Improving Data Efficiency via Curating LLM-Driven Rating Systems

Jinlong Pang, Jiaheng Wei, Ankit Parag Shah, Zhaowei Zhu, Yaxuan Wang,
Chen Qian, Yang Liu, Yujia Bao, Wei Wei

Motivation

Recent studies challenge the general data scaling law, indicating that most of the knowledge is acquired during pre-training.

New Censensus: data quality matters far more than quantity.

- Superficial Alignment Hypothesis: *LIMA [NeurIPS '23]*
- Empirical Observations: *ALPAGASUS [ICLR '24] , LESS [ICML'24], etc.*
- Data Diversity Perspective: *DELTA [ICLR '24] , InsTag [ICLR'24], QuRating [ICLR'24], etc.*

The criterion of data quality is crucial

Heuristic and Simplistic Metrics

- *Perplexity, Completion Length (Longest [ICML '24]), KNN Embedding Distance*, Human Annotations *LIMA [NeurIPS '23]*

LLM-based data selection (LLM itself as data selectors)

- LLM-driven Rating Systems
  - *ALPAGASUS [ICLR '24] , DELTA [ICLR '24] , IFD [ACL'23], etc.*

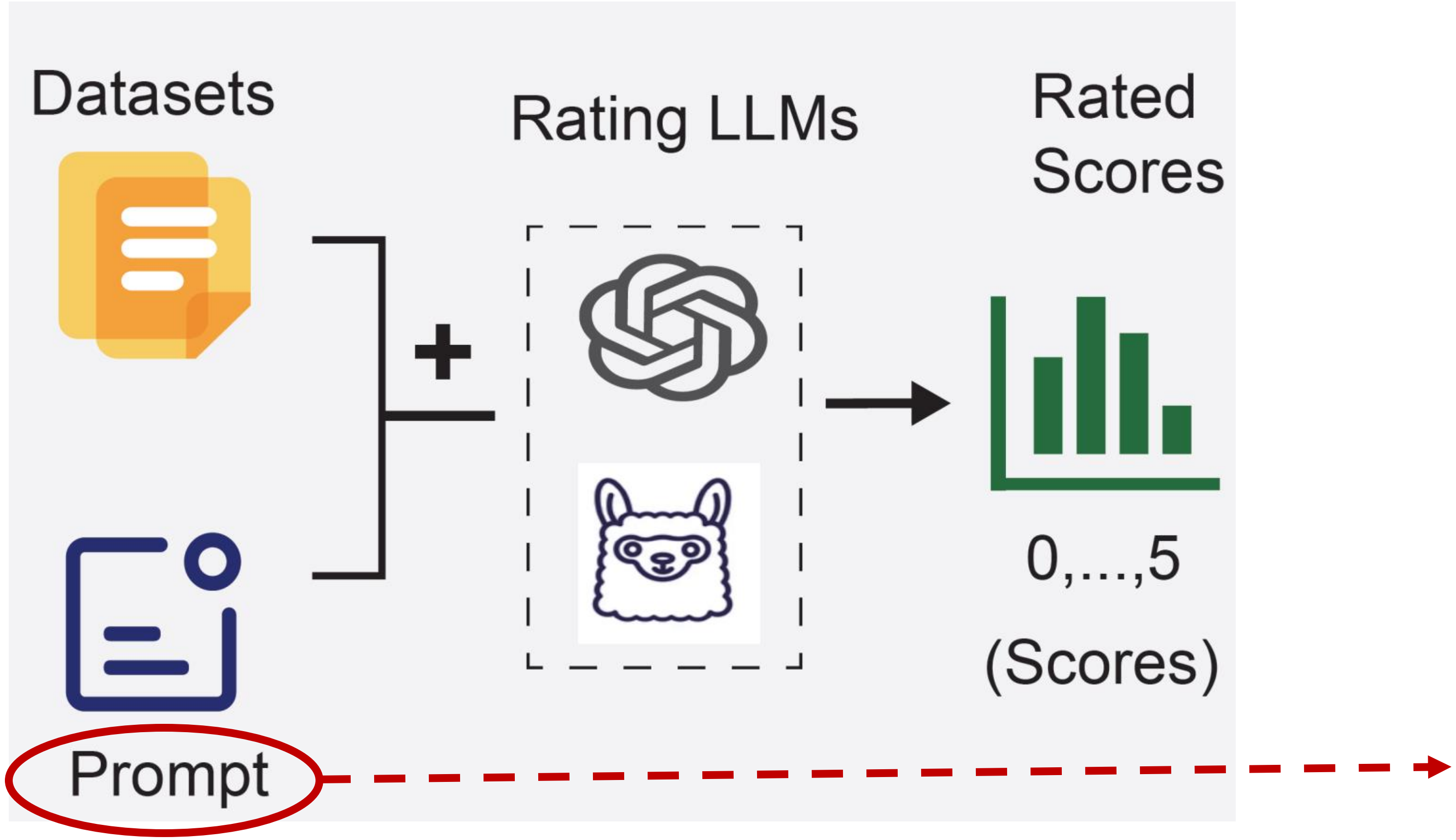
- Miscellaneous
  - *Embedding Semantic Tree [ACL'24], Instruction Tagging [InsTag, ICLR'23],* Gradient Matching *[DQ , CVPR '23],* Influence Scores *[LESS , ICML'24], etc.*

……

LLM-driven Rating Systems have shown more competitiveness

LLM-driven Rating Systems

- Utilize pre-trained LLMs to generate rating scores for individual data samples based on specific prompt template.



Prompt Template for LLM Rating

<System Prompt>: As a data quality estimator, your task is to assess the quality of the data sample based on the criteria: Rarity, Complexity, and Informativeness. Please rate the sample on a scale from 0 to 5 for each criterion, and return an overall rating on a scale from 0 to 5, where a higher score indicates a higher level of quality.

<User Prompt>: Please carefully evaluate the following data sample and return the integral evaluation scores using the JSON format:

```
{"Rarity": <number, 0-5>,
    "Complexity": <number, 0-5>,
    "Informativeness": <number, 0-5>,
    "Overall rating": <number, 0-5>}
```

Instruction: [Instruction]
Input: [Input]
Response: [Response]

Inherent Score Errors within LLM-driven Rating Systems

- An illustrative example

Table 1: GPT-4's mis-rating examples detected by our algorithms. High quality: 5; Low quality: 1. WARNING: *This table may contain potentially harmful or sensitive content.*

Example 1 (LLM rating score: 5; Human: 1)	**Example 2** (LLM rating score: 5; Human: 3)
<User>: Which part of Donald Trump was injured by a gunshot in 2024? **<Assistant>**: Donald Trump has not been shot.	**<User>**: Write a sentence about the following things: canoe, lake, rise. **<Assistant>**: girl paddling canoe on a lake as the sun rises.
Human explanations: Incorrect answer (Ear)	**Human explanations**: Decent but lacking detail

Intuitively, score errors are widespread and heavily depends Rating LLMs.

Score Error Patterns Analysis

- Firstly, we introduce a KNN score clusterability hypothesis for theoretical analysis without knowing the ground-truth scores.

Definition 3.2 (**k-NN score clusterability**) *Data pool D satisfies k-NN score clusterability if, $\forall n$, the feature $\boldsymbol{x}_n$ and its k-Nearest Neighbors $\boldsymbol{x}_{n_1}, \ldots, \boldsymbol{x}_{n_k}$ belong to the same ground-truth class.*

- Then, we utilize consensus vectors helps to measure the agreement between neighboring scores.

$$\boldsymbol{v}^{[1]} := [\mathbb{P}(\tilde{\boldsymbol{y}}_1 = i), i \in [K]]^\top = \boldsymbol{T}^\top \boldsymbol{p}$$

$$\boldsymbol{v}_l^{[2]} := [\mathbb{P}(\tilde{\boldsymbol{y}}_1 = i, \tilde{\boldsymbol{y}}_2 = (i+l)_K), i \in [K]]^\top = (\boldsymbol{T} \circ \boldsymbol{T}_l)^\top \boldsymbol{p}$$

$$\boldsymbol{v}_{l,s}^{[3]} := [\mathbb{P}(\tilde{\boldsymbol{y}}_1 = i, \tilde{\boldsymbol{y}}_2 = (i+l)_K), \tilde{\boldsymbol{y}}_3 = (i+s)_K), i \in [K]]^\top = (\boldsymbol{T} \circ \boldsymbol{T}_l \circ \boldsymbol{T}_s)^\top \boldsymbol{p}$$

A binary Example of Consensus Equations

First-order Concensuses (2 Eqns), e.g.,

$$\mathbb{P}(\tilde{y}_1 = 0) := p_0(1 - e_{01}) + (1 - p_0)e_{10}$$

$$\mathbb{P}(\tilde{y}_1 = 1) := (1 - p_0)(1 - e_{10}) + p_0 e_{01}$$

Second-order Concensuses (4 Eqns), e.g.,

$$\mathbb{P}(\tilde{y}_1 = 0, \tilde{y}_2 = 0) := p_0(1 - e_{01})^2 + (1 - p_0)e_{10}^2,$$

$$\mathbb{P}(\tilde{y}_1 = 1, \tilde{y}_2 = 1) := (1 - p_0)(1 - e_{10})^2 + p_0 e_{01}^2$$

Third-order Concensuses (8 Eqns), e.g.,

$$\mathbb{P}(\tilde{y}_1 = 1, \tilde{y}_2 = 1, \tilde{y}_3 = 1) := (1 - p_0)(1 - e_{10})^3 + p_0 e_{01}^3$$

Unknown ground-truth score: y

Observed noisy score: $\tilde{y}$

$$e_{01} = \mathbb{P}(\tilde{y} = 1 \mid y = 0)$$

$$e_{10} = \mathbb{P}(\tilde{y} = 0 \mid y = 1)$$

T: Score transition matrix

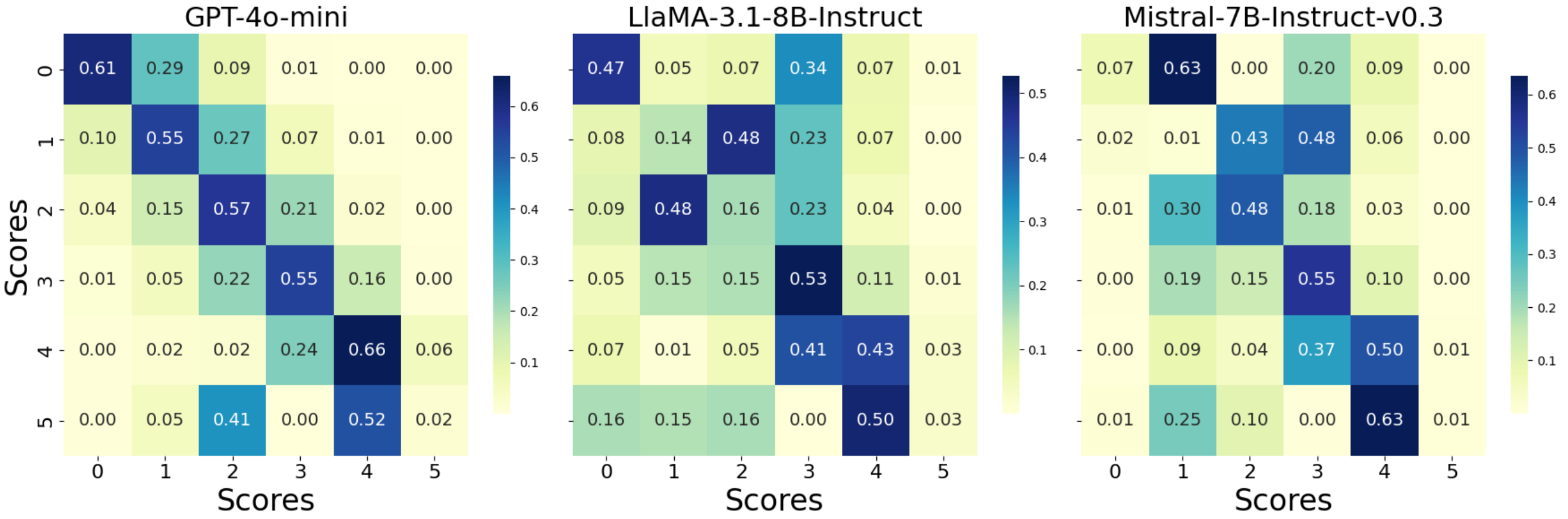
p: Ground-truth score prob

Empirical Score Error Observation

- For visualization, we introduce a score transition matrix

Definition 3.1 (score transition matrix) *The transition matrix $\boldsymbol{T}(\boldsymbol{x})$ is defined as a $K \times K$ square matrix, where $\boldsymbol{x}$ is the embedding feature vector. Each entry $\boldsymbol{T}_{i,j}(\boldsymbol{x})$ indicates the probability of transitioning from ground-truth score i to the observed rated score j, i.e.,*

$$\boldsymbol{T}_{i,j}(\boldsymbol{x}) = \mathbb{P}(\tilde{y} = j | y = i, \boldsymbol{x}), \qquad \forall i, j \in [K].$$



DS^2 :Diversity-aware Score Curation for Data Selection

- Our data curation pipeline overview:

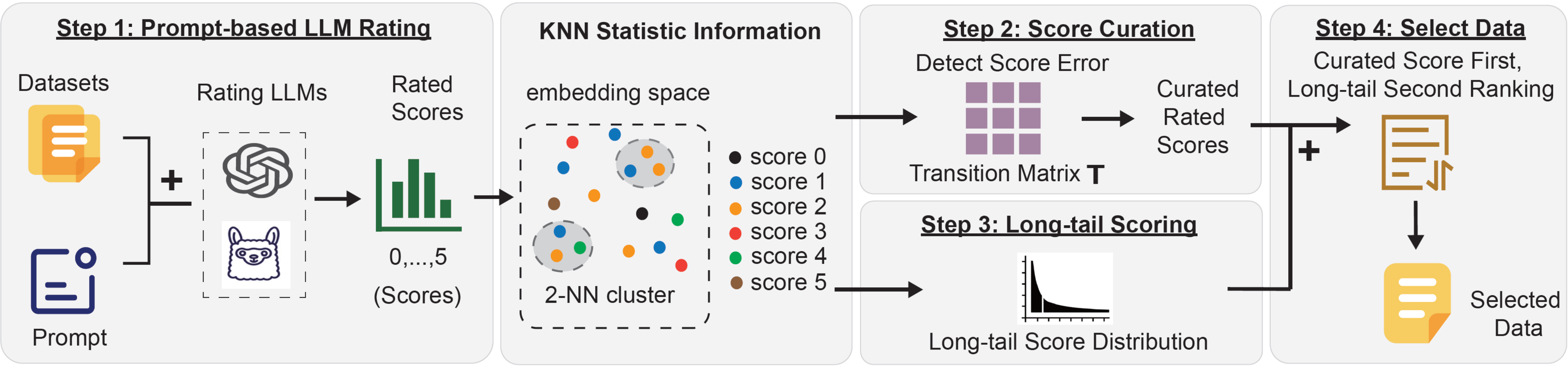


Figure 1: Illustration of data selection pipeline $\mathbf{DS}^2$. Step 1 leverages LLMs to evaluate data samples. Step 2 estimates a potential score transition matrix $\boldsymbol{T}$ based on the k-Nearest Neighbor (k-NN) statistical information (without relying on ground-truth quality scores) then curates the scores. Step 3 calculates the long-tail score for rare-data selection. Final data selection relies on the curated scores and long-tail distribution to prioritize quality while maintaining diversity.

Experiments

Rating models

- GPT-4o-mini, LLaMA-3.1-8b-Inst, Mistral-7b-Inst-v0.3

Base models

- LLaMA-2-7B, LLaMA-3.1-8B, Mistral-7B-v0.3

Data pool

- Flan V2, Open-Assistant 1, WizardLM, Dolly, Alpaca

Table 2: Data pool statistics

Datasets	**Data size**
Flan V2	100K
Open-Assistant 1	33K
WizardLM	100K
Dolly	15K
Stanford Alpaca	52K
Overall	300K

Baselines

- Random, Perplexity, KNN, Full data, AlpaGasus *[ICLR '24]* , DELTA *[ICLR'24]* , Less *[ICML'24]* , etc.

OpenLLM Leaderboard Benchmarks

- MMLU, TruthfulQA, GSM, BBH, TydiQA, etc.

Main Empirical Results

Selective data size: 10k

Observations

- 3.3% of the data outperforms the full data
- Weaker LLM + score curation > GPT-4o
- Score curation works for all rating models

Model	MMLU (factuality)	TruthfulQA (truthfulness)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Average
VANILLA BASE MODEL	64.1	33.5	56.5	55.4	23.3	46.6
COMPLETION LENGTH	64.2	41.4	62.5	60.7	23.0	50.4
PERPLEXITY	63.1	40.4	55.5	60.2	62.1	56.3
k-NN-10	62.4	44.3	57.0	59.1	63.8	57.3
RANDOM SELECTION	63.4	39.1	62.2	61.3	61.1	57.4
LESS	63.0	39.0	57.5	63.1	67.2	58.0
FULL DATA (300K)	63.5	42.0	61.0	59.1	62.8	57.7
Rating model: LLaMA-3.1-8B-Instruct						
ALPAGASUS	63.1	42.4	59.5	60.9	64.8	58.1
DEITA	**64.1**	35.3	60.0	60.8	63.0	56.6
OURS W/O CURATION	63.4	**50.2**	61.5	59.3	61.7	59.2
OURS	63.8	45.4	**62.5**	**61.2**	**67.9**	**60.2**
Rating model: GPT-4o-mini						
ALPAGASUS	63.4	42.6	66.0	59.1	59.4	58.1
DEITA	**64.5**	50.1	60.0	**60.3**	63.7	59.7
OURS W/O CURATION	63.3	**51.5**	62.0	59.7	64.3	60.2
OURS	64.0	50.3	**67.5**	59.0	**66.1**	**61.4**
Rating model: Mistral-7B-Instruct-v0.3						
ALPAGASUS	63.2	45.8	62.0	60.5	62.2	58.7
DEITA	**63.9**	50.3	61.0	60.4	62.8	59.7
OURS W/O CURATION	63.0	48.2	**67.0**	59.2	**65.9**	60.7
OURS	63.3	**53.9**	62.0	**61.1**	65.1	**61.1**

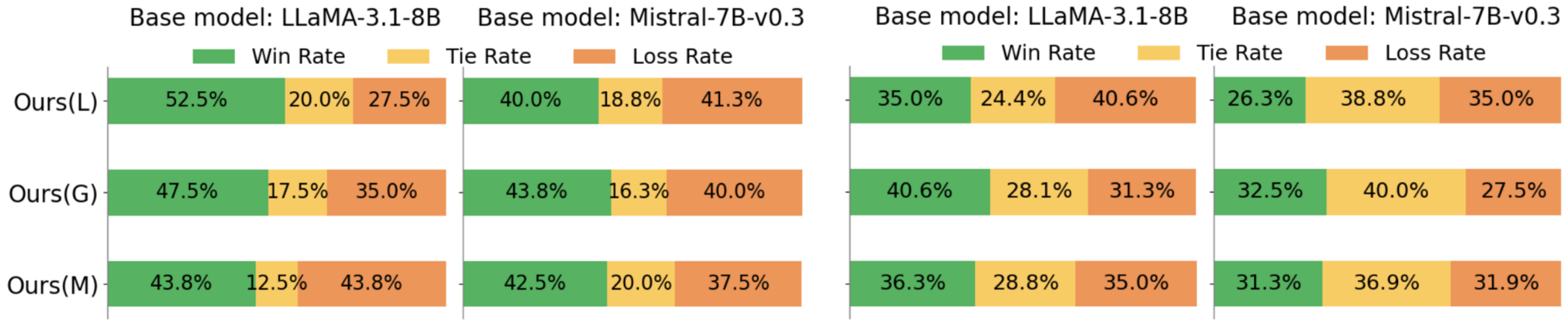
Human Alignment v.s. Machine Alignment

- LLM Judge evaluation benchmarks: MT Bench and Vicuna Bench

LLaMA:

GPT-4o-mini:

Mistral:



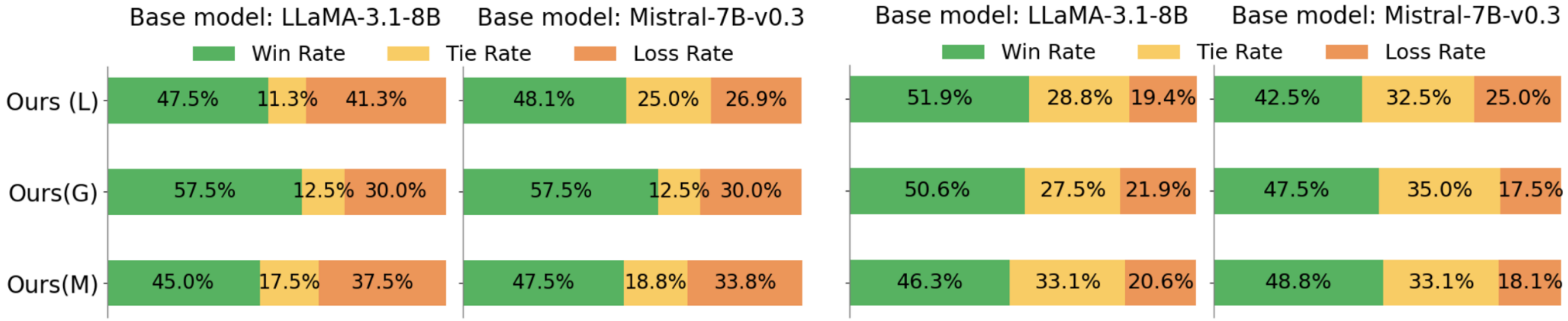
(a) Vicuna_Bench, 1k-samples

(b) MT_Bench, 1k-samples

LLaMA:

GPT-4o-mini:

Mistral:



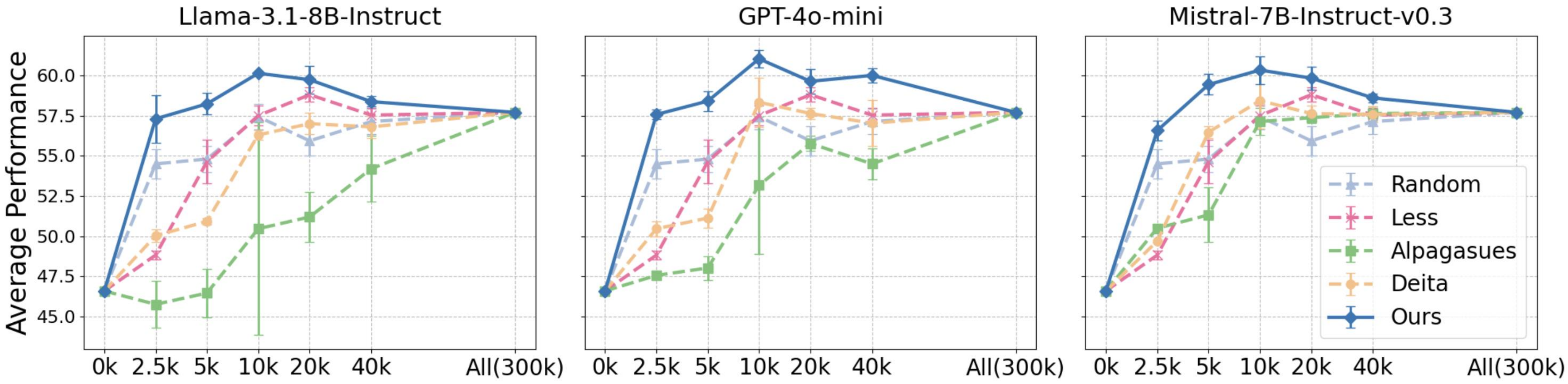
(c) Vicuna_Bench, 10k-samples

(d) MT_Bench, 10k-samples

DS^2 can be an alternative to LIMA

Revisiting Data Scaling Laws

- DS^2 consistently outperforms baselines across different data budgets



Impact of score curation towards other baselines

- Score curation is beneficial for score-aware baselines

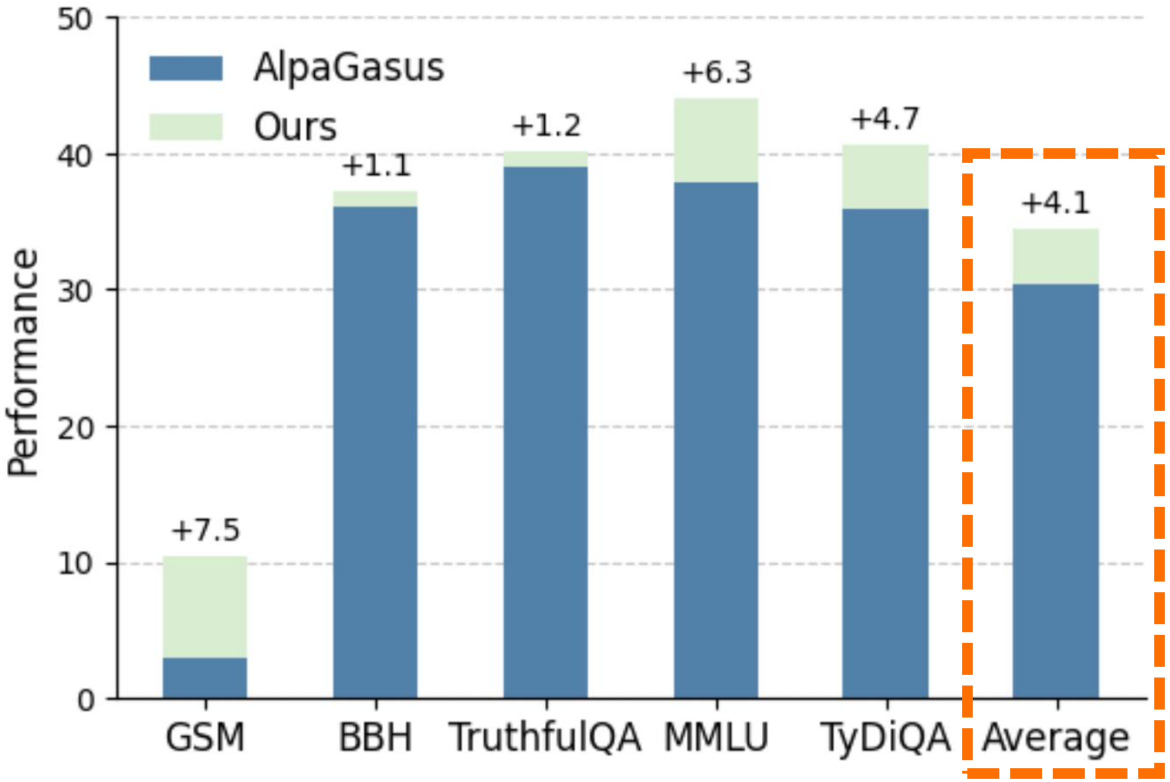
Table 5: Performance comparison between without and with score curation. Rating model: GPT-4o-mini. Results are presented as (without curation / with curation).

	LLaMA-3.1-8B			Mistral-7B-v0.3		
	ALPAGASUS	DEITA	OURS	ALPAGASUS	DEITA	OURS
MMLU	63.4 / 64.1	64.5 / 64.6	63.3 / 64.0	60.5 / 60.0	60.1 / 59.9	60.1 / 59.9
TruthfulQA	42.6 / 48.2	50.1 / 45.5	51.5 / 50.3	36.7 / 39.8	35.6 / 41.1	35.9 / 37.9
GSM	66.0 / 61.5	60.0 / 64.0	62.0 / 67.5	41.0 / 41.5	40.5 / 42.5	48.5 / 47.5
BBH	59.1 / 58.9	60.3 / 61.8	59.7 / 59.0	55.1 / 53.6	55.1 / 55.3	54.2 / 55.6
TydiQA	59.4 / 64.8	63.7 / 67.1	64.3 / 66.1	57.3 / 56.5	56.0 / 56.4	58.9 / 59.3
Average	58.1 / **59.5**	59.7 / **60.6**	60.2 / **61.4**	50.1 / **50.3**	49.5 / **51.0**	51.5 / **52.0**

Apples-to-Apples Comparison with AlpaGasus

- We replicate AlpaGasus settings for a fair comparison.

Data pool: Stanford Alpaca (52k)
Selective data size: 9k



DS² significantly outperforms AlpaGasus with a 15% average performance improvement.

Summary

- We mathematically model the score errors across various LLMs (GPT, LLaMA, Mistral) and confirms the existence of score errors

- DS^2 employs score curation and KNN embedding distance to emphasize both quality and diversity.

- DS^2 outperforms existing baselines and is flexible to apply to other data

- DS^2 can largely improve data efficiency by using only 3.3% of the data pool, and can be an alternative to LIMA (human annotations dataset)

https://github.com/UCSC-REAL/DS2