked in **bold**.

Model	Method	Math		Med	Law	Reasoning	Avg	Avg. Rank
		GPQA-D	MMLU	MMLU	LegalBench	StrategyQA		
DeepSeek-R1-Distill -Qwen-1.5B	Base	37.37	61.08	39.21	47.12	50.41	47.038	4.4
	RFT	35.35	64.15	39.72	47.71	51.96	47.778	3.4
	Outcome-DPO	37.61	62.15	38.64	49.77	50.23	47.680	3.6
	R1-SFT	**40.11**	**67.18**	**40.93**	50.47	49.92	49.722	2.0
	Self-DPO (Ours)	38.94	64.85	40.42	**52.19**	**52.69**	**49.818**	**1.6**
DeepSeek-R1-Distill -Qwen-7B	Base	45.45	86.75	56.63	29.88	56.93	55.128	4.2
	RFT	48.42	85.06	57.25	27.01	58.82	55.312	3.6
	Outcome-DPO	46.15	87.80	57.44	31.21	57.72	56.064	3.0
	R1-SFT	47.71	85.87	**58.99**	**34.48**	56.33	56.676	2.8
	Self-DPO (Ours)	**48.48**	**88.76**	58.48	33.33	**59.67**	**57.744**	**1.4**
DeepSeek-R1-Distill -Llama-8B	Base	45.45	71.35	71.98	41.95	67.54	59.654	4.4
	RFT	46.93	73.24	70.77	39.65	67.11	59.540	4.4
	Outcome-DPO	46.90	73.82	73.36	42.92	68.06	61.012	3.0
	R1-SFT	**49.01**	73.44	**74.83**	**44.55**	**68.66**	62.098	**1.4**
	Self-DPO (Ours)	48.99	**75.64**	74.79	43.12	68.45	**62.198**	1.8

with rank $r = 8$ applied to all linear layers. Common hyperparameters include a batch size of 8, a learning rate of 5×10^{-6}, and 5 training epochs. For DPO, we use $\beta = 0.1$ and a cosine learning rate schedule with 5% warmup. SFT baselines use a linear schedule.

Evaluation Model performance is evaluated using Accuracy on the benchmark datasets. For MMLU and GPQA-Diamond, accuracy reflects correct multiple-choice option selection. For StrategyQA and LegalBench-hearsay, it reflects correct binary classification. The final answer is extracted by parsing the model output following the '</think>' token in the generated reasoning path.

5 Experimental Results and Analyses

5.1 Main Results

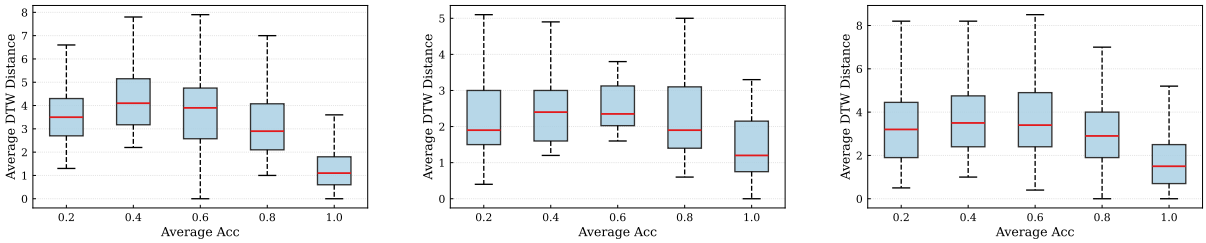
As presented in Table 1, our process-oriented Self-DPO outperforms the outcome-oriented DPO baseline across all models in both average accuracy and average rank. This result empirically validates our core hypothesis: leveraging the reasoning process itself to construct preference pairs provides a significantly more effective learning signal than relying solely on final answer correctness.

The comparison with the strong distillation baseline, R1-SFT, reveals a nuanced trend of complementary strengths. R1-SFT often excels in knowledge-intensive domains like Mathematics and Medicine, where knowledge transfer from a larger teacher is beneficial. In contrast, Self-DPO demonstrates a distinct advantage in tasks requiring more complex contextual inference, such as Law and Commonsense Reasoning. This suggests that our self-improvement approach is particularly effective at refining the intrinsic reasoning process.

The average rank metric highlights the overall robustness of our method. Self-DPO achieves the best average rank on the 1.5B and 7B models, underscoring its consistency across diverse domains. While R1-SFT also performs strongly on the 8B model, Self-DPO's performance establishes it as a powerful and resource-efficient teacher-free alternative for enhancing reasoning capabilities.

5.2 Ablation Study

Dispreferred Response Selection Strategy We investigate the impact of different strategies for selecting the dispreferred response P_l used in constructing the DPO preference dataset. Table 2 compares our proposed strategy of maximizing the DTW distance between contribution vectors against selecting the path with minimum DTW distance or a random path. The results clearly indicate that DTW-Max consistently outperforms both alternative strategies across all evaluation domains. This suggests that selecting a dispreferred example that is maximally divergent from the preferred reasoning path provides a stronger and more effective learning signal for the DPO fine-tuning process.



(a) multi-choice (b) hearsay judgment (c) reasoning judgment

Figure 3: Relationship between reasoning path similarity (average pairwise DTW distance) and answer accuracy across three QA tasks: (a) multi-choice, (b) hearsay judgment, and (c) reasoning judgment. Results show a consistent trend where higher accuracy, particularly for correct answers (Acc=1.0), corresponds to significantly lower DTW distances, indicating greater convergence in reasoning paths leading to correct solutions.

Table 2: Ablation study evaluating the impact of different components within the Self-DPO framework. Results show Acc on the 1.5B model across domains.

Method	Math	Med	Law	Reasoning
Self-DPO	**38.94**	**40.42**	**52.47**	**52.69**
Dispreferred Selection Strategy				
DTW-Min	38.38	37.81	49.42	52.25
DTW-Rand	36.36	38.99	45.97	49.49
Reasoning Contribution Evaluation				
TFIDF-Sim	35.85	39.23	52.07	50.42
LLM-Eval	38.88	39.17	52.22	50.81

Reasoning Path Evaluation Method We ablate the method used for evaluating reasoning path segments to generate the contribution vectors $\mathbf{v}_i$, which subsequently inform the DTW calculation for P_l selection. Table 2 compares our proposed NLI-based evaluation using DeBERTa-v2-XXL-MNLI against two alternative approaches: using TF-IDF cosine similarity or a separate LLM (Deepseek-V3) to score reasoning contribution. The results clearly show that the NLI-based approach consistently outperforms the alternatives across all domains, validating its effectiveness for generating contribution vectors that accurately reflect segment importance for the reasoning process.

Contribution Evaluator To justify our choice of the NLI model for contribution evaluation, we compare its performance against several alternatives, including other NLI models and a powerful LLM judge (Gemini-2.0-Flash). The results are summarized in Table 3. Our chosen model, DeBERTa-v2-XXL-MNLI, demonstrates a strong balance, identifying a high number of relevant reasoning steps. Crucially, its judgments show very high consistency with the Gemini LLM judge, achieving 94.6% agreement on 'chosen' (positive) paths, while being significantly more computationally efficient. Our tests indicate that the NLI model is over 7x faster than the LLM-based approach. This combination of high reliability and efficiency validates our choice, confirming the NLI model as a deterministic and scalable tool for our framework.

Table 3: Comparison of different models for contribution evaluator. Consistency is measured against our chosen model, DeBERTa-v2-XXL-MNLI.

Model	Steps Class.		Consistency(%)	
	Pos.	Neg.	Chosen	Rejected
RoBERTa-Large	2,704	897	35.4	39.1
DeBERTa-v2-XL-MNLI	6,190	2,173	88.7	74.5
Gemini-2.0-Flash	7,075	1,942	94.6	81.9
DeBERTa-v2-XXL-MNLI	**6,882**	**2,070**	**(Ref.)**	**(Ref.)**

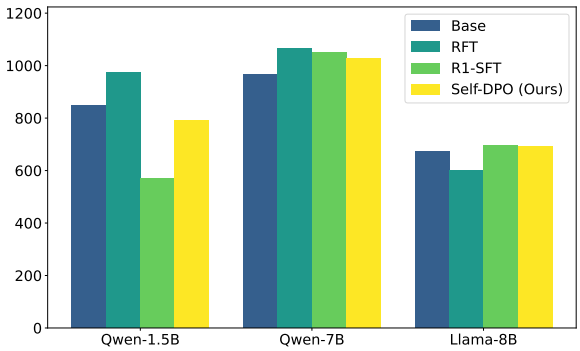


Figure 4: Token efficiency comparison (average tokens per correct response) for Self-DPO against baseline methods across three model variants.

5.3 Analysis

Effectiveness of DTW Selection Figure 3 validates our use of DTW distance as a proxy for reasoning path quality. Across all tasks, a clear

Table 4: Case study comparing reasoning paths of the base model and Self-DPO model.

Question	Base Model Reasoning (633 tokens)	Self-DPO Model Reasoning (335 tokens)
Would a goblin shark eat at Crossroads Kitchen? (False)	The goblin shark can be found in the ocean. . . . . Wait, I'm not entirely sure. Let me try. . . . . Typically, sharks are known for. . . . . To determine if the goblin shark would eat there. . . . . Alternatively, if there's no lore supporting. . . . . So considering all these factors—the presence of predator-like features and typical prey availability—the likelihood increases that such an adaptation would lead to eating fish.</think>ANSWER: True	let me identify whether the following statements: A goblin shark can eat at least one meal per day. . . . . The goblin shark is a type of fish.To determine if a goblin shark would eat at Crossroads Kitchen, we analyze each statement: Location of Crossroads Kitchen. . . . . Since the location cannot be confirmed and the eating behavior is uncertain. . . . . </think>ANSWER: False

inverse correlation emerges: higher answer accuracy corresponds to lower average DTW distance between reasoning paths. Notably, paths leading to correct answers (Acc=1.0) show a significantly lower and tighter distribution of DTW distances, indicating a convergence towards a canonical reasoning structure. This finding directly supports our core methodological choice: selecting a maximally divergent path (high DTW) as a challenging negative example for DPO.

Cross-Domain Robustness As visualized for the Qwen-7B model in Appendix A.1, Self-DPO demonstrates superior cross-domain robustness. It is the only method to achieve positive accuracy gains across all four evaluation domains. This consistency contrasts sharply with the variable performance of baselines, which may exhibit strong but specialized gains in some areas alongside significant performance regressions in others. This highlights the uniformity of our method's enhancement, a key advantage of the teacher-free approach.

Inference Token Cost We analyze the average token cost per correct response as a measure of computational efficiency (Figure 4). While no single method is consistently the most token-efficient, Self-DPO demonstrates a compelling balance between reasoning performance and computational cost. On the larger 7B and 8B models, Self-DPO achieves leading reasoning performance (Table 1) while maintaining a token cost at or below that of the strong R1-SFT baseline. This indicates that Self-DPO's performance gains are not achieved at the expense of efficiency, establishing it as a well-balanced approach.

5.4 Case Study

We analyze a commonsense reasoning question in Table 4: "Would a goblin shark eat at Crossroads Kitchen?". The base model exhibits flawed reasoning, getting sidetracked by general shark facts and expressing uncertainty ("Wait, I'm not entirely sure"), ultimately failing to recognize the fundamental incompatibility between a deep-sea creature and a restaurant, leading to an incorrect answer. In contrast, the Self-DPO model demonstrates a more structured approach, focusing on the core constraints: the goblin shark being a fish and the implicit terrestrial location of the restaurant. By correctly identifying this mismatch ("Since the location cannot be confirmed..."), it arrives at the correct conclusion. This comparison underscores Self-DPO fine-tuned model's ability to enhance logical coherence and task-relevant reasoning compared to the base model.

6 Conclusion

We propose Self-DPO, a novel approach demonstrating that low-parameter models can enhance their reasoning capabilities across diverse domains through self-sampling and preference optimization, without relying on larger teacher models. Our extensive experiments show that Self-DPO consistently improves reasoning performance, often matching or exceeding traditional distillation techniques, highlighting its efficacy and resource efficiency. This work presents a promising direction for developing capable reasoning models and motivates future research into advanced self-improvement mechanisms for complex reasoning.

Limitations

We acknowledge the limitations of this paper. Our approach relies on the base model's capacity for diverse path generation and the accuracy of the NLI evaluator. Preference construction employs heuristics (shortest correct path, max DTW divergence) as proxies for reasoning quality. The framework necessitates labeled answers, limiting unsupervised application, and introduces computational overhead through its sampling and evaluation pipeline.

References

Pranjal Aggarwal and Sean Welleck. 2025. L1: Controlling how long a reasoning model thinks with reinforcement learning. *Preprint*, arXiv:2503.04697.

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning from mistakes makes LLM better reasoner. *CoRR*, abs/2310.20689.

Donald J. Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series. In *Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop, Seattle, Washington, USA, July 1994. Technical Report WS-94-03*, pages 359–370. AAAI Press.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Tong Wang, Samuel Marks, Charbel-Raphaël Ségerie, Micah Carroll, Andi Peng, Phillip J. K. Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, and 13 others. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *Trans. Mach. Learn. Res.*, 2023.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *CoRR*, abs/2107.03374.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? A question answering benchmark with implicit reasoning strategies. *Trans. Assoc. Comput. Linguistics*, 9:346–361.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, K. Aditya, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, and 21 others. 2023. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 80 others. 2024. Openai o1 system card. *CoRR*, abs/2412.16720.

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 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Bespoke Labs. 2025. Bespoke-stratos: The unreasonable effectiveness of reasoning distillation. www.bespokelabs.ai/blog/bespoke-stratos-

the-unreasonable-effectiveness-of-reasoning-distillation. Accessed: 2025-01-22.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022. Solving quantitative reasoning problems with language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 158–167. Association for Computational Linguistics.

Yingqian Min, Zhipeng Chen, Jinhao Jiang, Jie Chen, Jia Deng, Yiwen Hu, Yiru Tang, Jiapeng Wang, Xiaoxue Cheng, Huatong Song, Wayne Xin Zhao, Zheng Liu, Zhongyuan Wang, and Ji-Rong Wen. 2024. Imitate, explore, and self-improve: A reproduction report on slow-thinking reasoning systems. *CoRR*, abs/2412.09413.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel J. Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *CoRR*, abs/2501.19393.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. GPQA: A graduate-level google-proof q&a benchmark. *CoRR*, abs/2311.12022.

Yesil Science. 2025. Llm health benchmarks dataset.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Andrew Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Agüera y Arcas, and 12 others. 2023. Towards expert-level medical question answering with large language models. *CoRR*, abs/2305.09617.

NovaSky Team. 2025. Sky-t1: Train your own o1 preview model within $450. https://novasky-ai.github.io/posts/sky-t1. Accessed: 2025-01-09.

Tianduo Wang, Shichen Li, and Wei Lu. 2024. Self-training with direct preference optimization improves chain-of-thought reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11917–11928, Bangkok, Thailand. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Yue Wang, Qiuzhi Liu, Jiahao Xu, Tian Liang, Xingyu Chen, Zhiwei He, Linfeng Song, Dian Yu, Juntao Li, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. 2025. Thoughts are all over the place: On the underthinking of o1-like llms. *CoRR*, abs/2501.18585.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *CoRR*, abs/2308.01825.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. Star: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

A Additional Experimental Details

A.1 Cross-Domain Robustness Analysis

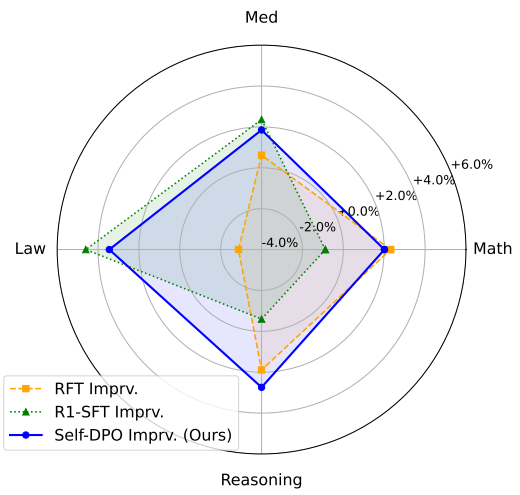


Figure 5: Acc improvement over the 7B model for baselines and Self-DPO. Self-DPO demonstrates the most consistent positive gains across four domains.

A.2 Dataset Statistics

Table 5 presents the distribution of training and test examples across domains.

Table 5: Training and test data distribution across domains.

Domain	Train	Test
Mathematics	927	459
Medicine	813	1019
Law	156	174
Implicit Reasoning	1272	687
Total	3168	2339

A.3 Hyperparameters

Table 6 shows the key hyperparameters used in our experiments.

A.4 Implementation Details

All experiments were conducted using PyTorch 2.0.1 and Transformers 4.31.0 on 8 NVIDIA RTX 3090 GPUs. For DPO training, we used a cosine learning rate schedule with 5% warmup. The self-sampling process generated 5 candidate reasoning paths per instance with temperature 1.0 and top-p sampling (p=0.98). Evaluation was conducted with a sampling temperature of 0.6 and top-p sampling (p=0.95).

Table 6: Hyperparameter settings for model fine-tuning.

Model	Hyperparameter	Value
DeepSeek-R1-Distill Qwen-1.5B	Learning Rate (LR)	5e-6
	Batch Size	8
	Epochs	3
	LoRA r	8
	DPO β	0.1
	Warm-up Steps	0.05
	Cut-off Length	3000
DeepSeek-R1-Distill Qwen-7B	Learning Rate (LR)	5e-6
	Batch Size	8
	Epochs	5
	LoRA r	8
	DPO β	0.1
	Warm-up Steps	0.05
	Cut-off Length	2048
DeepSeek-R1-Distill Llama-8B	Learning Rate (LR)	5e-6
	Batch Size	8
	Epochs	5
	LoRA r	16
	DPO β	0.1
	Warm-up Steps	0.05
	Cut-off Length	2048

A.5 Prompt Templates

The following templates are used for different task types. Each prompt instructs the model to first generate explicit reasoning steps (enclosed in <think>...</think>), followed by a concise final answer in the required format. This design encourages interpretable and robust model outputs across diverse domains.

- Multiple Choice Question (MCQ)

```
User:
Please solve this multiple choice question.
Question: {question}
Options: {choices_str}
Provide your answer in the format
'ANSWER: X' where X is A, B, C, or D.
```

- Law Hearsay Classification

```
User:
Here is a piece of legal evidence.
evidence: {text}
You need to classify if this piece of
evidence qualifies as hearsay.
Give your answer in format: 'ANSWER:Yes/No'.
```

- Strategy (True/False) Judgment

```
User:
Question: {question}
You need to carefully think through,
analyze, and solve this question.
Provide your answer in the format
'ANSWER: True/False'.
```