Aligning Large Language Models via Fully Self-Synthetic Data

Anonymous ACL submission

Abstract

Traditional reinforcement learning from human feedback (RLHF) for large language models (LLMs) relies on expensive human-annotated datasets, while Reinforcement Learning from AI Feedback (RLAIF) also incurs significant costs, requiring the collection of diverse prompts and corresponding responses, often necessitating external reward models or proprietary models like GPT-4 to annotate preference pairs. In this work, we introduce Self-Alignment Optimization (SAO), a fully self-synthetic framework for LLM alignment, where all training data, including prompts (*i.e.*, user queries), responses, and preferences, are generated by the model itself. Specifically, SAO first instructs the LLM to engage in persona role-play and generate diverse prompts and responses, which are then self-evaluated for preference optimization. Extensive experiments demonstrate that SAO effectively enhances the model's chat capabilities on standard benchmarks like AlpacaEval 2.0, while maintaining strong performance on downstream objective tasks (*e.g.*, question-answering, math reasoning). Our work provides a practical solution for self-improvement in aligning LLMs, and the code for reproducing our results is available at: `https://anonymous.4open.science/r/SAO-CODE`.

1 Introduction

Large language models (LLMs) have revolutionized the field of natural language processing (NLP), demonstrating remarkable capabilities in tasks such as mathematical reasoning, code generation, and dialogue generation (Cobbe et al., 2021; Wei et al., 2022; Bubeck et al., 2023; Chen et al., 2024b; Yin et al., 2025a,b).

A key advancement in LLMs is their alignment with human preference to create more helpful and reliable assistants (Mishra et al., 2021; Victor et al., 2022; Chung et al., 2022; Thoppilan et al., 2022). Common approaches include supervised fine-tuning (SFT) (Ouyang et al., 2022; Tunstall et al., 2023), based on human-demonstration pairs, and reinforcement learning from human feedback (RLHF) (Christiano et al., 2017; Ziegler et al., 2019; Stiennon et al., 2020; Bai et al., 2022), which leverages signals from human preferences.

However, collecting demonstrations and preference labels is a expensive, time-consuming process, involving substantial human annotating efforts. To address this challenge, reinforcement learning from AI feedback (RLAIF) has been gaining attention, where a reward model is trained using AI-labeled preference data or directly by LLMs through specially designed prompts to annotate preference pairs (Lee et al., 2024). However, RLAIF remains costly, typically requiring strong, proprietary models (*e.g.*, GPT-4) or specialized reward model designs (Jiang et al., 2023; Wang et al., 2024) to work effectively. Moreover, these approaches often incur additional overhead from data filtering to obtain the final clean dataset (Xu et al., 2024b).

Recent self-improvement approaches for LLM alignment, such as Self-Rewarding (Yuan et al., 2024) and SPPO (Wu et al., 2024), have demonstrated promising results. However, these approaches are not fully self-synthetic and still require external intervention: Self-Rewarding relies on human-labeled data as few-shot templates to generate synthetic data and then mixes both human and synthetic data for model optimization, while SPPO requires prompts from existing datasets and external reward models for preference labeling. Such dependencies limit their scalability and accessibility.

In this paper, we propose a fully self-synthetic method for LLM alignment, termed Self-Alignment Optimization (SAO), eliminating the need for costly training data collection and annotation. Drawing inspiration from the compress-and-decompress approach to world knowledge from a persona perspective (Tseng et al., 2024; Ge et al., 2024b,a) and the success of self-improvement mechanisms (Samuel, 2000; Chen et al., 2024b), SAO begins with (1) instructing the LLM to engage in persona role-play and generate diverse prompts (*i.e.*, user queries). (2) The LLM then generates paired responses for each prompt, which will be ranked through self-judgment. (3) Lastly, preference optimization is performed to fine-tune the model based on the ranked responses (Amini et al., 2024; Meng et al., 2024).

Across multiple standard chat benchmarks for LLM alignment, SAO demonstrates substantial performance improvements compared to the backbone model. For example, on AlpacaEval 2.0, SAO boosts the Length-Controlled Win Rate (LC) and Win Rate (WR) of Gemma-2-9B-it by 18.1% and 27.9%, respectively. On

MT-Bench, SAO improves the average score of Gemma-2-9B-it from 8.41 to 8.66 points. On Arena-Hard, SAO also boosts the model's WR from 40.8% to 54.3%.

In addition, we find that SAO maintains or even enhances the backbone LLM's performance on downstream objective NLP tasks, though the model was not trained using domain-specific data. In the evaluation of the Open LLM Leaderboard, SAO enables Gemma-2-9B-it to achieve an average score of 74.41 across all benchmarks, surpassing its baseline score of 74.28, without any further training. We highlight that this improvement is significant, as models trained on manually crafted datasets often enhance alignment at the cost of compromised general capability (Meng et al., 2024). For instance, Gemma-2-9B-it trained on UltraFeedback (Cui et al., 2024) achieves only a score of 70.38 using the same training algorithm as SAO. These results highlight SAO's effectiveness in synthesizing high-quality data that enhances a model's subject-specific capabilities while preserving its downstream performance in a more data-efficient manner.

To better understand the empirical results, we further conduct an in-depth analysis with the Gemma-2-9B-it model. Specifically, we compare the LLM's self-synthesized prompts with manually crafted prompts from UltraFeedback. Interestingly, we find that synthetic prompts lead to significantly better performance than those from the Ultrafeedback benchmark when followed by the same self-improvement process in SAO (*i.e.*, Step 2 and 3), with a 16.46% improvement in WR. This emphasizes the importance of prompt construction for LLM alignment. We also find that SAO's effectiveness largely stems from the surprisingly strong self-judging ability of the backbone model, which even surpasses the external reward model ArmoRM-Llama3-8B-v0.1 (Wang et al., 2024) and even GPT4-level feedback when used within the SAO framework. This is further confirmed by the findings in Section 4.4, which show that the model is robust to various judgment criteria designs, indicating that the model's strong evaluation ability, thereby contributing to the effectiveness of SAO.

Our contributions are summarized as follows: (1) We propose a self-alignment optimization framework, which aligns the model with its own generated prompts and feedback as preferences, eliminating the need for expensive data collection and annotation. (2) On multiple standard chat benchmarks, the LLMs trained with SAO achieve competitive results compared to strong proprietary models like GPT-4 and state-of-the-art open-weight models trained on expensively annotated external datasets. (3) In contrast to previous resource-intensive approaches, SAO enhances LLM alignment while maintaining strong downstream objective task performance, providing a practical and scalable solution for improving a model's chat abilities in a more balanced and effective manner. (4) We demonstrate that the SAO framework exhibits long-term scaling properties in both synthetic dataset size and iterative optimization, providing a sustainable path for continued model improvement.

2 Related Work

2.1 Synthetic Data for LLMs

In the context of fine-tuning LLMs, human-crafted data has proven remarkably effective, significantly enhancing performance on tasks like code generation (Roziere et al., 2023; Yang et al., 2023) and mathematical reasoning (Yuan et al., 2023; Luo et al., 2023; Zhu et al., 2025). While human-generated data is typically of high quality, acquiring sufficient amounts is often prohibitively expensive. Consequently, the use of synthetic data has gained popularity as a cost-effective substitute for human data. This approach primarily leverages advanced LLMs, such as the GPT series (Radford et al., 2019; Brown et al., 2020; OpenAI, 2023), to generate high-quality data (Josifoski et al., 2023; Taori et al., 2023; Chiang et al., 2023; Li et al., 2023b). Recent studies have also emphasized the benefits of using LLMs' rephrasing capabilities to improve prompt responses (Deng et al., 2023; Prasad et al., 2023), as well as augmenting synthetic data for more effective fine-tuning (Yu et al., 2023; Liu et al., 2023). Unlike prior research, which typically relies on more advanced models for generating synthetic training data during, our approach directly generates synthetic data from the target model itself, streamlining the process and reducing dependency on external resources.

2.2 Persona Roleplay

Persona roleplay in LLMs can be viewed as a compress-and-decompress mechanism for world knowledge. In this framework, world knowledge is compressed into distributed representations, which are then decompressed by various personas to generate texts based on their unique knowledge (Delétang et al., 2024; Ge et al., 2024b). Recently, Ge et al. (2024a) introduced the *Persona Hub*, a system automatically constructed from vast web data. This hub enables the exploration of most perspectives embedded within LLMs, facilitating the creation of diverse synthetic data at scale, without the need for seed corpora. In this work, we directly leverage personas from this hub, which can be seamlessly integrated into customized data synthesis prompts, leveraging the LLM's strong role-playing capabilities and offering exceptional versatility.

2.3 LLM-as-a-Judge

Using LLM-as-a-Judge prompting to evaluate language models has become a standard approach (Dubois et al., 2024; Li et al., 2023a; Fernandes et al., 2023; Bai et al., 2023; Saha et al., 2023; Chen et al., 2025; Wei et al., 2025a). This technique is not only employed for evaluation but also for training reward models and curating data, as mentioned in prior works (Lee et al., 2023; Chen et al., 2024a; Li et al., 2024b; Wei et al., 2025b). While some studies focus on creating training data to enhance an LLM's performance as a judge (Kim et al., 2023; Yuan et al., 2024), our approach uniquely integrates this

Algorithm 1 Self-Alignment Optimization (SAO)

```
Require: Base model $\mathcal{M}_{\theta_0}$, number of personas $n$, preference optimization algorithm $\mathcal{A}$
Ensure: Optimized model $\mathcal{M}_{\theta_1}$
    Initialize personas templates $\{r_1, r_2, \ldots, r_n\}$
    Initialize dataset $\mathcal{D} \leftarrow \emptyset$
    for $i = 1, 2, \ldots, n$ do
        $x^i_{\text{prompt}} \leftarrow \mathcal{M}_{\theta_0}(r_i)$                                        § 3.1
        $y_1, y_2 \leftarrow \mathcal{M}_{\theta_0}(\cdot \mid x^i_{\text{prompt}})$                      § 3.2
        $(y_{\text{w}}, y_{\text{l}}) \leftarrow \mathcal{R}_{\theta_0}(y_1, y_2 \mid x^i_{\text{prompt}}, x_{\text{rank}})$     § 3.3
        $\mathcal{D} \leftarrow \mathcal{D} \cup \{(x^i_{\text{prompt}}, y_{\text{w}}, y_{\text{l}})\}$                 § 3.4
    end for
    Optimize: $\theta_1 \leftarrow \arg\min_\theta \mathcal{L}_{\mathcal{A}}(\mathcal{M}_\theta)$
    Return $\mathcal{M}_{\theta_1}$
```

judging capability with general instruction-following skills, setting it apart from existing methods.

3 Self-Alignment Optimization

We present the overview of Self-Alignment Optimization (SAO) in Algorithm 1 and elaborate on its design in the following sections.

3.1 Diverse Prompt Generation

To facilitate a comprehensive range of training scenarios, we utilize an LLM denoted as $\mathcal{M}$, parameterized by θ, for the generation of diverse prompts tailored to specific persona roles, as depicted in Figure 5 (top). Given a set of role templates $\mathcal{R} = \{r_i\}_{i=1}^n$, we derive a unique prompt for each persona:

$$x^i_{\text{prompt}} = \mathcal{M}_\theta(r_i) \tag{1}$$

In this context, x^i_{prompt} represents the prompt generated for the i-th persona r_i. To ensure the diversity of generated prompts, we impose a constraint such that each persona can generate only a single question. The persona resources are randomly sampled from PersonaHub (Ge et al., 2024a).

3.2 Pairwise Response Generation

For each generated prompt, we create a pair of responses to enable comparative evaluation. Let $\mathcal{X}$ be the space of prompts and $\mathcal{Y}$ the space of responses. For each prompt $x^i_{\text{prompt}} \in \mathcal{X}$, we generate two responses $y_1, y_2 \in \mathcal{Y}$ using the $\mathcal{M}_\theta$:

$$y_1, y_2 \sim \mathcal{M}_\theta(\cdot \mid x^i_{\text{prompt}}) \tag{2}$$

Generating additional responses could potentially yield better performance but would increase computational costs and evaluation time. We leave this exploration for future work.

3.3 Self-Judgment

To assess the quality of generated responses, we implement a self-judgment mechanism. This process entails the LLM evaluating its own outputs, thereby simulating human preferences. As illustrated in Figure 5 (bottom), we query the LLM with a ranking prompt x_{rank} to compare the responses y_1 and y_2 based on their relevance and quality relative to x^i_{prompt}:

$$(y_{\text{w}}, y_{\text{l}}) = \mathcal{R}_\theta(y_1, y_2 \mid x^i_{\text{prompt}}, x_{\text{rank}}) \tag{3}$$

Here, y_{w} and y_{l} represent the superior and inferior responses, respectively. The function $\mathcal{R}_\theta$ encapsulates the LLM's decision-making process in ranking the responses.

3.4 Dataset Construction

We construct a synthetic dataset $\mathcal{D}$ by aggregating the generated prompts and ranked responses for each persona:

$$\mathcal{D} = \{(x^i_{\text{prompt}}, y^i_{\text{win}}, y^i_{\text{lose}})\}_{i=1}^n \tag{4}$$

where n is the total number of personas. This dataset forms the cornerstone of our preference optimization process, allowing the model to learn from its own generated and ranked responses across diverse personas.

3.5 Preference Optimization

Recent advancements in preference optimization have demonstrated significant potential in aligning LLMs with human preferences. Techniques such as Direct Preference Optimization (DPO) (Rafailov et al., 2023) and Simple Preference Optimization (SimPO) (Meng et al., 2024) have gained prominence due to their efficacy in fine-tuning LLMs to better reflect human preferences. In this study, we employ SimPO due to its suitability for our dataset, which frequently contains longer responses. Its length normalization technique effectively captures nuanced information at the token level, making it particularly well-suited to our requirements and we provide a more detailed analysis and comparison of these methods in Section 4.4.

SimPO introduces a length-normalized reward formulation that aligns with the likelihood metric guiding generation with a scaling constant β:

$$\begin{aligned} r(x, y) &= \frac{\beta}{|y|} \log \mathcal{M}_\theta(y \mid x) \\ &= \frac{\beta}{|y|} \sum_{i=1}^{|y|} \log \mathcal{M}_\theta(y_i \mid x, y_{<i}) \end{aligned} \tag{5}$$

Additionally, it incorporates a target reward margin $\gamma > 0$ to ensure a minimum difference between the rewards of winning and losing responses:

$$p(y_w \succ y_l \mid x) = \sigma\left(r(x, y_w) - r(x, y_l) - \gamma\right) \tag{6}$$

The overall objective is then formulated as:

$$\begin{aligned} \mathcal{L}(\mathcal{M}_\theta) = -\,\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \Bigg[\log \sigma \Bigg(\frac{\beta}{|y_w|} \log \mathcal{M}_\theta(y_w | x) \\ - \frac{\beta}{|y_l|} \log \mathcal{M}_\theta(y_l | x) - \gamma \Bigg) \Bigg] \end{aligned} \tag{7}$$

It guides the optimization process, enabling the model to learn from self-generated preferences and improve the alignment with desired outcomes.

4 Experiments

4.1 Experiment Setting

In our experiments, we use the Gemma-9B-it model as the base and apply a similar fine-tuning process to Llama-3-8B-Instruct. To ensure diverse prompts and responses, we set the temperature to 0.6 and utilize VLLM for accelerated generation. For preference optimization, we incorporate Flash Attention 2 and bfloat16 precision, with hyperparameters set to $\beta = 10$ and $\gamma = 3$. Additionally, we employ DeepSpeed with ZeRO-3 optimization for effective memory management and scalability. All experiments are conducted over a single epoch with a global batch size of 128 across four A100 GPUs. The learning rate is set to 1×10^{-6}, following a cosine decay scheduler with a warmup ratio of 0.1. We use a synthetic dataset of 60k samples as the default setting for self-alignment optimization.

4.2 Evaluation Metrics and Baselines

Evaluation Metrics. Our experimental evaluation employs a comprehensive set of metrics to assess model performance across various dimensions. For subjective benchmarks, we primarily focus on AlpacaEval 2 (Li et al., 2023a), an LLM-based automatic evaluation benchmark utilizing prompts from AlpacaFarm (Dubois et al., 2024). In this benchmark, model responses and GPT-4-Turbo generated reference responses are evaluated by GPT-4-Turbo or Qwen2-72B-Instruct annotators. We also incorporate GPT-4 to evaluate two additional subjective benchmarks: Arena-Hard (Li et al., 2024a), an automatic evaluation tool featuring 500 challenging user queries, and MT-Bench (Zheng et al., 2023), a set of 80 high-quality multi-turn open-ended questions covering topics such as writing, role-playing, math, and coding. For objective benchmarks, we utilize the Open LLM Leaderboard (Beeching et al., 2023), which comprises six datasets focusing on various aspects of language model evaluation, including math problem-solving, language understanding, human falsehood mimicking, and reasoning. We adhere to the standard evaluation process, using in-context learning to prompt the models and compute the average score across these six datasets to measure performance comprehensively.

Baselines. In our comparisons, we include a diverse set of baselines. These encompass vanilla models such as GPT-4o-05-13, Claude-3.5-Sonnet, and GPT4-Turbo-04-09. Additionally, we evaluate models trained on external labeled datasets, like Llama-3-Instruct-8B-SimPO (Meng et al., 2024), which has been fine-tuned using the Ultrafeedback dataset (Cui et al., 2024) for preference optimization.We also consider Self-Rewarding-70B-Iter3 (Yuan et al., 2024), which is trained using a mixture of external labeled datasets and synthetic data. Additionally, we compare against the recently developed data-distillation baseline, Llama-3-8B-Magpie-SFT-v0.1 (Xu et al., 2024b), which was originally trained on synthetic SFT pair data generated by the instruction-tuned model itself to improve its base model alignment capability. Furthermore, we examine Gemma-2-9B-SPPO-Iter3 (Wu et al., 2024), which generates responses based on Ultrafeedback prompts and utilizes preference pairs labeled by an external reward model.

4.3 Main Results

Performance on AlpacaEval 2.0. Our SAO-tuned models demonstrate substantial performance improvements on AlpacaEval 2.0 when evaluated by both GPT-4-Turbo-1106 and Qwen2-72B-Instruct. When assessed by GPT-4-Turbo-1106, Gemma-2-9B-it-SAO achieves a LC of 69.2% and a WR of 66.0%, representing increases of 18.1% and 27.9% respectively over the baseline Gemma-2-9B-it (51.1% LC, 38.1% WR). This performance surpasses all vanilla models, including the top-performing GPT-4o (05-13) at 57.5% LC and 51.3% WR. Moreover, Gemma-2-9B-it-SAO competes closely with models trained on external datasets, approaching the performance of Gemma-2-9B-it-SimPO (72.4% LC, 65.9% WR). Similarly, Llama-3-Instruct-8B-SAO exhibits significant improvements, reaching 33.3% LC and 39.0% WR, increases of 10.4% and 16.4% respectively over the baseline Llama-3-8B-Instruct (22.9% LC, 22.6% WR). When evaluated by Qwen2-72B-Instruct, Gemma-2-9B-it-SAO continues to excel, achieving 76.0% LC and 71.6% WR. These represent substantial improvements of 19.5% and 32.3% over the baseline Gemma-2-9B-it (56.5% LC, 39.3% WR) and even outperform models trained on external datasets, such as Gemma-2-9B-it-SimPO (74.5% LC, 65.5% WR). Llama-3-Instruct-8B-SAO also demonstrates significant improvement when evaluated by Qwen2-72B-Instruct, reaching 42.3% LC and 49.1% WR, increases of 12.9% and 19.9% over Llama-3-8B-Instruct (29.4% LC, 29.2% WR).

These results underscore the efficacy of our SAO method in enhancing model performance across different base models and evaluation metrics. Notably, SAO achieves these improvements without relying on external labeled datasets, highlighting its potential for efficient and scalable model enhancement.

Performance on MT-Bench and Arena-Hard. Our evaluation extended to two other mainstream subjective benchmarks, MT-Bench and Arena-Hard, yielding compelling results that underscore the efficacy of SAO fine-tuning. As shown in Table 2, on the MT-Bench benchmark, Gemma-2-9B-it-SAO achieved an average score of 8.66, surpassing the backbone model Gemma-2-9B-it by 0.25 points. In contrast, LLaMA-3-8B-Instruct-SAO maintained its average score of 7.84. The Arena-Hard benchmark revealed even more substantial performance gains, with Gemma-2-9B-it experiencing a remarkable increase in WR from 40.8% to 54.3% after SAO tuning,

Table 1: Comparative analysis of various baseline models and our proposed SAO method using AlpacaEval 2.0. The table presents Length-Controlled Win Rate (LC), Win Rate (WR), and Standard Deviation (STD) for each model, evaluated against GPT-4-Turbo-1106 and Qwen2-72B-Instruct. We also evaluate downstream performance generalization in Section 4.3 and provide additional judge analysis in Section 4.4.

Model	AlpacaEval 2.0					
	GPT-4-Turbo-1106			Qwen2-72B-Instruct		
	LC (%)	WR (%)	STD	LC (%)	WR (%)	STD
Off-the-shelf Models						
Llama-3-8B-Instruct	22.9	22.6	1.3	29.4	29.2	1.6
Yi-34B-Chat	27.2	29.7	1.3	33.3	37.0	1.7
GPT-4-Turbo-04-09	55.0	46.1	1.5	49.0	39.1	1.7
Gemma-2-9B-it	51.1	38.1	-	56.5	39.3	1.7
Claude-3.5-Sonnet	52.4	40.6	1.5	**56.8**	40.5	1.7
GPT-4o-05-13	**57.5**	**51.3**	1.5	51.8	**44.7**	1.8
Models Trained Using External Generated/Labeled Dataset						
Self-Rewarding-70B-Iter3 (Yuan et al., 2024)	-	20.4	-	-	-	-
Llama-3-Instruct-8B-SimPO (Meng et al., 2024)	53.7	47.5	-	54.2	45.9	1.8
Gemma-2-9B-SPPO-Iter3 (Wu et al., 2024)	53.3	47.8	-	-	-	-
Llama-3-8B-Magpie-SFT-v0.1 (Xu et al., 2024b)	24.2	25.2	-	26.2	29.2	1.6
Gemma-2-9B-it-SimPO (Meng et al., 2024)	**72.4**	**65.9**	1.4	**74.5**	**65.5**	1.7
Models Trained Only Using Self-Synthetic Dataset						
Llama-3-Instruct-8B-SAO (Ours)	33.3 (+10.4)	39.0 (+16.4)	1.4	42.3 (+12.9)	49.1 (+19.9)	1.8
Gemma-2-9B-it-SAO (Ours)	**69.2** (+18.1)	**66.0** (+27.9)	1.4	**76.0** (+19.5)	**71.6** (+32.3)	1.6

marking a 13.5 percentage point improvement. Similarly, LLaMA-3-Instruct-8B's WR rose from 20.6% to 28.1%, reflecting a 7.5 percentage point increase.

These significant advancements across the MT-Bench and Arena-Hard benchmarks highlight the potential of SAO tuning in enhancing model performance on multi-turn, open-ended questions and other diverse and challenging tasks.

Downstream task performance. Since the SAO framework typically generates instruction data pairs without ground-truth-style data (e.g., math and code), it is essential to assess its influence on downstream objective task performance. As detailed in Table 3, we conducted a comprehensive evaluation across diverse tasks using the Open LLM Leaderboard benchmarks. The results demonstrate that SAO-tuned models generally maintain or slightly improve their capabilities compared to their baseline counterparts.

For the Gemma-2-9B series, the SAO-tuned version achieves an average score of 74.41 across all benchmarks, marginally surpassing the baseline Gemma-2-9B-it (74.28). Notably, Gemma-2-9B-it-SAO shows improvements in ARC (+0.42), TruthfulQA (+2.61), and HellaSwag (+0.80), while maintaining comparable performance in other tasks. Similarly, Llama-3-8B-Instruct-SAO (68.20) slightly outperforms its baseline (68.19), with notable enhancements in ARC (+1.62) and HellaSwag (+0.18).

Interestingly, models optimized with externally labeled datasets like Ultrafeedback, such as Gemma-2-9B-it-SimPO and Llama-3-8B-Instruct-SimPO, while achieving impressive results on alignment tasks, show a decrease in overall performance across these general benchmarks. Gemma-2-9B-it-SimPO's average score (70.38) is significantly lower than both the baseline and SAO-tuned versions, with notable declines in Winograd (-4.34) and HellaSwag (-15.08). Llama-3-8B-Instruct-SimPO, despite improvements in specific areas like TruthfulQA (+12.16), also shows a slight overall decrease (67.73) compared to its baseline, primarily due to a substantial drop in GSM8K performance (-20.01).

Table 2: Performance on MT-Bench and Arena-Hard benchmarks. MT-Bench shows the lowest agreement with Chatbot Arena compared to AlpacaEval 2.0 and Arena-Hard (Meng et al., 2024), but is included due to its wide adoption in prior work.

Model	**MT-Bench**	**Arena-Hard**
Llama-3-8B-instruct	7.84	20.6
+ SAO	7.84 (+0.00)	28.1 (+7.5)
Gemma-2-9B-it	8.41	40.8
+ SAO	8.66 (+0.25)	54.3 (+13.5)

We hypothesize that this performance discrepancy stems from the nature of externally annotated datasets, which may not align perfectly with the current capabilities of these language models. While such datasets can yield improvements in specific alignment tasks, they may inadvertently compromise the model's general abilities, which is also referred to as the "alignment tax" issue(Lin et al., 2024). In contrast, the SAO method, which utilizes self-generated subjective instruction data, appears to more accurately represent and enhance the model's intrinsic capabilities, leading to consistent performance across a wide range of tasks without significant trade-offs.

Table 3: Performance comparison of models on downstream NLP benchmarks from the Open LLM Leaderboard. The values in parentheses indicate the number of few-shot examples (shots). We provide additional results on Math500 and a detailed logits distribution shift analysis in Appendix C and Appendix D.

Model	ARC (25)	TruthfulQA (0)	Winograd (5)	GSM8K (5)	HellaSwag (10)	MMLU (5)	Average
Gemma-2-9B-it-SAO	71.50	62.76	77.35	80.29	82.53	72.02	**74.41**
Gemma-2-9B-it-SimPO	69.11	59.00	73.72	81.96	66.65	71.82	70.38
Gemma-2-9B-it	71.08	60.15	78.06	82.34	81.73	72.30	74.28
Llama-3-8B-Instruct-SAO	63.57	49.58	74.66	76.72	78.96	65.72	**68.20**
Llama-3-8B-Instruct-SimPO	66.64	63.86	74.74	55.65	78.97	66.51	67.73
Llama-3-8B-Instruct	61.95	51.70	75.30	75.66	78.78	65.72	68.19

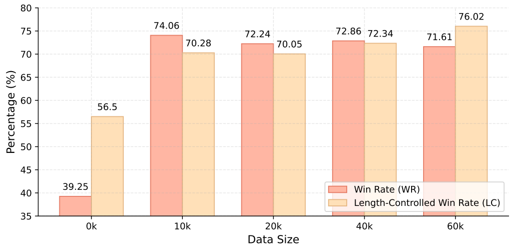


Figure 1: Impact of dataset size on model performance.

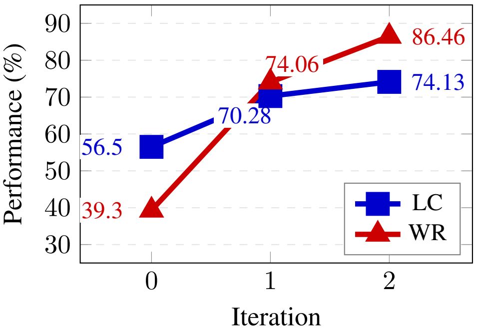


Figure 2: Impact of iterative optimization.

4.4 Analysis and Ablations

In this subsection, we primarily utilize Gemma-2-9B-it-SAO to investigate various factors influencing model performance on AlpacaEval 2.0, leveraging Qwen2-72B-Instruct as the evaluation judge. Our observations indicate that Qwen2-72B-Instruct achieves evaluation results comparable to GPT-4, making it a cost-effective alternative. This evaluation includes analyzing the scaling law of the synthetic dataset size. Additionally, for other factors such as optimization algorithms, persona prompt generation, and judging methods, we observed that even with a 10k synthetic dataset, the SAO-tuned model demonstrated promising improvements. Therefore, we focused on these aspects using the 10k synthetic dataset to enhance the efficiency and cost-effectiveness of the evaluation process.

Scaling law of synthetic dataset size. As illustrated in Figure 1, the performance of Gemma-2-9B-it-SAO improved significantly with an increase in synthetic dataset size. The WR rose from 39.25% for the vanilla model to 74.06% with a 10k dataset, stabilizing around 72% for larger datasets. Similarly, the LC metric reached 76.02% with a 60k dataset. Interestingly, we observed that even a small amount of self-alignment data can significantly enhance model alignment performance. However, the model's performance appears to saturate after 10k examples, suggesting diminishing returns with further increases in dataset size.

We hypothesize that more complex prompts generated in the post-SAO stage could further unlock the potential of the model, enabling additional performance gains. Future work could explore innovative prompt generation techniques or incorporate domain-specific nuances to overcome this saturation and maximize the model's capabilities.

Iterative self-optimization. We also explore iterative optimization, where the improved model generates new preference data to further train itself. This shows promising scalability, and we consider future work on balancing dataset size and iteration count for optimal cost-effectiveness. As shown in Figure 2, both LC and WR metrics demonstrate consistent improvement across iterations. Notably, the WR metric exhibits substantial gains, jumping from 39.30% in the vanilla model to 74.06% after the first iteration, and further increasing to 86.46% in the second iteration. The LC metric also shows steady growth from 56.50% to 74.13%. These results demonstrate the effectiveness of iterative self-optimization in enhancing model alignment capabilities.

Table 4: Experiments with different preference optimization algorithms using same generated dataset.

Algorithms	Win Rate (%)
None	39.25
DPO	49.81
ORPO	67.33
SimPO (default)	74.04

Different optimization algorithms. To investigate the influence of different optimization algorithms, we compared three mainstream approaches: DPO (Xu et al., 2024a), ORPO (Hong et al., 2024), and SimPO (Meng et al., 2024). Table 4 illustrates the performance of these algorithms. Starting from the baseline Gemma-

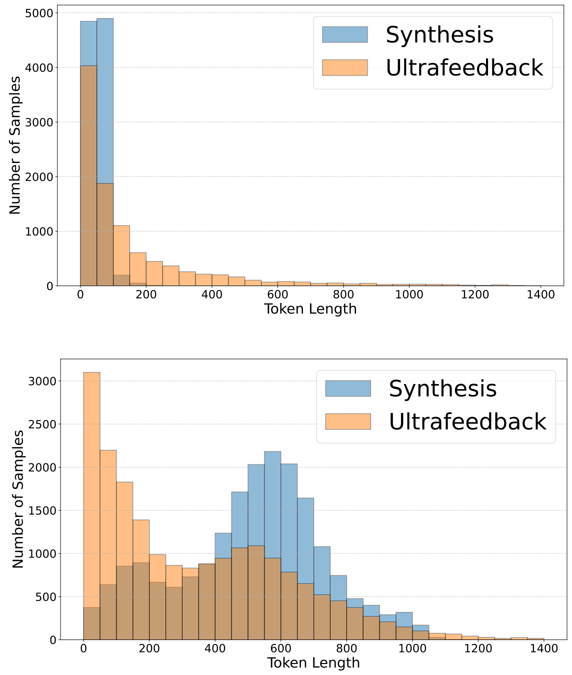


Figure 3: Distribution of prompt and response lengths.

2-9B-it model (39.25% WR), we observed progressive improvements: DPO raised the WR to 49.81%, ORPO increased it further to 67.33%, and SimPO achieved the highest WR of 74.04%.

The superior performance of SimPO may be attributed to the characteristics of our generated dataset, as shown in Figures 3. Compared to externally labeled datasets, our synthetic dataset tends to generate shorter prompts (less than 200 words) and longer responses (mainly in the 400-800 words range), compared to the existing Ultrafeedback dataset. This characteristic makes SimPO's length normalization formula particularly effective in this context. Examples of the synthetic dataset are provided in Table 13 in the Appendix.

Table 5: Experiment with different prompt sources.

Source	Win Rate (%)	Repetition (%)
UltraFeedback	55.84	0.0
Random (Ours)	62.50	45.7
Persona (Ours)	**72.30**	0.7

Different prompt sources. A key component of our method is persona-based prompt generation, which enhances the diversity of the generated prompts. Table 5 illustrates the effect of using different prompt sources. When the synthetic dataset was generated with persona role-play, the model achieved a win rate (WR) of 72.30% after training with it, along with a significantly lower prompt repetition rate of 0.7% (only occurred when the model rejected generating prompts for certain personas). In contrast, when no persona role-play was used, the WR dropped to 62.50%, and the repetition rate increased to 45.7%.

Table 6: Comprehensive evaluation of feedback sources.

Feedback Source	LC (%)	WR (%)
Gemma-2-9B-it-SAO		
Random feedback	–	8.82
Length-Based feedback	–	3.29
ArmoRM feedback	–	41.43
GPT-4o feedback	63.37	52.80
GPT-4o-mini feedback	**71.54**	68.51
Self-feedback(Ours)	70.28	**74.04**
w/o Criterion 1	70.15	73.66
w/o Criterion 2	70.22	73.98
w/o Criterion 3	69.87	73.14
w/o Criterion 4	69.43	72.67

These results emphasize the crucial role of persona-based prompt generation in improving model performance and reducing repetition. Notably, even in the presence of repetitive prompts, the model's performance remains strong. We attribute this robustness to the SAO, which allows the model to continue self-improving even when facing redundant prompts. The model remains capable of generating diverse responses, thanks to the next-token prediction mechanism and the temperature settings applied during generation. For comparison, when the same SAO process was applied using randomly sampled prompts from UltraFeedback, the WR was only achieved 55.84%. We believe this drop is due to the inclusion of prompts with mathematical and reasoning-based ground-truth data in the UltraFeedback dataset, which the model struggles to handle correctly, resulting in suboptimal optimization.

Evaluation of feedback sources. To comprehensively evaluate the effectiveness of different feedback sources, we compared multiple feedback mechanisms. We tested several baseline approaches: Random (randomly selecting responses), Length-Based (preferring shorter responses), ArmoRM (using ArmoRM-Llama3-8B-v0.1 (Wang et al., 2024)), as well as feedback from external models (GPT-4o, GPT-4o-mini) and self-feedback. Additionally, we conducted an ablation study on the judging criteria used in our self-feedback template.

As shown in Table 6, the results reveal clear patterns. For Gemma-2-9B-it, self-feedback significantly outperforms all alternatives, achieving WR of 74.04% compared to GPT-4o feedback (52.80%), GPT-4o-mini feedback (68.51%), and ArmoRM feedback (41.43%). Heuristic methods (Random: 8.82%; Length-Based: 3.29%) performed poorly, confirming the necessity of meaningful evaluation. The ablation study shows all criteria contribute to performance, with Criterion 4 being most critical (72.67% when removed).

We further conducted a fine-grained quality analysis on responses selected by Gemma-2-9B-SAO feedback for given instructions, using five explicit metrics (0-10 scale), as shown in Table 7. Self-feedback consistently

Table 7: Quality assessment of feedback from different sources in Gemma-2-9B-SAO dataset.

Metric	**Self**	**GPT-4o-mini**	**GPT-4o**
Correctness	9.41	9.37	9.36
Completeness	9.45	9.42	9.34
Clarity	9.85	9.82	9.81
Usefulness	9.43	9.39	9.33
Relevance	9.78	9.76	9.76
Avg. Score	**9.58**	9.55	9.52

Table 8: Performance of Gemma-2-9B-it under different generation and judging configurations. Generator produces response pairs; judge selects preferred responses.

Generator + Judge	**LC (%)**	**WR (%)**
Vanilla Gemma-2-9B-it	56.50	39.30
Gemma + LLaMA	58.39	34.35
LLaMA + LLaMA	59.32	34.41
LLaMA + Gemma	67.30	72.73
Gemma + Gemma	**70.28**	**74.06**

achieves the highest average score (9.58), particularly excelling in Clarity (9.85) and Relevance (9.78).

Quality of generation vs. judging. To disentangle the relative contributions of generation quality versus judging ability, we trained Gemma-2-9B-it on SAO datasets constructed using different combinations of generators and judges. Specifically, we used either Gemma-2-9B-it or LLaMA-3-8B-instruct as the generator (producing response pairs for prompts generated by Gemma-2-9B-it) and as the judge (selecting preferred responses from these pairs).

As shown in Table 8, judging ability has a substantially greater impact on alignment performance than generation quality. When Gemma serves as the judge, both configurations achieve strong results: Gemma-generated + Gemma-judged yields the best performance (LC: 70.28%, WR: 74.06%), while LLaMA-generated + Gemma-judged also performs well (LC: 67.30%, WR: 72.73%). In contrast, when LLaMA serves as the judge, performance drops significantly regardless of the generator—even Gemma-generated responses with LLaMA judging achieve only LC: 58.39% and WR: 34.35%, barely improving over the vanilla baseline.

These results demonstrate that judging quality is the critical factor in alignment effectiveness. A model with strong judging ability can effectively utilize preference data regardless of its generation source, while weak judging ability limits performance even with high-quality generated data. This finding validates our core claim that leveraging a model's intrinsic judging capability can effectively drive self-improvement in alignment.

Multi-judge evaluation. To further validate the robustness of SAO improvements and address potential evaluation bias, we evaluated our models using multiple independent judge models beyond the default GPT-4 and Qwen2-72B-instruct judges in AlpacaEval 2.0. Specifically, we employed three diverse state-of-the-art LLM evaluators: LLaMA-3.3-70B, LLaMA-3.1-70B, and Qwen-2.5-72B to evaluate models trained with self-feedback and LLaMA-feedback.

Table 9: Multi-judge evaluation on AlpacaEval 2.0 showing consistent improvements across different evaluators.

Judge	**Model Variant**	**LC (%)**	**WR (%)**
LLaMA-3.3-70B	Gemma-2-9B-it	64.19	45.22
	+ SAO (self)	**68.26**	**73.11**
	+ SAO (LLaMA)	59.96	36.77
LLaMA-3.1-70B	Gemma-2-9B-it	58.46	42.66
	+ SAO (self)	**64.54**	**69.50**
	+ SAO (LLaMA)	58.65	38.39
Qwen-2.5-72B	Gemma-2-9B-it	61.33	42.61
	+ SAO (self)	**71.89**	**76.09**
	+ SAO (LLaMA)	55.82	32.67

As shown in Table 9, Gemma-2-9B-it with self-feedback demonstrates consistent and substantial improvements across all three judges. With LLaMA-3.3-70B as the evaluator, WR increases from 45.22% to 73.11%; with LLaMA-3.1-70B, from 42.66% to 69.50%; and with Qwen-2.5-72B, from 42.61% to 76.09%. The LC metric also shows steady gains across all evaluators.

In contrast, SAO trained with LLaMA-feedback consistently underperforms the self-feedback approach, with WR ranging from 32.67% to 38.39% across different judges. These results confirm that the performance gains from our SAO method are robust and generalize across different evaluation perspectives, demonstrating genuine alignment improvements rather than judge-specific overfitting.

5 Conclusion

In this paper, we propose a self-alignment optimization framework for aligning LLMs with human preferences using only self-generated synthetic data. Our approach enables model itself to generate and judge instruction-preference pairs, eliminating the need for costly external annotations. Through comprehensive experiments, we demonstrate that our framework achieves substantial performance gains on subjective evaluation benchmarks while maintaining or improving performance on downstream objective tasks. We further demonstrate promising scalability through dataset size scaling and iterative optimization, where improved models generate higher-quality preference data for subsequent rounds. We hope SAO provides a practical path toward self-improvement and unlocking the latent capabilities of LLMs.

6 Limitations

While our experimental results are promising, this study is constrained by the use of models smaller than 10 billion parameters due to resource limitations. We anticipate that scaling the SAO framework to larger models could yield even greater performance enhancements. Additionally, although our approach has demonstrated effectiveness with simple prompt templates, investigating more complex templates may provide further improvements. Future research should address these areas to fully leverage the potential of the SAO framework.

References

Afra Amini, Tim Vieira, and Ryan Cotterell. 2024. Direct preference optimization with an offset. *arXiv preprint arXiv:2402.10571*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, Jiayin Zhang, Juanzi Li, and Lei Hou. 2023. Benchmarking foundation models with language-model-as-an-examiner. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Edward Beeching, Clémentine Fourrier, Nathan Habib, Sheon Han, Nathan Lambert, Nazneen Rajani, Omar Sanseviero, Lewis Tunstall, and Thomas Wolf. 2023. Open LLM leaderboard. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. 2024a. AlpaGasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations*.

Wei-Lin Chen, Zhepei Wei, Xinyu Zhu, Shi Feng, and Yu Meng. 2025. Do LLM evaluators prefer themselves for a reason? *arXiv preprint arXiv:2504.03846*.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. 2024b. Self-play fine-tuning converts weak language models to strong language models.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing GPT-4 with 90%* chatgpt quality.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv e-prints*, pages arXiv–2210.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Ultrafeedback: Boosting language models with scaled ai feedback.

Grégoire Delétang, Anian Ruoss, Paul-Ambroise Duquenne, Elliot Catt, Tim Genewein, Christopher Mattern, Jordi Grau-Moya, Li Kevin Wenliang, Matthew Aitchison, Laurent Orseau, Marcus Hutter, and Joel Veness. 2024. Language modeling is compression.

Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. 2023. Rephrase and respond: Let large language models ask better questions for themselves. *arXiv preprint arXiv:2311.04205*.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2024. Alpacafarm: A simulation framework for methods that learn from human feedback.

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083, Singapore. Association for Computational Linguistics.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024a. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. 2024b. In-context autoencoder for context compression in a large language model.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. Orpo: Monolithic preference optimization without reference model.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction. *arXiv preprint arXiv:2303.04132*.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, et al. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. *arXiv preprint arXiv:2310.08491*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. RLAIF: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2024. Rlaif vs. rlhf: Scaling reinforcement learning from human feedback with ai feedback.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. 2024a. From live data to high-quality benchmarks: The Arena-Hard pipeline.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2024b. Self-alignment with instruction backtranslation. In *The Twelfth International Conference on Learning Representations*.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023a. AlpacaEval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023b. Textbooks are all you need ii: phi-1.5 technical report.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. 2024. Mitigating the alignment tax of rlhf.

Bingbin Liu, Sebastien Bubeck, Ronen Eldan, Janardhan Kulkarni, Yuanzhi Li, Anh Nguyen, Rachel Ward, and Yi Zhang. 2023. Tinygsm: achieving> 80% on gsm8k with small language models. *arXiv preprint arXiv:2312.09241*.

Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions. *arXiv preprint arXiv:2104.08773*.

OpenAI. 2023. GPT-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2023. Rephrase, augment, reason: Visual grounding of questions for vision-language models. *arXiv preprint arXiv:2310.05861*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*.

Baptiste Roziere, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Swarnadeep Saha, Omer Levy, Asli Celikyilmaz, Mohit Bansal, Jason Weston, and Xian Li. 2023. Branch-solve-merge improves large language model evaluation and generation. *arXiv preprint arXiv:2310.15123*.

Arthur L Samuel. 2000. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 44(1.2):206–226.

Jiaming Shen, Ran Xu, Yennie Jun, Zhen Qin, Tianqi Liu, Carl Yang, Yi Liang, Simon Baumgartner, and Michael Bendersky. 2024. Boosting reward model with preference-conditional multi-aspect synthetic data generation.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

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

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in llms: A survey of role-playing and personalization.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Sanh Victor, Webson Albert, Raffel Colin, Bach Stephen, Sutawika Lintang, Alyafeai Zaid, Chaffin Antoine, Stiegler Arnaud, Raja Arun, Dey Manan, et al. 2022. Multitask prompted training enables zero-shot task generalization. In *International Conference on Learning Representations*.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. *arXiv preprint arXiv:2406.12845*.

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

Zhepei Wei, Wei-Lin Chen, and Yu Meng. 2025a. InstructRAG: Instructing retrieval-augmented generation via self-synthesized rationales. In *The Thirteenth International Conference on Learning Representations*.

Zhepei Wei, Xiao Yang, Kai Sun, Jiaqi Wang, Rulin Shao, Sean Chen, Mohammad Kachuee, Teja Gollapudi, Tony Liao, Nicolas Scheffer, et al. 2025b. TruthRL: Incentivizing truthful LLMs via reinforcement learning. *arXiv preprint arXiv:2509.25760*.

Yue Wu, Zhiqing Sun, Huizhuo Yuan, Kaixuan Ji, Yiming Yang, and Quanquan Gu. 2024. Self-play preference optimization for language model alignment.

Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. 2024a. Is DPO superior to PPO for LLM alignment? a comprehensive study. *arXiv preprint arXiv:2404.10719*.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024b. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing.

Yu Yang, Aaditya K Singh, Mostafa Elhoushi, Anas Mahmoud, Kushal Tirumala, Fabian Gloeckle, Baptiste Rozière, Carole-Jean Wu, Ari S Morcos, and Newsha Ardalani. 2023. Decoding data quality via synthetic corruptions: Embedding-guided pruning of code data. *arXiv preprint arXiv:2312.02418*.

Shangjian Yin, Peijie Huang, JiaTian Chen, Haojing Huang, and Yuhong Xu. 2025a. ECLM: Entity level language model for spoken language understanding with chain of intent. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Shangjian Yin, Peijie Huang, and Yuhong Xu. 2025b. MIDLM: Multi-intent detection with bidirectional large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models.

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

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *NeurIPS Datasets and Benchmarks Track*.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. 2025. The surprising effectiveness of negative reinforcement in LLM reasoning. *arXiv preprint arXiv:2506.01347*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

A Impact Statements

The introduction of the SAO framework offers a valuable solution to the challenges of fine-tuning LLMs without extensive external supervision. This approach can significantly reduce the manual effort and time required for model training, thereby enhancing accessibility to NLP technologies for individuals and organizations with limited resources. However, it is crucial to exercise caution with this self-improvement framework, as it relies entirely on self-synthesized datasets, which may lead to the generation of inaccuracies or hallucinations in certain cases.

B Experiments on Small-Sized LLMs

Interestingly, when we extend our SAO framework to smaller models (1B-3B), we observe that the initial subjective and alignment capabilities of the backbone model significantly impact the results. For instance, Llama-3.2-Instruct-3B shows a performance degradation after applying SAO, while Gemma-2-2B-it demonstrates notable improvements. We hypothesize that weaker models may lack sufficient judgment ability to effectively conduct self-alignment.

Table 10: Performance of small language models on AlpacaEval 2.0 using Qwen2-72B-Instruct as the judge.

Model	AlpacaEval 2.0		
	LC (%)	WR (%)	STD
Llama-3.2-Instruct-3B	21.87	24.97	1.52
Llama-3.2-Instruct-3B-SAO	20.08 (-1.79)	22.05 (-2.92)	1.46
Gemma-2-2B-it	38.08	41.74	1.74
Gemma-2-2B-it-SAO	46.18 (+8.10)	49.13 (+7.39)	1.76

C Additional Result on Math500

We additionally evaluate the downstream performance of the model on the Math500 benchmark. As shown in Table 11, our method outperforms the baselines, demonstrating the effectiveness of the proposed approach on mathematical reasoning tasks.

Table 11: Performance comparison on the Math500 benchmark using Gemma-2-9B-It as the backbone model.

Model	**Math500(%)**
Gemma-2-9B-It	46.40
Gemma-2-9B-It-SimPO	47.60
Gemma-2-9B-It-SAO (Ours)	**49.60**

D Logits Shift Analysis

To further investigate the impact of different training methods on the model's internal representations, we analyze the logits shift of the generated responses. We randomly sample 20 prompts from the AlpacaEval 2.0 and Math500 datasets, respectively. For each output, we extract the logits, defined as the unnormalized scores $z_i \in \mathbb{R}$ produced by the model before applying the softmax function:

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}. \tag{8}$$

To analyze the overall distributional behavior, we flatten all collected logits across all responses into a single one-dimensional vector $\mathbf{z}_{\text{flat}} = \{z_1^{(1)}, \dots, z_{m_k}^{(k)}\}$, where k is the total number of generated samples. To quantify the asymmetry of the flattened logits distribution, we compute the skewness:

$$\text{Skewness} = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{z_i - \mu}{\sigma} \right)^3, \tag{9}$$

where μ and σ are the mean and standard deviation of the flattened logits vector $\mathbf{z}_{\text{flat}}$, respectively. A positive skewness indicates a right-tailed distribution.

As visualized in Figure 4, we compare the logits distributions of Base, SimPO, and our SAO method. We observe that SAO makes the model's logits distribution slightly right-skewed compared to the Base model. This positive skewness suggests that SAO encourages the model to be more confident in its correct predictions without fundamentally disrupting the original distributional characteristics.

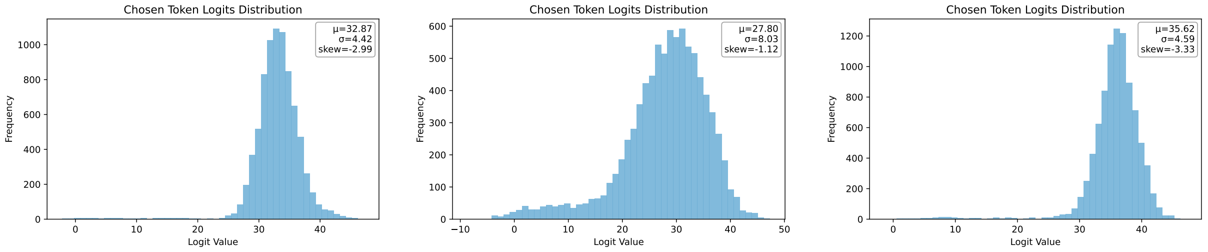
In contrast, SimPO (optimized using externally labeled datasets) significantly alters the logits distribution. Although this shift does not degrade performance on Math500, we hypothesize that such a drastic distributional change may impose a potential "alignment tax" on other downstream tasks. By significantly deviating from the base model's intrinsic representation, the model risks losing generalization capabilities on broader reasoning domains. SAO, by preserving the intrinsic distribution while enhancing confidence, effectively mitigates this risk, achieving a robust balance between alignment and general reasoning capability.

E Repetition in Generated Prompts Under Random Generation

We also investigated the diversity of the generated data under random sampling conditions. Table 12 lists the top five most frequently recurring prompts within the 10k synthesized dataset generated without persona role-play. As observed, the model exhibits a strong tendency to repeat specific topics.

F Prompt Template

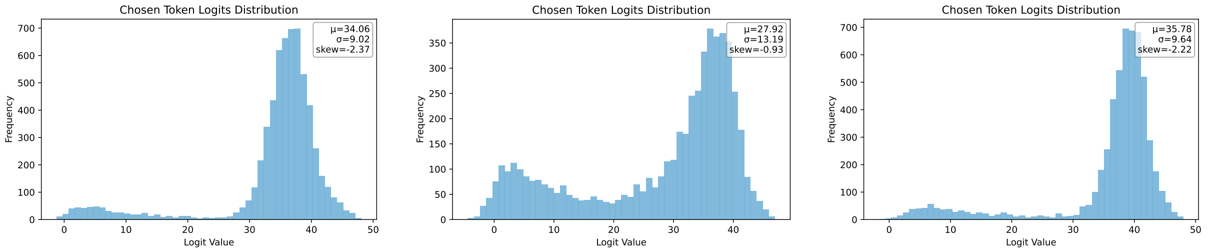
To standardize our data generation and evaluation processes, we employed specific prompt templates shown in Figure 5. The upper panel illustrates the Persona Instruction template used to synthesize diverse and domain-specific prompts by conditioning the model on a specific persona. The lower panel displays the Pairwise Response Ranking template, which instructs the



(a) Base (Alpaca)

(b) SimPO (Alpaca)

(c) SAO (Alpaca)



(d) Base (Math500)

(e) SimPO (Math500)

(f) SAO (Math500)

Figure 4: Logits distribution comparison on AlpacaEval 2.0 (top) and Math500 (bottom). SAO maintains a distribution shape similar to the Base model but with a slight right-skew, indicating increased confidence. In contrast, SimPO significantly alters the logits distribution.

Table 12: Top 5 repeated randomly generated prompts without persona role-play in 10k synthesized dataset.

Prompt	**Count**
Summarize the main plot points of the novel "Pride and Prejudice" by Jane Austen, focusing on the development of the relationship between Elizabeth Bennet and Mr. Darcy.	267
Summarize the main plot points of the novel "Pride and Prejudice" by Jane Austen, focusing on the relationship between Elizabeth Bennet and Mr. Darcy.	258
Write a short story (around 500 words) about a young inventor who creates a device that can translate animal languages, but struggles with the ethical implications of their invention.	178
Summarize the main plot points of the novel *Pride and Prejudice* by Jane Austen, focusing on the relationship between Elizabeth Bennet and Mr. Darcy.	114
Summarize the main plot points of the novel *Pride and Prejudice* by Jane Austen, focusing on the development of the relationship between Elizabeth Bennet and Mr. Darcy.	105

model to act as an impartial judge, evaluating and ranking candidate responses based on relevance, accuracy, completeness, and clarity.

G Examples

Table 13 illustrates a representative example from our synthesized dataset, while Table 14 provides a qualitative comparison demonstrating the improved comprehensiveness of our SAO-trained model compared to the baseline.

Persona Instruction Example

Guess a prompt that the following persona may ask you to do:

A Political Analyst specialized in El Salvador's political landscape.

Note:
1. The prompt should be informative and specific.
2. Your output should start with "User prompt:"

Pair-wise Response Ranking

You are an impartial judge. Your task is to rank two answers to a given prompt based on their quality.

Prompt: {prompt}

Response 1: `<Response 1>` {response 1} `</Response 1>`

Response 2: `<Response 2>` {response 2} `</Response 2>`

Please carefully read each response and evaluate them based on the following criteria:
1. Relevance and specificity to the prompt
2. Accuracy and correctness of information
3. Completeness and comprehensiveness
4. Clarity and understandability

Then, rank these two responses from best to worst. You must output your ranking strictly in the following format: **ranking: X > Y**, where X and Y represent one of 1 or 2, without repetition.

Remember, you must output a complete ranking including both options. Now, please provide your ranking:

Figure 5: The top box displays the persona instruction prompt, which directs the LLM to generate a specific prompt based on a given persona. The bottom box illustrates the pair-wise response ranking prompt, instructing the LLM to compare and rank responses based on specific criteria modified from Shen et al. (2024).

Input Persona	A newly hired general counsel at TurpCo Industries
Generated Prompt	As a newly hired General Counsel at TurpCo Industries, I need to understand the legal landscape of our operations. Could you provide a detailed report on the major legal risks and compliance issues our company might face in the next fiscal year, including any potential changes in legislation that could affect our business? Additionally, please include recommendations for mitigating these risks and ensuring our compliance with all relevant laws and regulations.
Response 1	**Executive Summary:** As the General Counsel of TurpCo Industries, it is essential to understand the legal risks and compliance issues that our company may face in the next fiscal year. This report provides an overview of the major legal risks and compliance issues that our company may face, including potential changes in legislation that could affect our business. The report also includes recommendations for mitigating these risks and ensuring our compliance with all relevant laws and regulations. **Legal Risks and Compliance Issues:** 1. **Data Privacy and Cybersecurity:** As the world becomes increasingly digital, data privacy and cybersecurity risks have become more prevalent. Our company must ensure that we comply with all relevant data protection laws and regulations, such as the General Data Protection Regulation (GDPR) in Europe and the California Consumer Privacy Act (CCPA) in the United States. We must also have robust cybersecurity measures in place to protect our company's and our customers' data......
Response 2	**Subject: Legal Risk Assessment and Compliance Report for TurpCo Industries** Dear [Recipient's Name], I am pleased to present a comprehensive report on the major legal risks and compliance issues that TurpCo Industries may face in the upcoming fiscal year. This report will also address potential legislative changes that could impact our operations, along with recommendations for risk mitigation and ensuring legal compliance. 1. **Overview of Legal Risks and Compliance Issues:** a. **Environmental Regulations:** - **Risk:** TurpCo Industries' manufacturing processes may be subject to stringent environmental regulations, including those related to emissions, waste disposal, and water usage. - **Potential Changes:** The Environmental Protection Agency (EPA) is considering updating regulations to reduce greenhouse gas emissions and tighten waste disposal standards.....
Ranking	2 > 1

Table 13: Example of synthetic dataset.

Instruction
What are the names of some famous actors that started their careers on Broadway?
Gemma2-9B-it Output
Here are some famous actors who started their careers on Broadway: **Legends:** * **Meryl Streep:** Made her Broadway debut in 1975 and won a Tony Award for "The Cherry Orchard." * **Denzel Washington:** Won a Tony Award for his performance in "Fences" and has starred in many other Broadway productions. * **Al Pacino:** Rose to fame on stage before transitioning to film, winning a Tony Award for "Does a Tiger Wear a Necktie?" * **Robert De Niro:** Made his Broadway debut in 1964 and has since returned for several productions. * **James Earl Jones:** Known for his iconic voice, he won a Tony Award for "Fences" and has been a mainstay on Broadway for decades. **More Recent Stars:** * **Lin-Manuel Miranda:** Created and starred in the smash hit "Hamilton," winning multiple Tony Awards for his work. * **Andrew Garfield:** Earned Tony Award nominations for "Death of a Salesman" and "Angels in America." * **Scarlett Johansson:** Made her Broadway debut in "A View from the Bridge" and has since returned for other productions. * **Nathan Lane:** A Tony Award-winning actor known for his comedic roles in musicals like "The Producers" and "Guys and Dolls." * **Hugh Jackman:** Known for his role as Wolverine, he has also starred in several Broadway musicals, winning a Tony Award for "The Boy From Oz." This is just a small sample of the many talented actors who have honed their craft on Broadway. Let me know if you'd like to explore actors from a specific genre or era!
Gemma2-9B-it-SAO Output
Here's a list of famous actors, spanning generations and genres, who honed their craft and gained initial recognition on Broadway: **Stage Legends & Classics:** * **Marlon Brando, James Dean, Montgomery Clift:** Pioneered Method acting, rose to fame through iconic stage productions before Hollywood. * **Humphrey Bogart, Katharine Hepburn, Laurence Olivier, Ingrid Bergman:** Established themselves as stage stars before transitioning to film. * **Judy Garland, Ethel Merman, Carol Channing, Yul Brynner:** Musical theatre icons who built enduring careers on Broadway. * **Angela Lansbury, Julie Andrews, Audra McDonald:** Multiple Tony Award winners with long, distinguished Broadway careers. **Contemporary Stars:** * **Denzel Washington, Viola Davis, Nathan Lane, Patrick Stewart:** Highly acclaimed actors with both stage and screen success. * **Tom Hanks, Meryl Streep, Daniel Day-Lewis, Michelle Williams:** Oscar winners who began their careers in theatre. * **Idina Menzel, Lin-Manuel Miranda, Andrew Garfield, Jake Gyllenhaal:** Known for both stage and screen, often returning to Broadway. * **Hugh Jackman, Bradley Cooper, Emma Stone, Scarlett Johansson:** Made their mark on Broadway before achieving major film success. **Notable Mentions:** * **Robert De Niro, Al Pacino, Christopher Walken:** Strong stage backgrounds, though perhaps less known for their Broadway work specifically. This list is by no means exhaustive, but it highlights the rich history of actors who found their footing and developed their skills on the Broadway stage.

Table 14: Outputs from Gemma2-9B-it and Gemma2-9B-it-SAO models.