WHAT MAKES GOOD DATA FOR ALIGNMENT? A COMPREHENSIVE STUDY OF AUTOMATIC DATA SELECTION IN INSTRUCTION TUNING

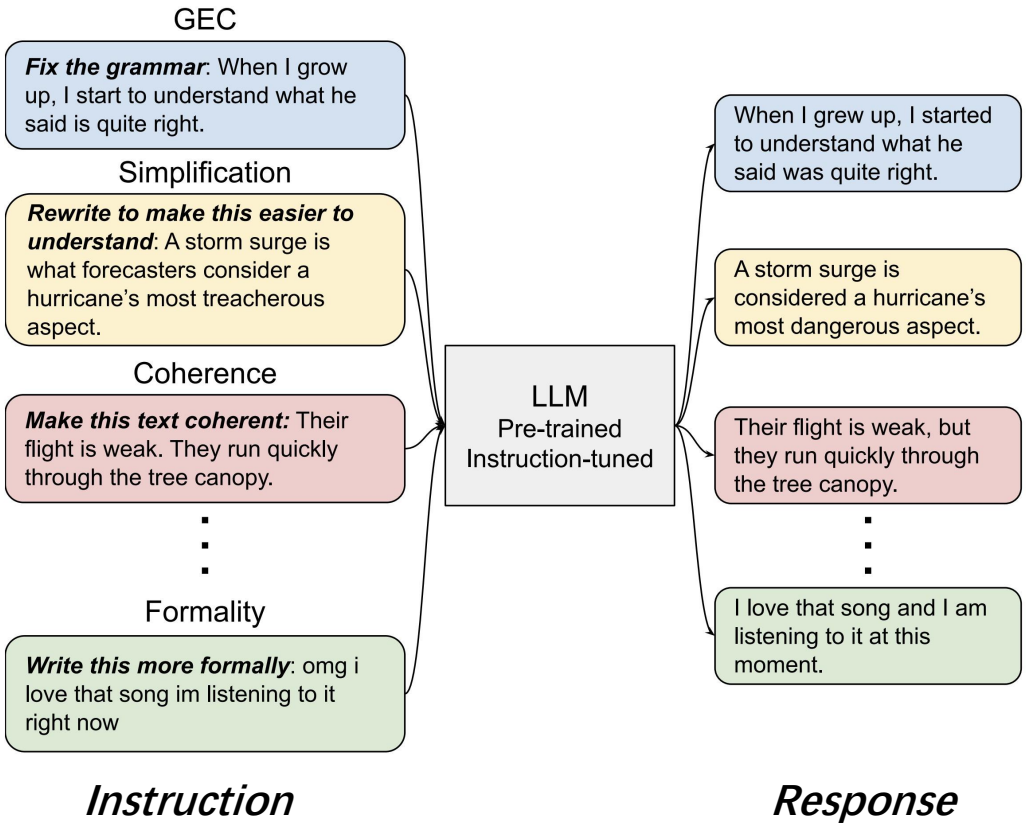
Wei Liu[*1] **Weihao Zeng**[*2] **Keqing He**[3] **Yong Jiang**[4] **Junxian He**[5]
[1]ShanghaiTech University [2]Beijing University of Posts and Telecommunications
[3]Meituan [4]Alibaba Group [5]The Hong Kong University of Science and Technology
liuwei4@shanghaitech.edu.cn zengwh@bupt.edu.cn
junxianh@cse.ust.hk

ICLR 2024

Introduction

Motivation: A relatively small high-quality dataset has been shown to be sufficient to align LLMs well[1-3]. ***How to systematically curate an effective dataset that ensures competitive performance with the least amount of data?***

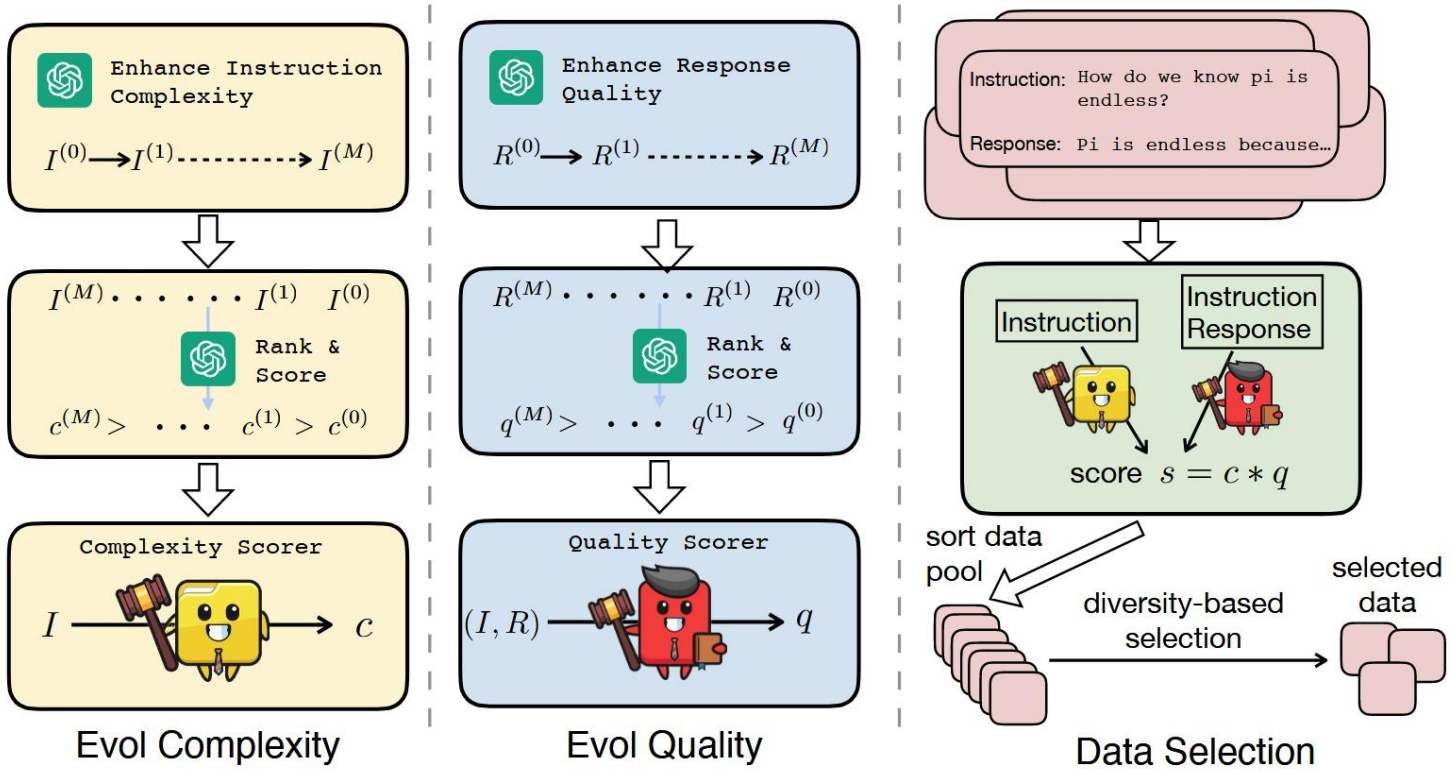
The authors explore various methods to quantitatively assess data examples from three key dimensions: **complexity, quality, and diversity**. They present **DEITA** (**D**ata-**E**fficient **I**nstruction **T**uning for **A**lignment), a series of models fine-tuned from LLaMA and Mistral models using data samples automatically selected with the proposed approach.
Empirically, DEITA performs better or on par with the SOTA open-source alignment models with only **6K** training data samples.



1. Wang, Yizhong, et al. "Self-instruct: Aligning language models with self-generated instructions." arXiv preprint arXiv:2212.10560 (2022).
2. Zhou, Chunting, et al. "Lima: Less is more for alignment." Advances in Neural Information Processing Systems 36 (2024).
3. Lu, Keming, et al. "# instag: Instruction tagging for analyzing supervised fine-tuning of large language models." The Twelfth International Conference on Learning Representations. 2023.

Overview

Drawing inspiration from WizardLM[1], the authors measure data from 3 dimensions: complexity, quality, and diversity. For **Evol Complexity** and **Evol Quality**, they first collect samples with varying complexities or qualities through adopting an evolution-based approach, then ask **ChatGPT** (refering to gpt-3.5-turbo-0613) to rank and score the variants of the same data sample for a small seed dataset, and **train the complexity and quality scorers**.
In the last step, they use **the cosine distance** as the diversity metric to select the "good" data samples.



1. Xu, Can, et al. "Wizardlm: Empowering large language models to follow complex instructions." arXiv preprint arXiv:2304.12244 (2023).

Setup

Given a large instruction tuning data pool, $X = \{x_1, x_2, \cdots, x_n\}$, where x_i represents an individual data sample in the form of an *instruction-response* pair. We aim to select a subset $S_\pi^{(m)}$ of size m from X, using a selection strategy denoted by π. And we denote the alignment performance after instruction-tuning as Q, the optimal data selection strategy π^* satisfies:

$$\pi^* = \arg\max_{\pi} Q(S_\pi^{(m)}).$$

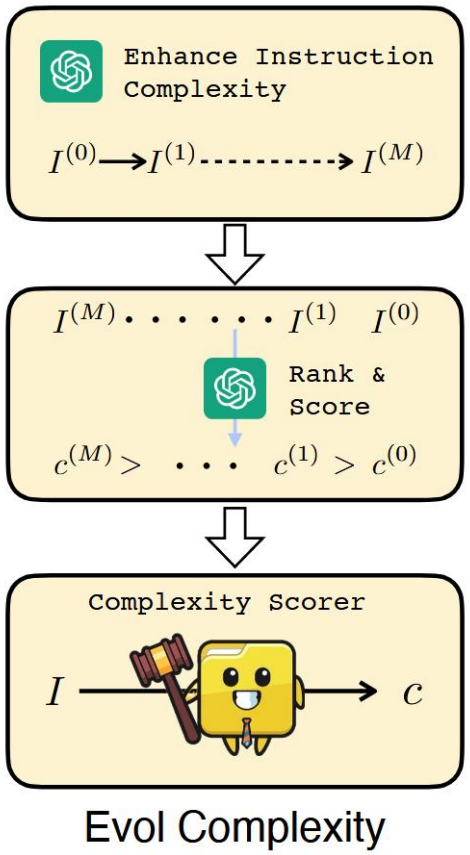
- X_{sota} is constructed by the training datasets of the state-of-the-art aligned LLMs. This represents the setting where a data pool that is relatively complex, diverse, and of high-quality is available.
- X_{base} is overall lower-quality and redundant.

The authors fine-tune LLaMA-1 13B on the sampled dataset with size of 6K, and perform controlled studies on a single metric to evaluate data at a time. MT-Bench is utilized to evaluate alignment performance.

Data Pool	Dataset Source	Sample Size
X_{sota}	ShareGPT	58 K
	UltraChat	105 K
	WizardLM	143 K
X_{base}	Alpaca	52 K
	Dolly	15 K
	OAssit	10 K
	FLAN 2022	23 K

Table 1: Statistics of data pools X_{sota} and X_{base}. The Dataset Source indicates the source of the data used for sampling. The Sample Size represents the number of samples in the respective dataset.

From The Complexity Perspective – Evol Complexity



Evol Complexity

	Prompt Templates
concretizing	I want you act as a Prompt Rewriter. Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., ChatGPT and GPT4) a bit harder to handle. But the rewritten prompt must be reasonable and must be understood and responded by humans. Your rewriting cannot omit the non-text parts such as the table and code in #Given Prompt#:. Also, please do not omit the input in #Given Prompt#. You SHOULD complicate the given prompt using the following method: Please replace general concepts with more specific concepts. or You should try your best not to make the #Rewritten Prompt# become verbose, #Rewritten Prompt# can only add 10 to 20 words into #Given Prompt#. '#Given Prompt#', '#Rewritten Prompt#', 'given prompt' and 'rewritten prompt' are not allowed to appear in #Rewritten Prompt# #Given Prompt#: <Here is instruction> #Rewritten Prompt#:

	Prompt Templates
Rank & Score	Ranking the following questions according to the difficulty and complexity. Score 1-5. You can give a score of 6 if the question is too complex for you to answer it. You should respond with the format:\n [1] Score: 1\n [2] Score: 2\n [1] <Instruction 1> [2] <Instruction 2> [3] <Instruction 3> [4] <Instruction 4> [5] <Instruction 5>

1. Use 2K examples randomly sampled from the Alpaca as the seed dataset.
2. Use the In-Depth Evolving Prompt from WizardLM to enhance the complexity through techniques such as adding constraints, deepening, concretizing and increasing reasoning steps.
3. Use the scores to train a LLaMA-1 7B model as the complexity scorer.

From The Complexity Perspective – Evol Complexity

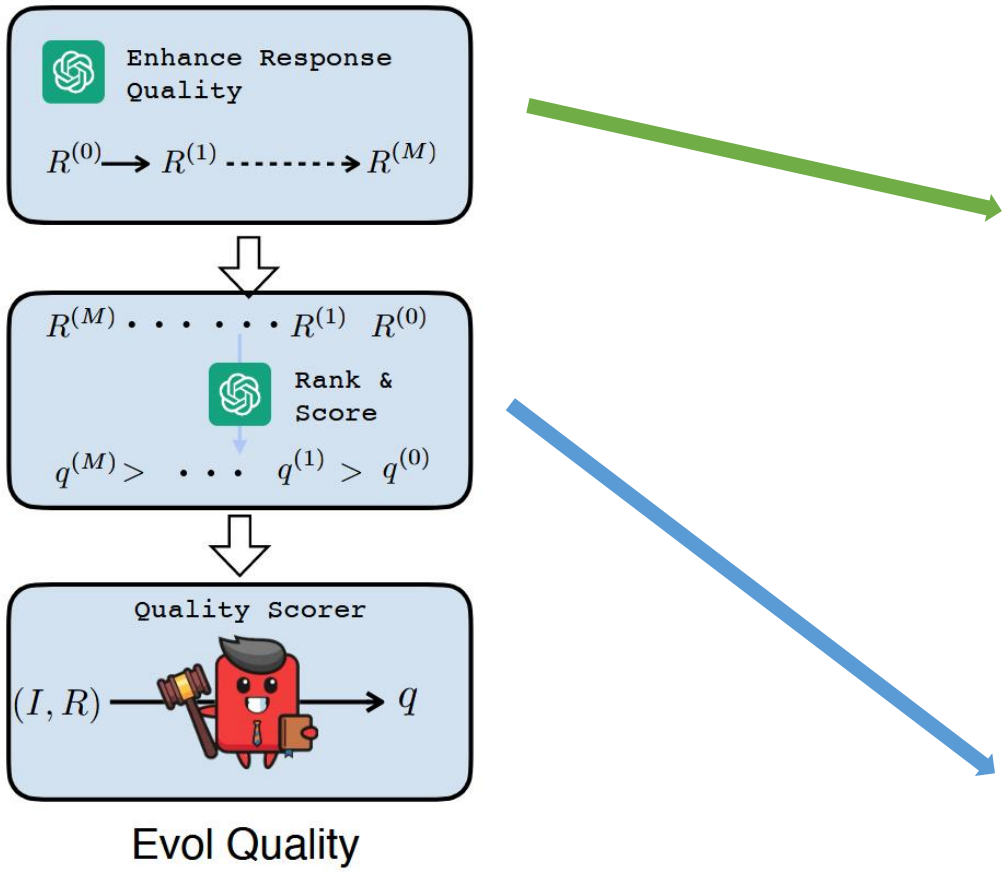
Model	X_{sota}	X_{base}
Random Selection	5.84	4.93
Instruction Length	5.89	4.00
Perplexity	4.06	1.89
IFD	5.91	2.46
Instag Complexity	6.18	4.98
Direct Scoring (Pool=50K)	5.16	4.87
Instruction Node (Pool=50K)	5.65	4.82
EVOL COMPLEXITY (Pool=50K)	5.73	5.29
EVOL COMPLEXITY	**6.27**	**5.57**

Table 2: MT-bench of different complexity metrics. All methods select 6K samples. "Pool=50K" denotes the data selection is conducted in a 50K-sized subset due to the cost of using ChatGPT to annotate the entire pool. We include the results of our method on the 50K data pool to make a fair comparison with baselines.

	Prompt Template
Direct Scoring	We would like you to evaluate and rate the difficulty and complexity of the following question. You should give an overall score on a scale of 1 to 10, where a higher score indicates higher difficulty and complexity. You must just give a score without any other reasons. Question: <Instruction> Score:
Instruction Node	You need to rewrite the following "instruction" to a TREE through Semantic Parsin in the natural language processing field and only count the total node number of the TREE. You must just give node number without any other reasons. Instruction: <Instruction> Node number:

EVOL COMPLEXITY achieves superior performance on both datasets, indicating strong robustness across different dataset pools.

From The Quality Perspective – Evol Quality



Evol Quality

	Prompt Templates
enhancing helpfulness	I want you to act as a Response Rewriter Your goal is to enhance the quality of the response given by an AI assistant to the #Given Prompt# through rewriting. But the rewritten response must be reasonable and must be understood by humans. Your rewriting cannot omit the non-text parts such as the table and code in #Given Prompt# and #Given Response#. Also, please do not omit the input in #Given Prompt#. You Should enhance the quality of the response using the following method: Please make the Response more helpful to the user. You should try your best not to make the #Rewritten Response# become verbose, #Rewritten Response# can only add 10 to 20 words into #Given Response#. '#Given Response#', '#Rewritten Response#', 'given response' and 'rewritten response' are not allowed to appear in #Rewritten Response# #Given Prompt#: Give three tips for staying healthy. #Given Response#: <Response> #Rewritten Response#:

	Prompt Templates
Rank & Score	Rank the following responses provided by different AI assistants to the user's question according to the quality of their response. Score each response from 1 to 5, with 6 reserved for responses that are already very well written and cannot be improved further. Your evaluation should consider factors such as helpfulness, relevance, accuracy, depth, creativity, and level of detail of the response. Use the following format: [Response 1] Score: [Response 2] Score: #Question#: <Instruction> #Response List#: [Response 1] <Response 1> [Response 2] <Response 2> [Response 3] <Response 3> [Response 4] <Response 4> [Response 5] <Response 5>

Same as the Evol Complexity.

From The Quality Perspective – Evol Quality

Model	X_{sota}	X_{base}
Random Selection	5.84	4.93
Response Length	5.94	5.65
Direct Scoring (Pool=50K)	5.61	4.44
EVOL QUALITY (Pool=50K)	5.85	5.29
EVOL QUALITY	6.19	5.67

Table 3: MT-bench of different quality measurements. All methods select 6K samples for training. "Pool=50K" denotes the data selection procedure is conducted in a 50K-sized subset due to the cost of ChatGPT to annotate the entire pool. We include the results of our method on the 50K data pool to make a fair comparison with the baselines.

	Prompt Templates
Direct Scoring	We would like to request your feedback on the performance of AI assistant in response to the given question displayed following. ##Tips:Please rate according to the accuracy of the response to the instruction and the input. Each assistant receives a score on a scale of 0 to 5, where a higher score indicates higher level of the accuracy. You must just give a score without any other reasons. ##Question: <Instruction> ##Response: <Response> ##Score:

The proposed EVOL QUALITY approach consistently exhibits superior alignment performance.

From The Diversity Perspective – An Embedding-based Approach

Repr Filter: Filter data by calculating the embedding distance (cosine distance) between sample x_i and samples in the selected dataset S to enhance dataset diversity.

1. Sort the data pool by complexity and quality scores.
2. Use the LLaMA-1 13B model to encode sentences and compute cosine distance d.
3. Set the threshold $\tau = 0.9$. If $d < \tau$, add the sample to set S.
4. Filter samples one by one until S reaches the budget m.

Model	X_{sota}	X_{base}
Random Selection	5.82	4.34
Instag Diversity	6.10	4.46
Repr Filter	6.17	4.68

Table 4: MT-bench scores of different diversity measurements. All methods select 6K samples for instruction tuning.

Deita– Data Efficient Instruction Tuning For Alignment

Algorithm 1 Score-First, Diversity-Aware Data Selection

Input: The data pool X, data budget m
Output: The selected subset $S^{(m)}_{\pi_{\text{DEITA}}}$
Initialize Empty Dataset $S^{(m)}_{\pi_{\text{DEITA}}}$
Sorting X with the combined complexity score and quality score $s = q * c$;
Getting the sorted Pool X^*;
for Each Sample $x \in X^*$ **do**
    // $d(x, S)$ denotes the distance between x and its nearest neighbor in S
    if $d(x, S^{(m)}_{\pi_{\text{DEITA}}}) < \tau$ **then**
        $S^{(m)}_{\pi_{\text{DEITA}}} \leftarrow S^{(m)}_{\pi_{\text{DEITA}}} \cup \{x\}$
    else
        Continue
    end if
    $X \leftarrow X \setminus \{x\}$
    if $|S^{(m)}_{\pi_{\text{DEITA}}}|$ equals to m **then**
        Break
    end if
end for

Results

Model	Data Size / Alignment	MT-Bench	AlpacaEval(%)
Proprietary Models			
GPT-4	–	8.99	95.28
Claude-v2	–	8.06	91.36
gpt-3.5-turbo	–	7.90	89.37
Open-sourced Models based on LLaMA-1-13B			
Alpaca-13B	52K / SFT	4.53	–
WizardLM-13B	70K / SFT	6.35	75.31
Vicuna-13B-v1.3	125K / SFT	6.39	**82.11**
TAGLM-13B†	6K / SFT	6.09	72.80
Random-Select	10K / SFT	6.03	71.52
DEITA-LLaMA1-13B$_{6K}$	6K / SFT	6.46	77.08
DEITA-LLaMA1-13B$_{10K}$	10K / SFT	**6.60**	78.01
Open-sourced Models based on LLaMA-2-13B			
LLaMA2-13B-Chat	>100K / SFT + >1M / RLHF	6.65	81.09
Vicuna-13B-v1.5	125K / SFT	6.57	78.80
TÜLÜ 2 13B	326K / SFT	6.70	78.90
TÜLÜ 2 + DPO 13B	326K / SFT + 60K / DPO	7.00	89.50
Random-Select	10K / SFT	5.78	65.19
DEITA-LLaMA2-13B$_{6K}$	6K / SFT	6.65	80.75
DEITA-LLaMA2-13B$_{10K}$	10K / SFT	**6.79**	**81.09**
Open-sourced Models based on Mistral-7B			
Mistral-7B-Instruct-v0.1	–	6.84	69.65
Mistral-7B-Instruct-v0.2	–	7.60	93.65
zephyr-beta-sft*	200K / SFT	5.32	75.12
zephyr-beta	200K / SFT + 60K / DPO	7.34	90.60
Random-Select	10K / SFT	5.89	56.90
DEITA-Mistral-7B$_{6K}$	6K / SFT	7.22	80.78
DEITA-Mistral-7B$_{10K}$	10K / SFT	**7.32**	**81.67**
DEITA-Mistral-7B$_{6K}$ + DPO	6K / SFT +10K / DPO	7.55	90.06

Table 6: Results of different instruction-tuned models on MT-Bench and AlpacaEval. Best SFT-only numbers within the same base model are bolded, while the overall best numbers are underlined. † denotes the results obtained by using their released LLaMA-7B tagger model for a fair comparison. Zephyr-beta-sft* is the official checkpoint after the phase of supervised fine-tuning (SFT). We notice the performance of this checkpoint is lower than expected. We speculate the reason is that this checkpoint is not the best SFT checkpoint reported in their paper since the checkpoint is used for further DPO training.

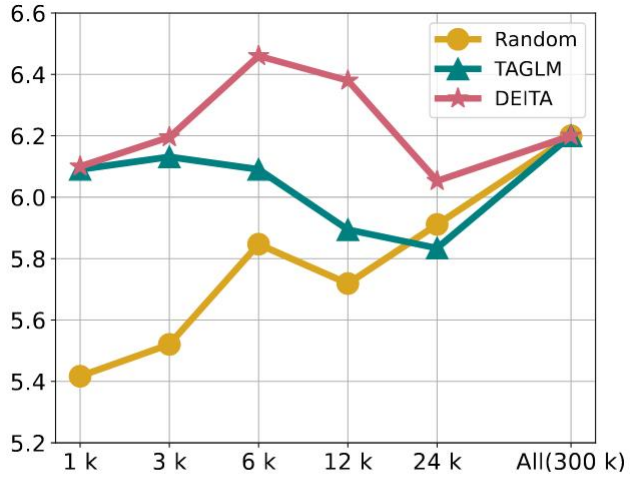


Figure 2: Data-scaling results on MT-Bench. The X-axis represents the # samples used.

For DPO training, the authors randomly sample 10K comparison data pairs used in Zephyr that is originally obtained from the **UltraFeedback** dataset.

Results

Model	Data Size	MT-Bench	AlpacaEval(%)
Random	6K	5.84	73.91
Alpagasus (Pool=50K)	6K	5.61	71.21
LIMA	1K	4.29	41.98
TAGLM†	6K	6.09	72.80
DEITA-LLaMA1-13B$_{6K}$	6K	**6.46**	**77.08**

Table 5: Comparison of different data selection approaches, the backbone is LLaMA-1-13B. For Alpagasus, we could not use ChatGPT to score all the examples in the pool due to the cost, thus we score 50K random examples and select. † denotes the results obtained by using their released LLaMA-7B tagger model for a fair comparison.

Model	Data Size / Alignment	ARC	HellaSwag	MMLU	TruthfulQA	Average
Open-sourced Models based on LLaMA-1						
LIMA	1K / SFT	59.22	84.25	49.60	46.20	59.82
WizardLM-13B	70K / SFT	57.25	80.88	52.92	50.55	58.96
Vicuna-13B-v1.3	125K / SFT	54.61	80.41	52.88	52.14	60.01
Random-Select	10K / SFT	55.80	79.95	47.35	57.44	60.14
DEITA-LLaMA1-13B$_{10K}$	10K / SFT	59.47	82.01	60.60	55.03	**64.27**
Open-sourced Models based on LLaMA-2						
Vicuna-13B-v1.5	125K / SFT	57.08	81.24	56.67	51.51	61.63
Random-Select	10K / SFT	61.52	83.69	55.22	44.84	61.32
DEITA-LLaMA2-13B$_{10K}$	10K / SFT	58.87	82.08	55.33	54.57	**62.71**
Open-sourced Models based on Mistral-7B						
Mistral-7B-Instruct-v0.1	–	54.52	75.63	55.38	56.28	60.45
zephyr-beta-sft	200K / SFT	57.68	81.98	61.04	43.00	60.93
zephyr-beta	200K / SFT + 60K / DPO	62.03	84.52	61.44	57.44	66.36
Random-Select	10K / SFT	55.38	79.16	58.73	53.59	61.72
DEITA-Mistral-7B$_{6K}$	6K / SFT	57.76	80.29	61.90	59.82	64.94
DEITA-Mistral-7B$_{6K}$+DPO	6K / SFT +10K / DPO	66.21	85.42	60.66	67.14	**69.86**

Table 7: Results on the Open LLM Leaderboard. Data size by default represents the number of examples in SFT unless specified otherwise.

Thanks