Condor: Enhance LLM Alignment with Knowledge-Driven Data Synthesis and Refinement

Maosong Cao[1,*], Taolin Zhang[1,2,*], Mo Li[1,2], Chuyu Zhang[1], Yunxin Liu[2]
Haodong Duan[1,†], Songyang Zhang[1,†,‡], Kai Chen[1,†]
[1]Shanghai AI Laboratory [2]Tsinghua University
Datasets:https://hf.co/datasets/internlm/Condor-SFT-20K
Github:https://github.com/InternLM/Condor

Abstract

The quality of Supervised Fine-Tuning (SFT) data plays a critical role in enhancing the conversational capabilities of Large Language Models (LLMs). However, as LLMs become more advanced, the availability of high-quality human-annotated SFT data has become a significant bottleneck, necessitating a greater reliance on synthetic training data. In this work, we introduce **Condor**, a novel two-stage synthetic data generation framework that incorporates **World Knowledge Tree** and **Self-Reflection Refinement** to produce high-quality SFT data at scale. Our experimental results demonstrate that a base model fine-tuned on only 20K Condor-generated samples achieves superior performance compared to counterparts. The additional refinement stage in Condor further enables iterative self-improvement for LLMs at various scales (up to 72B), validating the effectiveness of our approach. Furthermore, our investigation into the scaling for synthetic data in post-training reveals substantial unexplored potential for performance improvements, opening promising avenues for future research.[1]

1 Introduction

Large Language Models (LLMs) are experiencing rapid advancements, with proprietary models such as like GPT (Achiam et al., 2023) and Gemini (Team et al., 2023), alongside open-source counterparts such as LLaMA (Dubey et al., 2024), Qwen (Yang et al., 2024), Mistral (Jiang et al., 2023a), and Deepseek (Liu et al., 2024a) evolving at an unprecedented pace. However, this rapid iteration comes with a significant challenge: the depletion of existing high-quality data for Supervised Fine-Tuning (SFT). Moreover, the internet is

Figure 1: **Comparison between Official Models and Condor-Refined Models.** The performances are obtained from CompassJudger-1-32B. Different families of popular LLMs benefit greatly from the Condor Refine engine.

increasingly inundated with synthetic data of varying and often questionable quality, making it ever more difficult to construct and filter higher-quality training data for LLMs.

Empirical studies on SFT training have conclusively shown that both data quality and quantity play crucial roles in enhancing model performance (Shen, 2024). While scaling laws suggest that models can achieve extraordinary capabilities when trained on large datasets, high-quality data can yield comparable results even at smaller scales. This highlights the importance of generating substantial amounts of high-quality data to further enhance large-scale model capabilities. Recent research introduce various methods for synthesizing SFT training data (Wang et al., 2024a; Ding et al., 2023; Xu et al., 2023; Yuan et al., 2024; Tang et al., 2024). However, these approaches face several lim-

---

[1]This work is done when Taolin Zhang and Chuyu Zhang are on internship at Shanghai AI Laboratory, ∗ means equal contribution, † means corresponding author, ‡ means project lead.
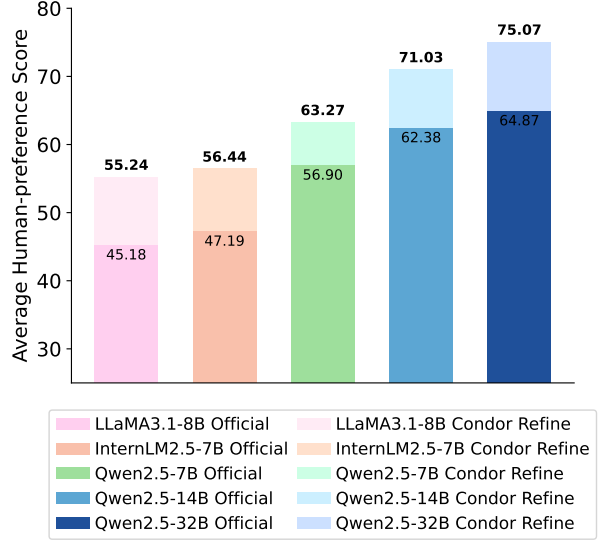
itations: they often rely heavily on existing datasets for seed prompts, lack prompt diversity, and cannot guarantee the quality of the generated data.

To address these challenges, we introduce **Condor**, an automated two-stage pipeline for synthetic data generation that effectively leverages existing world knowledge. Our approach begins with constructing a comprehensive **World Knowledge Tree** by prompting LLM to generate tags, and then creating a complete tag chain from coarse to fine through an auto-regressive pipeline. After generating diverse questions and original responses using tags, we then employ **Self-Reflection Refinement** to obtain synthesized SFT data with more higher-quality replies. Condor further generates extensive and high-quality SFT data by integrating up-to-date world knowledge tags. In this way, Condor not only ensures the relevance and diversity of the content but also maintains scalability in data synthesis. We conduct experiments on multiple models using the data synthesized by Condor, as shown in Figure 1, the results indicate that the data synthesized by Condor significantly improves the subjective chat capabilities of the models across the board.

To summarize, our contributions are as follows:

• We propose **Condor Void**, a novel approach incorporating a knowledge inspiration strategy based on the World Knowledge Tree, enabling models to acquire rich world knowledge for generating diverse questions.

• We further introduce **Condor Refine**, a self-reflection mechanism that allows models to iteratively refine their responses during the construction phase, resulting in higher-quality responses.

• We investigate the scaling and self-iteration techniques for Condor datasets, offering valuable insights and potential pathways for the scalable expansion of synthetic data.

2 Related Work

2.1 Synthesizing Instruction Tuning Data

With the rapid advancement of Large Language Models (LLMs) (Achiam et al., 2023; Team et al., 2023; Dubey et al., 2024; Yang et al., 2024; Jiang et al., 2023a; Liu et al., 2024a) and Large Multi-Modality Models (LMMs) (Radford et al., 2021; Zhu et al., 2023; Zhang et al., 2024a,b; Liu et al., 2024b; Chen et al., 2024), the demand for high-quality training data continues to grow. As a result, synthetic data generation has become increasingly crucial for model development. While high-quality

synthetic data can significantly enhance model performance, low-quality synthetic data may lead to model degradation or collapse.

WizardLM (Xu et al., 2023) introduces an Evol-Instruct scheme, which expands an initial question dataset by generating new questions through techniques such as deep thinking, adding constraints, and reconstruction. However, the quality and diversity of the generated data are strongly influenced by the characteristics of the original dataset. Similarly, Self-Reward (Yuan et al., 2024) builds on a seed dataset, enabling the model to generate questions via few-shot prompting and produce multiple corresponding responses. These responses are subsequently evaluated by a reward model and labeled as training data for reinforcement learning with human feedback (RLHF).

Magpie(Xu et al., 2024) explores leveraging models themselves to generate synthetic data by utilizing chat templates as prompts to directly produce dialogues. However, this approach is not universally applicable to all model types, and therefore lacks generalization ability. Additionally, the synthetic data generated often lacks stability, containing significant noise that requiring extensive filtering to maintain quality. Agent-based data synthesis methods, such as MATRIX-Gen (Tang et al., 2024), simulate multiple world scenarios using a multi-agent framework and collect dialogue data from interactions within these scenarios. While effective in some cases, this method is time-consuming, labor-intensive, and limited by the need for multiple models to operate concurrently.

In contrast to these methods, Condor eliminates the dependence on a seed dataset, requiring only tags as seeds for expansion and generation. This enables exceptional diversity and scalability in data production. Furthermore, Condor handles all data generation tasks with a single model, significantly reducing costs while maintaining high efficiency.

2.2 LLM Self-Iteration

Recent studies have explored methods for large language models (LLMs) to improve themselves through iterative self-enhancement. For instance, I-SHEEP (Liang et al., 2024) and Self-Instruct (Wang et al., 2023) utilize seed data and LLMs to generate new instructions. However, these methods often produce instructions that closely resemble those in the seed dataset, limiting the diversity and novelty of the generated data. (Tao et al., 2024) categorizes existing self-iteration meth-
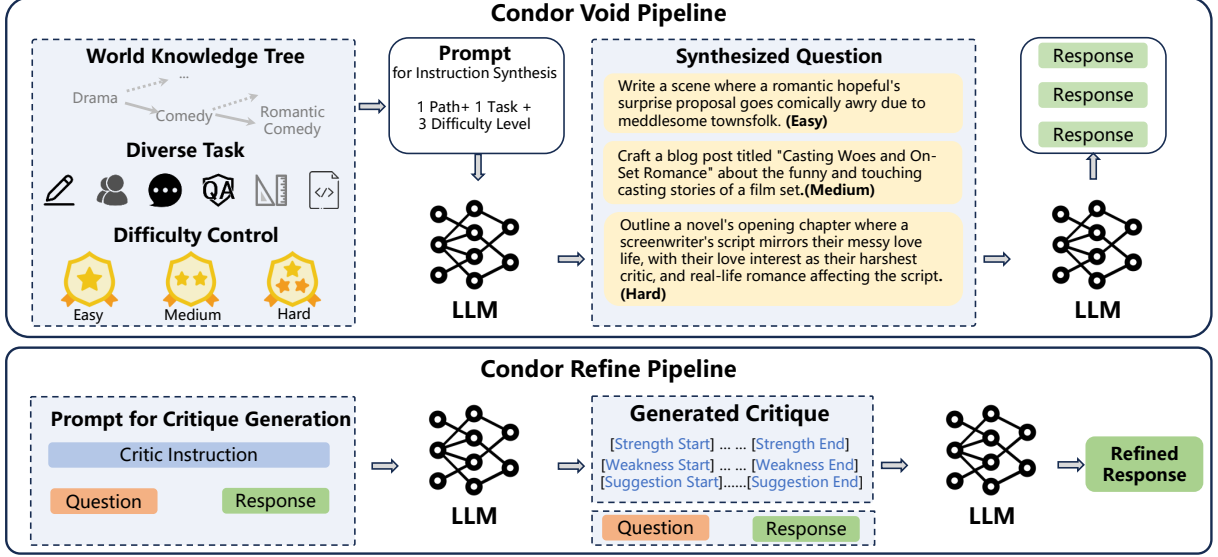
Figure 2: **The Pipeline of Condor Engine.** The pipeline is mainly divided into (1) Data synthesis: LLM is inspired from the **World Knowledge Tree** with different task diversity and difficulty to construct the original question-answer dataset D_V; (2) Data refinement: LLM is utilized to reflect on the synthesized data and generate critiques for refinement and improve the quality of responses, producing the dataset D_R.

ods, highlighting works such as self-instruct (Wang et al., 2023), self-improving (Huang et al., 2022), and self-training (Gulcehre et al., 2023). Specifically, the self-improving framework proposed by (Huang et al., 2022) enhances the reasoning abilities of LLMs without requiring labeled data by generating multiple reasoning paths and answers, then selecting high-confidence answers through a self-consistency mechanism.

Reinforced Self-Training (ReST) (Gulcehre et al., 2023) achieves self-iteration by generating multiple outputs, scoring them with a reward model, and fine-tuning the LLM using the highest-scoring samples. While effective, this approach depends on an external reward model to provide additional knowledge. Other notable works, such as Reflexion (Shinn et al., 2023), Self-Refine (Madaan et al., 2023), and Refiner (Paul et al., 2024), propose self-iteration techniques where models iteratively refine their answers based on feedback from specific tasks. However, these methods are primarily designed for specific domains like mathematics, limiting their ability for generalization.

By contrast, Condor achieves self-improvement on a wide range of human-preference datasets solely using the model's own capabilities, enabling broader applicability while maintaining scalability and efficiency.

3 Methodology

3.1 Pipeline Overview

As shown in Figure 2, the pipeline of Condor is divided into two stages: **data synthesis** and **data refinement**. In the data synthesis stage, we begin by introducing the **World Knowledge Tree**, which serves as a foundation of tags for data generation. Next, we apply task and difficulty expansion to enhance the diversity and complexity of questions under each tag, leading to the creation of the initial synthetic QA dataset D_V. In the data refinement stage, we employ a **Self-Reflection Refinement** strategy, enabling the model to iteratively optimize the generated responses and yield D_R. Notably, we utilize a single model during the entire pipeline for better efficiency.

3.2 Condor Void

In the data synthesis stage, we generate the synthetic data based on existing world knowledge. This stage consists of two key components: **(1) World Knowledge Tree** and **(2) Q&A pair generation**. The dataset produced from this stage is referred to as D_V.

3.2.1 World Knowledge Tree

To construct a comprehensive multi-level tag system that encapsulates diverse aspects of world knowledge, we propose a methodology that en-

hances the model's capacity for inspiration and knowledge stimulation. We define a set of root tags, *i.e.*, $R = \{r_1, r_2, \ldots, r_n\}$, representing broad themes such as *marriage*, *entertainment*, and *artificial intelligence*. Using large language models (LLMs), we expand each root tag into a set of detailed leaf tags, denoted as $L_i = \{l_{i1}, l_{i2}, \ldots, l_{im}\}$. This hierarchical structure forms the foundation of the World Knowledge Tree, $\mathcal{T}$, which can be expressed as a union of subtrees rooted at each r_i: $\mathcal{T} = \bigcup_{i=1}^{n}\{(r_i, L_i)\}$.

To ensure the tree remains reflective of real-world themes and contexts, we augment it with additional tags collected from trending topics on platforms such as Zhihu and Reddit, denoted as $S = \{s_1, s_2, \ldots, s_k\}$. This creates an enriched tree, $\mathcal{T} = \bigcup_{i=1}^{n}\{(r_i, L_i \cup S_i)\}$, where $S_i \subseteq S$ ensures each root branch is contextually informed and relevant. Moreover, the World Knowledge Tree is designed to be dynamic, adapting over time with an update function $U(\mathcal{T}, t)$ that that continuously integrates new information. This robust, evolving framework, comprising over 8,400 tags, is pivotal for maintaining the current and applicable nature of knowledge required for the iteration of LLMs.

3.2.2 Q&A Pair Generation

After acquiring the seed tags, we employ various methods to enhance the diversity of the generated questions. First, we identify and summarize the most common chat scenarios in human interactions with LLMs, such as daily conversation, creative tasks, and role-playing (the total 7 tasks with detailed descriptions are provided in Figure A.2).

The model is then required to deeply engage in a specific task under the corresponding tag theme to generate questions. This approach enhances the diversity of the generated questions and encourages knowledge extrapolation, allowing the model to explore themes it may not have frequently encountered—or not encountered at all—in a particular scenario. To further expand the range of questions, we implement difficulty control, prompting the model to adapt to three difficulty levels and generate questions of varying complexity. This step ensures greater diversity and richness in the generated data, contributing to a more well-rounded and comprehensive dataset. Finally, we obtain Condor Void datasets with approximately 200k Q&A pairs, *i.e.*, $|\boldsymbol{D_V}| \approx 200,000$.

3.3 Condor Refine

In this stage, we enable the model to engage in self-reflection and critical examination of its own responses, followed by revisiting the questions to generate higher-quality replies. The resulting refined dataset is referred as $\boldsymbol{D_R}$.

The initial QA pairs contain responses directly generated by the model, which are inherently limited by the model's current capabilities. Therefore, we implement data refinement to further improve the quality of the responses. We carefully develop a set of fixed templates that guide the model in identifying both strengths and weaknesses in its responses, leading to specific suggestions for improvement. Subsequently, we utilize these self-generated improvement suggestions to prompt the model to produce enhanced responses. The model is tasked with preserving the effective elements of its original responses while addressing identified weaknesses to generate superior responses. Complete examples of QA pairs that have undergone the entire Condor pipeline are provided in the Appendix. Finally, we obtain 200k high-quality refinement datasets from $\boldsymbol{D_V}$, *i.e.*, $|\boldsymbol{D_R}| \approx 200,000$.

4 Experiments and Results

4.1 Training and Evaluation Settings

Training Settings. We use xTuner (Contributors, 2023b) as the training framework. To ensure fairness in comparison, we set the initial learning rate to 2e-5 and train for 3 epochs in all experiments. Unless specifically stated, we start training from the base model and compare it with the official RLHF model.

Evaluation Settings. We use OpenCompass (Contributors, 2023a) and employ greedy inference to uniformly evaluate all models, ensuring fairness across various datasets. We selected 8 human-preference benchmarks for evaluation the chat capabilities of the models. We evaluate the results and reports the average normalized score at the percentage scale. Additionally, due to the high cost of conducting subjective evaluations with a paid API model, we use GPT4o as the Judge Model for main results only, and we judge with the open-source CompassJudger-1-32B (Cao et al., 2024) in ablation study and scaling experiments. Furthermore, we also select a range of knowledge-based Q&A datasets for groundtruth-based evaluations.

Datasets	Score Range	Qwen2.5-7B-Instruct	Qwen2.5-7B-Base		Qwen2.5-7B-Instruct	
		Official	Condor Void	Condor Refine	Condor Void	Condor Refine
CompassArena (Contributors, 2023a)	0~100	33.8	32.43	36.13	35.17	**40.12**
FoFo (Xia et al., 2024)	0~1	**0.52**	0.41	0.45	0.50	0.47
AlignBenchv1.1 (Liu et al., 2023)	0~10	6.22	6.04	6.20	**6.24**	6.20
AlpacaEvalv2 (Dubois et al., 2024)	0~10	34.66	32.84	44.10	38.76	**44.60**
ArenaHard (Li et al., 2024)	0~100	53.65	42.44	60.87	53.56	**61.53**
FollowBench (Jiang et al., 2023b)	0~1	0.84	0.85	0.83	**0.88**	0.85
MTBench101 (Bai et al., 2024)	0~10	**8.60**	8.22	8.37	8.41	8.43
WildBench (Lin et al., 2024)	-100~100	16.71	14.09	30.13	21.54	**32.70**
Average	0~100	58.02	54.09	60.03	59.14	**61.29**

Table 1: **Performance Comparison on Human-preference Benchmarks Judged by GPT4o-0806.** The results demonstrate that Condor Refine significantly enhances the subjective chat capability of models. The **Average Score** is calculated by normalizing the scores for each dataset to a percentage scale and then taking their average.

Datasets	Qwen2.5-7B-Instruct	Qwen2.5-7B-Base		Qwen2.5-7B-Instruct	
	Official	Condor Void	Condor Refine	Condor Void	Condor Refine
BoolQ (Clark et al., 2019)	86.06	**87.83**	86.82	87.34	86.33
GPQA Diamond (Rein et al., 2023)	35.35	34.34	**35.86**	35.35	35.35
Math (Hendrycks et al., 2021)	**83.21**	82.31	81.91	82.67	82.47
GSM8K (Cobbe et al., 2021)	**92.27**	91.21	90.75	91.43	91.96
GaokaoBench (Zhang et al., 2024c)	60.42	**63.87**	63.64	60.69	60.13
MMLU-Pro (Wang et al., 2024b)	54.70	52.15	50.70	**54.73**	53.02
DROP (Dua et al., 2019)	80.81	79.31	78.50	**81.53**	80.46
IFEval loose (Zhou et al., 2023)	81.70	81.70	79.85	**83.36**	80.96
HumanEval (Chen et al., 2021)	**89.63**	89.02	85.98	87.20	86.59
Race-high (?)	85.05	84.88	83.56	**85.16**	84.28
MBPP (Austin et al., 2021)	74.32	74.32	73.54	74.32	**74.71**
CMO (Contributors, 2023a)	22.60	25.00	**28.85**	23.08	20.19
Average	70.51	70.50	70.00	**70.57**	69.70

Table 2: **Performance Comparison on Knowledge-Based Benchmarks.** The score range across all the tasks are 0~100. When trained on high-quality SFT data, the differences in the model's performance on knowledge-intensive question-answering tasks are minimal.

4.2 Data Synthesis Settings

Only one model is required for data synthesis and data refinement throughout the entire pipeline. In the main experiments, we use 200k data synthesized by Qwen2.5-72B for model training. Additionally, for SOTA comparisons and self-interaction experiments, we also generate 200k data points using Qwen2.5-7B to test the effectiveness of the Condor pipeline.

4.3 Main Results

Using Condor data generated by Qwen2.5-72B-Instruct, we first train both the Base and Instruct versions of Qwen2.5-7B and compare their performance against the official model on both Chat and Knowledge benchmarks.

Human-preference Evaluation. We select several widely-used human-preference benchmarks and calculate the final normalized average scores on these datasets (detailed calculation logic is provided in AppendixA.1). We use GPT-4o as the Judge Model, and the average results are reported in Table 1. As shown in the table, the results of training the Base model with D_R surpassed the of-

ficial model on almost all human-perference benchmarks. It is noteworthy that this stage only employs SFT and does not include RLHF as applied in the official model, highlighting the significant potential of Condor's constructed data in enhancing the model's subjective chat capabilities. Furthermore, continue tuning on the Instruct model further improves performance, demonstrating that the data synthesized by Condor complements the original model's capabilities and can enhance human-preference performance without introducing conflicts. To further verify these results, we also conducted evaluations using CompassJudger-1-32B (Cao et al., 2024). As shown in Figure 3, evaluations using both Judge Models reached a consistent conclusion.

Knowledge Performance Evaluation. We further conduct experiments on several knowledge-based benchmarks to assess the model's objective question-answering capabilities. The detailed results are presented in Table 2. It is worth noting that our synthetic data doesn't specifically include knowledge-based QAs like the official models, yet it achieves comparable or even superior results.
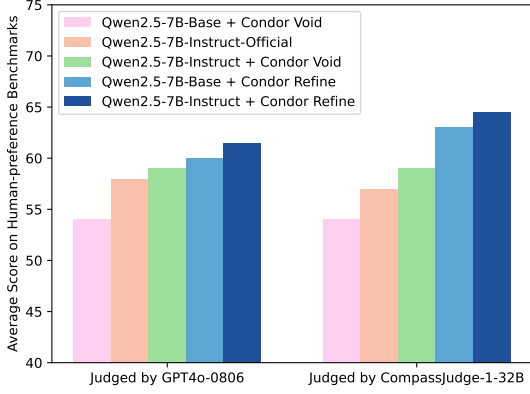
Figure 3: **Average Score on Human-preference Benchmarks.** CompassJudger-1-32B and GPT-4o provide a consistent trend in scoring.

Method	Avg Sub Score	Data Size	Synth Model
Tulu v2 (Ivison et al., 2023)	24.65	326k	Mixed
Evol Instruct (Xu et al., 2023)	34.33	70k	GPT
WildChat (Zhao et al., 2024)	39.53	520k	GPT Mixed
Magpie (Xu et al., 2024)	55.67	1M	Qwen2.5-72B
Qwen2.5-7B-Instruct	58.02	500k+	Unknown
Condor 7B	58.38	200k	Qwen2.5-7B
Condor 72B	**60.03**	200k	Qwen2.5-72B

Table 3: **Comparison Results with Other Data Synthesis Methods.** We adopt SFT data generated by various approaches to tune Qwen2.5-7B-Base and use GPT-4o as the judge model.

The outcome indicates that our dataset not only enhances the model's human-preference performance but also preserves its objective knowledge QA capabilities. Additionally, the results demonstrate that the model's chat ability and knowledge capability are largely independent. The knowledge capability primarily relies on the foundational base model's abilities, whereas the chat ability can be independently improved through SFT and RLHF.

Comparing with the State-of-the-art. We generate two sets of Condor data using Qwen2.5-7B and Qwen2.5-72B and compare them with existing state-of-the-art synthetic datasets. We use the data obtained from various data synthesis methods to train starting from the Qwen2.5-7B-Base, and evaluate the model with GPT-4o judge. As shown in Table 3, Condor data generated by both the 7B and 72B models significantly outperforms other data synthesis methods and even surpasses the official model after RLHF. This observation indicates that self-improvement is achievable even at the 7B scale, which will be further analyzed in detail in Section 4.6. Moreover, Condor requires only a single model for the entire data synthesis and refinement pipeline with and mach smaller volume of data than most synthetic datasets, highlighting

its efficiency and effectiveness.

4.4 Ablation Study

We further investigate the impact of model type, size, and question difficulty on Condor training through a series of ablation experiments using Condor data generated by Qwen2.5-72B-Instruct. For the type ablation, we evaluate models of different families, including Qwen, InternLM, and Llama. For the size ablation, we test Qwen2.5 at 7B, 14B, and 32B parameter scales. Additionally, in the difficulty ablation, we train models with datasets containing questions of varying difficulty levels. In all experiments, we utilize CompassJudger as judger for subjective evaluation.

Model Type. From Table 4, it is evident that training on D_R consistently improves performance across different models. Almost all models demonstrate significant enhancements on all subjective evaluation datasets. Compared to the official model, Qwen2.5-7B shows an improvement more than 6% (56.9% to 63.3%) after training on Condor Refine. InternLM[2] and Llama exhibit an even larger improvement by about 10%. These results highlight that the quality of the base models plays a crucial role in determining the conversational performance of the final Instruct model.

Model Size. We further evaluate the adaptability of models of different sizes to D_R. Table 5 show results aligning with the observations from Table 4. Interestingly, we find that the improvement in performance for the 14B and 32B models is notably greater compared to the 7B variant. This suggests that larger models demonstrate a greater capacity to learn and benefit from the refined data and achieve better performance.

Task Difficulty. Table 6 presents the performance of models trained with Condor data of varying difficulty levels. The results show that tasks of increasing difficulty contribute to greater performance improvements, with more challenging tasks yielding higher average scores. Additionally, combining all three difficulty levels during training further enhances performance, resulting in an average improvement of 0.34%.

4.5 Scaling of Condor Data

The performance of models within the Condor pipeline improves with respect to the increasing

---

[2]We apply the SFT to an internal version of InternLM2.5-7B, which is enhanced with instruction data focused on mathematics and coding.

Datasets	Qwen2.5-7B		Internlm2.5-7B		Llama3.1-8B	
	Official	Condor Refine	Official	Condor Refine	Official	Condor Refine
CompassArena	34.95	**43.62**	34.03	**42.92**	9.23	**20.48**
FoFo	0.45	**0.48**	0.40	**0.41**	0.37	**0.50**
AlignBenchv1.1	6.16	**6.26**	5.64	**5.79**	4.66	**5.29**
AlpacaEvalv2	32.42	**60.00**	26.34	**55.53**	24.1	**47.33**
ArenaHard	54.72	**61.37**	16.19	**36.02**	31.33	**45.73**
FollowBench	**0.86**	0.84	**0.81**	0.79	0.83	**0.84**
MTBench101	**8.37**	8.18	8.00	**8.07**	8.18	**8.19**
WildBench	15.69	**31.13**	-13.37	**17.71**	-2.24	**20.45**
Average	56.90	**63.27**	47.19	**56.44**	45.18	**55.24**

Table 4: **Performance Comparison across Different Types of Models.** Condor Refine shows strong adaptability over different types of models.

Datasets	Qwen2.5-7B		Qwen2.5-14B		Qwen2.5-32B	
	Official	Condor Refine	Official	Condor Refine	Official	Condor Refine
CompassArena	34.95	**43.62**	35.78	**55.05**	40.18	**60.18**
FoFo	0.45	**0.48**	0.55	**0.58**	0.59	**0.63**
AlignBenchv1.1	6.16	**6.26**	6.56	**6.66**	6.78	**6.88**
AlpacaEvalv2	32.42	**60.00**	33.66	**71.18**	35.53	**74.41**
ArenaHard	54.72	**61.37**	71.00	**74.95**	76.82	**86.83**
FollowBench	**0.86**	0.84	**0.92**	0.90	**0.94**	0.92
MTBench101	**8.37**	8.18	**8.43**	8.30	**8.49**	8.33
WildBench	15.69	**31.13**	23.42	**40.80**	22.46	**45.84**
Average	56.90	**63.27**	62.38	**71.03**	64.87	**75.07**

Table 5: **Performance Comparison across Different Model Sizes.** Qwen2.5-7B, 14B, and 32B all benefit from Condor Refine by a significant margin.

Different level	Exp. Easy	Exp. Medium	Exp. Hard	Exp. All
Easy Data	✓			✓
Medium Data		✓		✓
Hard Data			✓	✓
CompassArena	42.22	38.95	43.07	**43.62**
FoFo	0.45	0.46	0.47	**0.48**
AlignBenchv1.1	6.18	6.22	6.16	**6.26**
AlpacaEvalv2	58.76	56.89	58.51	**60.00**
ArenaHard	58.55	61.02	**61.81**	61.37
FollowBench	0.83	**0.86**	0.84	0.84
MTBench101	**8.18**	8.17	8.16	**8.18**
WildBench	26.98	29.41	**31.53**	31.13
Average Score	61.71	62.14	62.93	**63.27**

Table 6: **Ablations on Different Levels of Task Difficulty.** Harder tasks yield greater improvements.

amount of training data, tags, and tasks. To investigate these relationships in detail, we conduct comprehensive experiments, as outlined below.

Training Data Amount. We visualize the relative performance under different sampling proportions of training data in Figure 4. Surprisingly, we find that the model retains a high percentage of its performance even when trained on only a small proportion of the data. Additionally, while some performance fluctuations are observed on specific

benchmarks due to the relatively smaller data sizes – the overall average score consistently improves as the total amount of training data scales up.

Number of WKT Tags and Tasks. To better understand the impact of tags and tasks in the Condor pipeline, we conduct ablation experiments by incrementally increasing the proportions of Tasks and WKT Tags. As shown in Figure 5, while there are some fluctuations, the fitted average performance consistently improves with the increasing number of tags and tasks, indicating that both factors contribute positively to enhancing the model's human-preference capabilities. Furthermore, the influence of tasks is observed to be more significant than that of tags, as evidenced by the larger performance improvements associated with adding tasks. Detailed scores are provided in Tables 10 and 11.

4.6 Self Iteration

We conduct self-iteration experiments using Condor data generated by Qwen2.5-7B and Qwen2.5-72B to evaluate whether a single model can enhance its chat capabilities by itself. We train the Base model and then report human-preference per-
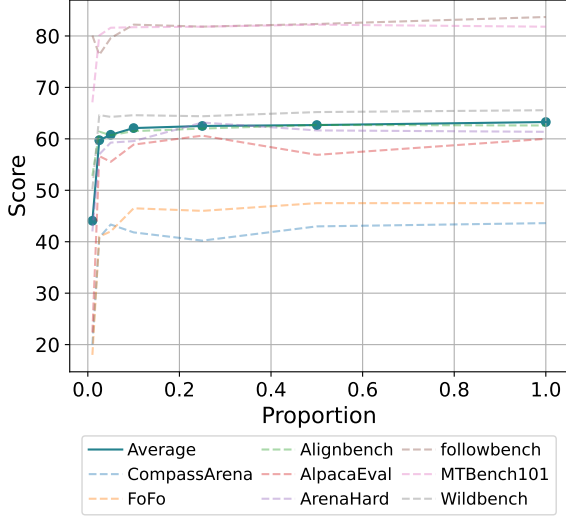
Figure 4: **Performance Comparison under Different Proportions of Training Data.** "Average" represents the average score across 8 benchmarks, and the results show a clear and consistent improvement as the size of the training data increases (Detailed scores in Table 9).
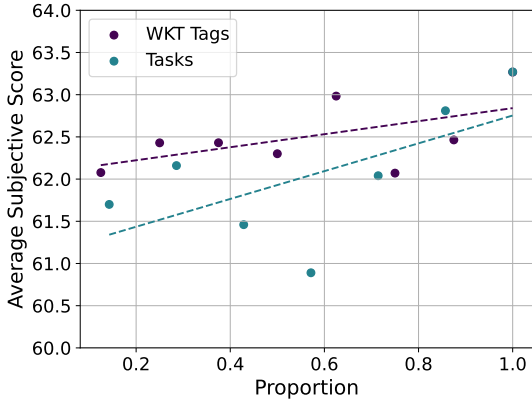

Figure 5: **Performance Results under Different Proportion of WKT Tags and defined Tasks.** Tasks have a more significant influence than tags, as evidenced by the significant drop with a small proportion of tasks.

formance scores across 8 benchmarks. As shown in Table 7, both the 7B and 72B models achieve self-improvement even compared with the RLHF models. Notably, the improvement for the 72B model is slightly greater than that of the 7B model, suggesting that larger, more capable models are better equipped to achieve self-improvement.

5 Analysis and Discussion

5.1 Which capability is improved the most?

We evaluate the scores for each benchmark across different sub-capability dimensions and observe improvements in sub-capabilities after Condor training (detailed information in Appendix A.2). As

Datasets	7B-IT	7B-Condor	72B-IT	72B-Condor
CompassArena	**33.80**	32.83	43.70	**48.73**
FoFo	0.52	0.45	**0.66**	0.60
AlignBenchv1.1	**6.22**	6.02	6.94	**6.98**
AlpacaEvalv2	34.66	**41.37**	47.64	**56.15**
ArenaHard	53.65	**60.39**	**85.67**	84.86
FollowBench	0.84	0.84	**0.92**	0.91
MTBench101	**8.60**	8.38	**8.45**	8.44
WildBench	16.71	**19.78**	45.23	**47.75**
Average	58.02	**58.38**	70.14	**71.12**

Table 7: **Condor Self Iteration Experiments on Qwen 7B and 72B.** Here, "IT" refers to the Instruct model after official RLHF. We use GPT-4o as the judge.
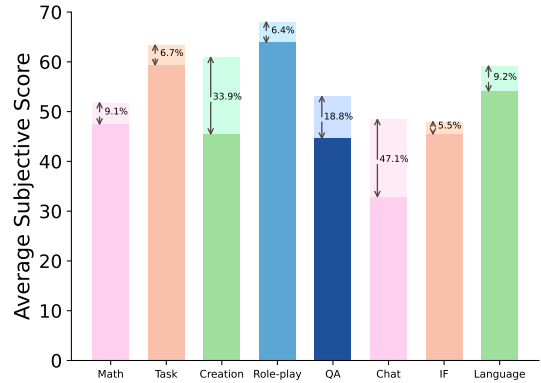

Figure 6: **Qwen2.5-7B Performance improvements on Various Sub-Dimensions.** Condor Refine consistently improves the model among all the sub-dimensions.

shown in Figure 6, the model demonstrates improvement across all sub-capabilities, with the most significant score increases observed in the tasks of Creation, QA, and Chat. These areas align closely with the core aspects of the model's human-preference capability, highlighting Condor's effectiveness in enhancing key conversational skills.

5.2 How large is the synthetic data coverage?

We extract the embeddings of questions from Condor and Magpie and use t-SNE for dimensionality reduction and visualization. To ensure a fair and balanced comparison, we randomly sample 200k questions from each dataset for this analysis. Note that according to Magpie's claimed method, it is capable of feedbacking the model's own training data, which means that the data distribution obtained by Magpie is somewhat close to the model's original SFT data. As shown in Figure 7, Condor has a similar or even broader distribution compared to Magpie, which indicates that the questions generated by Condor have good diversity.
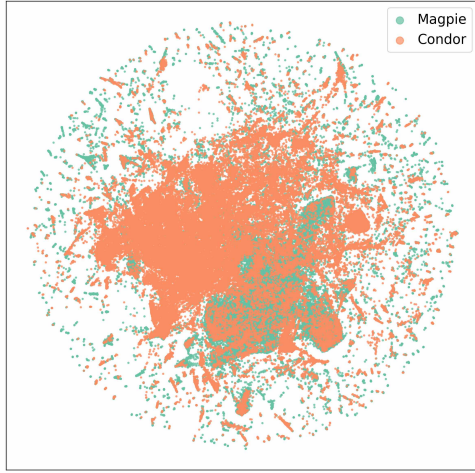
Figure 7: **t-SNE visualization of Condor and Magpie.** Condor shows great alignment with Magpie, highlighting its large diversity and generalization capabilities.

5.3 What does SFT actually enhance?

A comprehensive analysis of the experimental results reveals important insights, particularly from the knowledge-based evaluation results (Table 2). The findings indicate that the SFT phase has minimal impact on the model's intrinsic knowledge capabilities. This suggests that the vast majority of knowledge acquisition occurs during the pre-training phase. These observations suggest that the SFT phase of Condor contributes very little to enhancing the model's foundational knowledge. Instead, its primary focus is on refining the model's ability to utilize existing knowledge to answer questions effectively, which can be achieved with a relatively small amount of high-quality data.

6 Conclusion

In this paper, we propose Condor, a two-stage data synthesis engine to generate high quality data for supervised fine-tuning. Extensive experiments demonstrate that with the high quality data generated by Condor, the performance of the fine-tuned model surpasses many existing methods and the official RLHF models using a small amount of synthetic data. We also explore the scaling of synthetic data and self-iteration experiments, demonstrating that models can achieve self-iteration through synthetic data.

7 Limitations

Despite the significant improvements brought by Condor in human preference performance, there are still many experiments that require further exploration, such as the use of multi-round iterative synthetic data and how to further enhance the diversity of the synthetic data. Additionally, the hallucinations produced by LLMs in synthetic data could also become a potential risk. These issues need to be further addressed and improved, which will in turn enhance the quality of the synthetic data.

References

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

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Ge Bai, Jie Liu, Xingyuan Bu, Yancheng He, Jiaheng Liu, Zhanhui Zhou, Zhuoran Lin, Wenbo Su, Tiezheng Ge, Bo Zheng, et al. 2024. Mt-bench-101: A fine-grained benchmark for evaluating large language models in multi-turn dialogues. *arXiv preprint arXiv:2402.14762*.

Maosong Cao, Alexander Lam, Haodong Duan, Hongwei Liu, Songyang Zhang, and Kai Chen. 2024. Compassjudger-1: All-in-one judge model helps model evaluation and evolution. *arXiv preprint arXiv:2410.16256*.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *Preprint*, arXiv:1905.10044.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

OpenCompass Contributors. 2023a. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

XTuner Contributors. 2023b. Xtuner: A toolkit for efficiently fine-tuning llm. https://github.com/InternLM/xtuner.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. *Preprint*, arXiv:1903.00161.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. 2023. Reinforced self-training (rest) for language modeling. *Preprint*, arXiv:2308.08998.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *Preprint*, arXiv:2103.03874.

Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. Large language models can self-improve. *Preprint*, arXiv:2210.11610.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2023b. Followbench: A multi-level fine-grained constraints following benchmark for large language models. *arXiv preprint arXiv:2310.20410*.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. 2024. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. *arXiv preprint arXiv:2406.11939*.

Yiming Liang, Ge Zhang, Xingwei Qu, Tianyu Zheng, Jiawei Guo, Xinrun Du, Zhenzhu Yang, Jiaheng Liu, Chenghua Lin, Lei Ma, Wenhao Huang, and Jiajun Zhang. 2024. I-sheep: Self-alignment of llm from scratch through an iterative self-enhancement paradigm. *Preprint*, arXiv:2408.08072.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*.

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. 2024a. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Xiao Liu, Xuanyu Lei, Shengyuan Wang, Yue Huang, Zhuoer Feng, Bosi Wen, Jiale Cheng, Pei Ke, Yifan Xu, Weng Lam Tam, et al. 2023. Alignbench: Benchmarking chinese alignment of large language models. *arXiv preprint arXiv:2311.18743*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. Refiner: Reasoning feedback on intermediate representations. *Preprint*, arXiv:2304.01904.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

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. *Preprint*, arXiv:2311.12022.

Ming Shen. 2024. Rethinking data selection for supervised fine-tuning. *arXiv preprint arXiv:2402.06094*.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Shuo Tang, Xianghe Pang, Zexi Liu, Bohan Tang, Rui Ye, Xiaowen Dong, Yanfeng Wang, and Siheng Chen. 2024. Synthesizing post-training data for llms through multi-agent simulation. *arXiv preprint arXiv:2410.14251*.

Zhengwei Tao, Ting-En Lin, Xiancai Chen, Hangyu Li, Yuchuan Wu, Yongbin Li, Zhi Jin, Fei Huang, Dacheng Tao, and Jingren Zhou. 2024. A survey on self-evolution of large language models. *Preprint*, arXiv:2404.14387.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Ke Wang, Jiahui Zhu, Minjie Ren, Zeming Liu, Shiwei Li, Zongye Zhang, Chenkai Zhang, Xiaoyu Wu, Qiqi Zhan, Qingjie Liu, et al. 2024a. A survey on data synthesis and augmentation for large language models. *arXiv preprint arXiv:2410.12896*.

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. *Preprint*, arXiv:2212.10560.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024b. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *Preprint*, arXiv:2406.01574.

Congying Xia, Chen Xing, Jiangshu Du, Xinyi Yang, Yihao Feng, Ran Xu, Wenpeng Yin, and Caiming Xiong. 2024. Fofo: A benchmark to evaluate llms' format-following capability. *arXiv preprint arXiv:2402.18667*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *arXiv preprint arXiv:2406.08464*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Taolin Zhang, Sunan He, Tao Dai, Zhi Wang, Bin Chen, and Shu-Tao Xia. 2024a. Vision-language pre-training with object contrastive learning for 3d scene understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7296–7304.

Taolin Zhang, Jinpeng Wang, Hang Guo, Tao Dai, Bin Chen, and Shu-Tao Xia. 2024b. Boostadapter: Improving test-time adaptation via regional bootstrapping. *arXiv preprint arXiv:2410.15430*.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. 2024c. Evaluating the performance of large language models on gaokao benchmark. *Preprint*, arXiv:2305.12474.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *Preprint*, arXiv:2311.07911.

Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 2023. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2911–2921.

A   Appendix

A.1   Detailed Calculation Method for the Subjective Evaluation Scores.

Since the metrics for statistical scoring of various subjective chat datasets are not the same, and the scoring ranges are not all 0-100, we use the following mapping relationships when calculating the average subjective chat score for the model:

- The scoring ranges for CompassArena, AlpacaEvalv2, and ArenaHard are 0-100, and no special treatment is needed when calculating the overall mean score.

- For FoFo and Followbench, the scoring range is 0-1, and we multiply by 100 when calculating the overall mean score.

- For AlignBenchv1.1 and MTBench101, the scoring range is 0-10, and we multiply by 10 when calculating the overall mean score.

- For WildBench, the scoring range is -100 to 100, and we add 100 to the score and then divide by 2 for mapping.

- The final calculated total Average score is the mean score of these datasets after they have been mapped to the 0-100 range.

A.2   Detailed Information for Sub-capabilities Improvement of Condor

Based on the subscore for each capability dimension provided by each dataset, we aggregated the model's scores by dimension, thereby obtaining the model's scores on each capability dimension across these subjective test datasets. Specifically, according to the different subscores for various capability dimensions provided by different datasets, we aggregated the scores in the following manner:

- Math: The math and reasoning capabilities of model, we aggregate the sub-score from AlignBench, CompassArena.

- Task: The task problem svoling capability of model, we aggregate from AlignBench, ArenaHard and MTBench101.

- Creation: The ability of the model to create various types of content as required is aggregated from AlignBench, CompassArena, MTBench101, and WildBench.

- Role-play: The role-playing capability of the model is aggregated from AlignBench, AlpacaEval, and WildBench.

- QA: The knowledge-based question-answering capability of the model is aggregated from AlignBench and CompassArena.

- Chat: The daily chat capability of the model is aggregated from AlignBench, AlpacaEval, ArenaHard, MTBench101, and WildBench.

- IF: The instruction following capability of the model is aggregated from FoFo and FollowBench.

- Language: The language understanding and processing capability of the model is aggregated from AlignBench and CompassArena.

Datasets	Qwen2.5-7B-Instruct	Qwen2.5-7B-Base		Qwen2.5-7B-Instruct	
	Official	Condor Void	Condor Refine	Condor Void	Condor Refine
CompassArena	34.95	33.78	43.62	38.17	**46.68**
FoFo	0.45	0.38	0.48	**0.50**	0.48
AlignBenchv1.1	6.16	6.08	6.26	**6.32**	**6.32**
AlpacaEvalv2	32.42	33.04	60.00	43.98	**63.85**
ArenaHard	54.72	43.35	61.37	57.33	**63.30**
FollowBench	**0.86**	**0.86**	0.84	0.83	0.85
MTBench101	**8.37**	8.12	8.18	8.28	8.24
WildBench	15.69	13.82	31.13	18.52	**32.99**
Average	56.9	54.01	63.27	59.67	**64.85**

Table 8: **Detailed Results for Condor on Subjective Evaluation Datasets Judged by CompassJudger-1-32B.**

Proportion Datasets	1%	2.5%	5%	10%	25%	50%	100%
CompassArena	20.02	40.80	43.35	41.82	40.22	42.98	**43.62**
FoFo	0.18	0.41	0.42	0.47	0.46	**0.48**	**0.48**
AlignBenchv1.1	5.28	6.14	6.07	6.15	6.20	**6.27**	6.26
AlpacaEvalv2	22.36	56.65	55.53	58.88	**60.62**	56.89	60.00
ArenaHard	42.00	56.99	59.27	59.56	**63.17**	61.64	61.37
FollowBench	0.80	0.76	0.80	0.82	0.82	0.82	**0.84**
MTBench101	6.71	8.01	8.16	8.17	8.18	**8.22**	8.18
WildBench	0.44	29.23	28.54	29.2	28.79	30.41	**31.13**
Average	44.07	59.74	60.79	62.09	62.50	62.68	**63.27**

Table 9: **Performance Comparison under Different Proportions of Training Data.** Here, 100% indicates the use of the full dataset, which is approximately 200K synthetic data for training, while the other proportions involve random sampling of the full dataset based on the given percentage.

Proportion / Datasets	12.50%	25%	37.50%	50%	62.50%	75%	87.50%	100%
CompassArena	42.57	40.98	41.87	42.25	42.88	41.75	43.5	**43.62**
FoFo	0.45	0.47	0.49	0.48	**0.50**	0.46	0.48	0.48
AlignBenchv1.1	6.14	6.18	6.18	6.24	6.25	6.19	6.14	**6.26**
AlpacaEvalv2	**60.12**	55.65	55.03	56.89	56.77	57.76	58.14	60.00
ArenaHard	62.19	**62.43**	60.81	60.01	59.73	60.11	60.99	61.37
FollowBench	0.80	0.86	0.85	0.83	**0.87**	0.83	0.81	0.84
MTBench101	8.18	**8.20**	8.15	8.17	8.14	8.18	**8.20**	8.18
WildBench	28.83	27.96	29.70	28.97	27.15	28.89	29.25	**31.13**
Average	62.08	62.43	62.43	62.3	62.98	62.07	62.46	**63.27**

Table 10: **Performance of the model after training with different numbers of tags in the training data.** The total number of Chinese tags is 4249, and the total number of English tags is 4296. We randomly sampled the Chinese and English tags in proportion and then combined the sampled Chinese and English data for training to ensure a balance of tags between the two languages.

Task Number / Datasets	1	2	3	4	5	6	7
CompassArena	42.83	42.62	41.1	42.63	42.85	**43.88**	43.62
FoFo	0.42	0.42	0.44	0.44	**0.49**	0.46	0.48
AlignBenchv1.1	6.18	**6.28**	6.24	6.15	6.20	6.16	6.26
AlpacaEvalv2	56.65	58.26	59.75	56.65	57.64	58.51	**60.00**
ArenaHard	62.6	59.40	60.86	60.61	61.60	**62.91**	61.37
FollowBench	0.81	0.83	0.76	0.75	0.78	0.82	**0.84**
MTBench101	8.11	**8.24**	8.21	8.18	8.15	8.19	8.18
WildBench	33.07	**33.69**	31.38	31.38	29.33	30.94	31.13
Average	61.7	62.16	61.46	60.89	62.04	62.81	**63.27**

Table 11: **Performance of the model after training with different numbers of tasks in the training data.** All the results in the table were obtained by testing the Qwen2.5-7B-Base model after training on the relevant datasets, with the number of tasks increasing by inclusion. The first task is Role-Playing, the second task is Daily Chat, the third task is Domain Knowledge Q&A, the fourth task is Given Material Processing, the fifth task is Response Format Control, the sixth task is View, and the seventh task is Creation.

Prompt for Question Synthesis

Now we need to create high-quality SFT data for LLM training, so we need you to produce a batch of such data. You only need to create Questions. I will give you a theme and some examples of SFT data Questions. You need to create three Questions of different difficulty levels based on this new theme.

Your Questions must meet the following requirements:

1. You must strictly create only three Questions at a time. These three Questions must be in the domain of [[domain]] and the Questions should align with the given theme of [[theme]].

2. The Questions you create must have context and sufficient information; they should not be abrupt and directly ask the question.

3. Your reply must strictly follow the format below. Your Questions need to be included between [Question Start] and [Question End], and the difficulty level should be indicated at the beginning, as in the following format:

[Easy][Question Start]Question[Question End]
[Medium][Question Start]Question[Question End]
[Hard][Question Start]Question[Question End]

4. Your Questions of different difficulty levels should be distinct and actually reflect the different levels of difficulty.

Here are some examples of high-quality SFT data Questions for [[domain]]: [example list]

Now it's your turn. Please provide the three Questions of different difficulty levels you created about the theme of [[theme]] for [[domain]], according to the requirements. Do not be confined by the theme; your Questions only need to be related to the theme. You can use your rich imagination, but note that you cannot copy the expression from the examples; you must have your own new expression:

Prompt for Response Critic

There is now a user's question and a model's response. You need to write a critique for this response, pointing out the strengths and weaknesses of the model's answer to help the model improve its response.

Your critique must strictly adhere to the following format:

[Critique Start]
[Strength Start]Strength[Strength End]
[Weakness Start]Weakness[Weakness End]
[Suggestion Start]Suggestion[Suggestion End]
[Critique End]

That is, you must place the strength, weakness, and suggestion in their respective sections.

Here is the user's question and the model's response: [dialogue]

Now it's your turn. Please provide your Critique as required:

I was watching a documentary about mapping ancient cities, and it got me thinking about how maps have evolved over time. Have you ever seen an old map and marveled at how different it looks from today's maps? What's the oldest map you've ever seen?

As a digital AI assistant, I don't have personal experiences, emotions, or visual interactions, so I haven't "seen" maps in the classical sense. I don't have a physical presence, and my interactions are limited to text-based conversations. However, I'd be delighted to take you on a journey through the evolution of maps and highlight some notable examples of ancient and historic maps!

The Evolution of Maps
1. Prehistory (20,000 BCE - 3,000 BCE): Early maps were often simple, symbolic, and focused on specific areas, like the Turin Papyrus Map (1160 BCE), which depicts the location of gold mines in ancient Egypt.
2. Ancient Civilizations (3,000 BCE - 500 CE): - Babylonian Map of the World (600 BCE): A clay tablet showing the world as a flat disk, surrounded by an ocean. - Ptolemy's World Map (150 CE): A more accurate, yet still flawed, representation of the known world, using longitude and latitude.
3. Medieval Period (500 - 1500 CE): - Mappa Mundi (1300 CE): A symbolic, Christian-centric map, often displaying Jerusalem at the center.
4. Age of Exploration (1500 - 1700 CE): - Waldseemüller Map (1507 CE): One of the earliest maps to depict the New World, naming America after Amerigo Vespucci.
5. Modern Cartography (1700 CE - present):
- Triangulation and Surveying: Led to more accurate mapping of territories.
- Aerial Photography and Satellite Imaging (20th century): Revolutionized mapmaking with high-resolution, globally consistent data.
- Digital Maps (late 20th century - present): Enabled interactive, dynamically updated, and highly accurate maps, like Google Maps.

Some of the Oldest Known Maps
- Cave of El Castillo Map (14,000 BCE): A simple, prehistoric map etched into a cave wall in Spain, showing the location of a nearby water source.
- The Bedolina Map (5,000 BCE): A rock carving in Italy, depicting a valley with topographical features.
- The Nubian Map (2,000 BCE): An ancient Egyptian map, showing the location of Nubian gold mines and territories.

Marveling at Old Maps
While I may not have personal experiences, I can certainly help you appreciate the historical context, artistic value, and cartographic innovations of old maps. If you'd like to discuss a specific map or era in more detail, feel free to ask!

Which aspect of old maps fascinates you the most? Is it:
•A) The artistic and symbolic representations
•B) The historical context and cultural influences
•C) The cartographic innovations and technological advancements
•D) Something else (please share!)

Condor Refine Example - Reflection

[Strength Start]
Strengths:
1.Comprehensive Response: The model provides an exhaustive overview of the evolution of maps, covering various historical periods, notable examples, and technological advancements.
2.Structured Formatting: The use of clear headings, numbered lists, and concise bullet points enhances readability and facilitates easy understanding of complex information.
3.Engagement Initiator: The response ends with an interactive element, encouraging user participation and potentially leading to a more in-depth discussion.
4.Honesty About Limitations: The model transparently acknowledges its limitations as a digital AI assistant, setting clear expectations for the user.
[Strength End]

[Weakness Start]
Weaknesses:
1.Length and Information Overload: The response is lengthy and packed with numerous examples, which might overwhelm the user. Some points could be elaborated upon in subsequent interactions rather than all at once.
2.Lack of Visual Enhancement: Despite discussing maps, the response is text-only. Incorporating images, diagrams, or even suggesting external visual resources could significantly enhance the user's understanding and engagement.
3.Initial Acknowledgement Could Be Brief: While honesty about the model's limitations is appreciated, the initial acknowledgement could be more concise to quickly transition to the more engaging and informative sections.
4.Question Response Mismatch: The user inquired about the oldest map the model has "seen," which the model addresses by stating its inability to see. However, the model could more directly address the implicit curiosity about old maps by initially highlighting one or two of the oldest known maps before delving into the broader evolution.
[Weakness End]

[Suggestion Start]
Suggestions for Improvement:
1.Tiered Information Disclosure: Initially provide a brief overview of the evolution of maps and highlight 2-3 of the oldest known maps. Offer the user the option to explore specific eras or types of maps in more detail, facilitating a more interactive and paced information exchange.
2.Incorporate Visual Aids or References: Suggest reputable online resources or include descriptions that encourage users to visually explore the mentioned maps, enhancing their understanding of cartographic developments.
3.Refine the Initial Limitation Disclosure: Condense the initial acknowledgement to a single sentence, swiftly moving the focus to the engaging content (e.g., "As a text-based AI, I'll guide you through the fascinating evolution of maps, highlighting notable examples and innovations.").
[Suggestion End]

Condor Refine Example - Refined Answer

Discovering the Ancient Roots of Modern Maps

As a text-based AI, I'll guide you through the fascinating evolution of maps, highlighting notable examples and innovations.

Let's Start with the Oldest Known Maps:
1.Cave of El Castillo Map (14,000 BCE):
Etched into a Spanish cave wall, this simple map shows the location of a nearby water source.
2.The Bedolina Map (5,000 BCE):
A rock carving in Italy, depicting a valley with topographical features.

Visual Exploration Suggestion: Visit the British Museum's Online Collection or Google Arts & Culture to see high-resolution images of these ancient maps.**

Curious About the Evolution of Maps? Here's a concise overview. Feel free to pick any era to explore in more depth:
- Prehistory (20,000 BCE - 3,000 BCE): Simple, symbolic maps (e.g., Turin Papyrus Map)
- Ancient Civilizations (3,000 BCE - 500 CE): Notable examples include Babylonian Map of the World and Ptolemy's World Map
- Medieval to Modern (500 - present): Advances from Mappa Mundi to Triangulation, Aerial Photography, and Digital Maps like Google Maps

Which Era or Aspect of Old Maps Would You Like to Explore Further?
•A) Ancient symbolic representations
•B) Medieval navigational charts
•C) The leap to digital cartography
•D) Something specific (please share, and I'll delve deeper!)

Detailed explanation of total 7 chat tasks

[Task 1-Role Playing]
Engage in simulated conversations or behaviors by assuming different roles to explore various perspectives or scenarios.

[Task 2-Daily Chat]
Participate in casual conversations, including greetings, small talk, and sharing everyday experiences.

[Task 3-Domain QA]
Provide specialized and accurate answers to questions within a specific domain or field.

[Task 4-Given Material Processing]
Analyze, process, and summarize given texts, data, or other materials.

[Task 5-Response Format Control]
Adjust the format, style, and expression of responses according to specified requirements.

[Task 6-View]
Offer personal opinions, insights, or perspectives on a particular topic.

[Task 7-Creation]
Generate new content such as articles, stories, poetry, designs, etc.

Is the ATF a violation in of itself against the American people?

The term "ATF" typically refers to the Bureau of Alcohol, Tobacco, Firearms and Explosives, a federal agency within the United States Department of Justice. The ATF is responsible for reducing crime and preventing violence by reducing the illegal flow and improper use of firearms, alcohol, tobacco, and explosives.

The ATF itself is not a violation against the American people. Rather, it is an agency that operates under the laws and regulations established by the U.S. government to enforce laws related to alcohol, tobacco, firearms, and explosives. The agency's actions are generally aimed at preventing crime and ensuring public safety.

However, like any government agency, the ATF can be subject to criticism and controversy. Some criticisms have been raised regarding the agency's enforcement practices, the impact of its regulations on law-abiding citizens, and concerns about civil liberties. These criticisms do not imply that the agency itself is a violation, but rather that its actions or policies may be subject to scrutiny and debate.

If you have specific concerns or questions about the ATF's actions or policies, it would be helpful to provide more context or details so that a more accurate and informed response can be given.

The term "ATF" typically refers to the Bureau of Alcohol, Tobacco, Firearms and Explosives, a federal agency in the United States responsible for enforcing federal laws related to the manufacture, possession, and use of alcohol, tobacco, firearms, and explosives. The ATF is not a violation against the American people; rather, it is an agency that operates within the legal framework established by the U.S. Congress to enforce specific laws.

However, the ATF's operations and the laws it enforces can be subjects of debate and criticism. For example, some argue that the agency's enforcement of gun control laws infringes on the Second Amendment rights of law-abiding citizens. Others may criticize the agency for its handling of certain cases or its budget and resource allocation. Additionally, there have been instances where the ATF has been involved in controversial operations, such as the Ruby Ridge and Waco incidents, which have fueled public debate and criticism.

It's important to note that the ATF's role is to enforce the laws as they are written, and the laws themselves are subject to interpretation and debate. The agency's actions are generally aimed at reducing crime and ensuring public safety, but the effectiveness and fairness of these actions are often subjects of public and political discourse.

If you have specific concerns or questions about the ATF's operations or the laws it enforces, it might be helpful to look at the broader context of U.S. law and policy, including the legislative history and the ongoing debates surrounding gun control and public safety. For more detailed information, you can refer to official ATF publications, such as the ATF's annual reports, or consult reputable news sources and legal analyses."