Technical Report: Competition Solution For BetterMixture

Shuaijiang Zhao,Xiaoquan Fang
Beike Inc., Beijing, China
{zhaoshuaijiang001,fangxiaoquan001}@ke.com

Abstract

In the era of flourishing large-scale models, the challenge of selecting and optimizing datasets from the vast and complex sea of data, to enhance the performance of large language models within the constraints of limited computational resources, has become paramount. This paper details our solution for the BetterMixture challenge, which focuses on the fine-tuning data mixing for large language models. Our approach, which secured third place, incorporates data deduplication, low-level and high-level quality filtering, and diversity selection. The foundation of our solution is Ke-Data-Juicer [1], an extension of Data-Juicer, demonstrating its robust capabilities in handling and optimizing data for large language models.

1 Introduction

The emergence of large-scale language models such as ChatGPT(OpenAI, 2023) has transformed natural language processing. Meanwhile, the rapid growth of Chinese open-source large language models, including ChatGLM(Zeng et al., 2022), Baichuan(Yang et al., 2023), Qwen(Bai et al., 2023), and BELLE(BELLEGroup, 2023), contributing positively to the field's evolution.

The swift development of Large Language Models (LLMs) has highlighted the critical need for vast quantities of high-quality data. In the response, BetterMixture emerges as a data-centric challenge that tests the analysis and combination capabilities of fine-tuning data for LLMs, bridging the gap between data needs and model optimization.

To tackle the challenge, we utilized our Ke-Data-Juicer system, an advancement of Data-Juicer. Data-Juicer(Chen et al., 2024) is a comprehensive one-stop data processing system for Large Language Models. It is capable of efficiently generating a variety of data recipes, exploring numerous combinations for creating data mixtures, and assessing their impact on model performance. Ke-Data-Juicer builds upon this foundation by enhancing high-level quality filtering and diversity selection capabilities.

Building on Ke-Data-Juicer, we applied standard filtering techniques, including text length, language identification, and specific word filtering, referred to as low-level quality filtering.

To enhance data quality filtering, we introduced high-level quality filtering, with a LLM serving as a trainable data selector. This process evaluates and assigns scores to each sample of instruction fine-tuning data. Specifically, we introduced Perplexity (PPL) calculated by the LLM to quantify the difficulty of instructions. Instruction Following Difficulty (IFD) (Li et al., 2023) also introduced to assess the challenge of responding to specific instructions. Furthermore, we introduced the IFD-Vote method, which utilizes multiple LLMs to refine quality assessment based on their collective scores.

In addition to quality, diversity is crucial. We employ the k-center-greedy algorithm to enhance the diversity of the selected data mixture

In summary, there are three main contributions of this paper:

- We proposed a complete solution for the BetterMixture challenge, securing third place in the competition.
- We introduced high-level quality filtering methods based on LLMs, including LLM perplexity filtering and LLM Instruction-Following Difficulty (IFD) filtering techniques.

[1] https://github.com/shuaijiang/ke-data-juicer

- We introduced the IFD-Vote method, leveraging multiple LLMs, to select high-quality instruction data.

2 BetterMixture Challenge

BetterMixture[1] is a data-centric challenge that assesses the ability to analyze and combine fine-tuning data for Large Language Models (LLMs). The organizers provide several candidate fine-tuning datasets, requiring participants to conduct data analysis, design mixing and sampling strategies, assign certain mixing ratios to each candidate subset, and create a mixed fine-tuning dataset within given computational constraints.

The candidate data originate from 20 datasets of Alpaca-CoT. During data analysis and sampling, participants can only use these specified datasets and are strictly prohibited from modifying any data or adding external data.

This competition exclusively uses the Baichuan2-7B-Base model, employing PEFT (LoRA(Hu et al., 2021)) training with a training data limit of 10M tokens.

Evaluation is of paramount prominence to the accuracy and fairness of the competition. The evaluation dataset is detailed in Table3, encompasses a broad range of capabilities. The evaluation metrics involve computing the ratio of the participant's evaluation score on each task to the baseline score of the Baichuan2-7B-Base model. The leaderboard score is derived by averaging all the individual task ratios.

3 Methodology

Formally, we define the instruction dataset X and employ a selection method F to extract a subset S from X, denoted as:

$$S = F(X) \tag{1}$$

An evaluation metric Q to assess the quality of subset S, guiding to obtain the optimal selection method $F^{\star}$:

$$F^{\star} = \underset{F}{\operatorname{argmax}}\, Q(S) \tag{2}$$

This selection illustrates our optimal selection method $F^{\star}$, encompasses several critical steps: deduplication, quality filtering, and diversity selection. The overview of our solution is shown in Figure1.

[1] https://tianchi.aliyun.com/competition/entrance/532174

3.1 Data Deduplication

Data deduplication has become a fundamental process that boosts training efficiency and has the potential to improve the model's performance. To avoid altering the data distribution, instead of hash based or model based deduplication approaches, we opted for MD5 deduplication method via exact match. This technique reduced the number of samples from 3.4 million to 2.7 million, streamlining dataset while preserving its diversity and richness.

3.2 Low-level Quality Filtering

After deduplication, we applied common standard filtering, namely low-level quality filtering, including text length and language identification filtering. Through analyzing text lengths and language scores shown in Figure 2, we established appropriate boundaries.

Text Length Filtering We conducted statistical analysis on the length of samples, which are composed of instruction, input, and output, following data deduplication. We retained samples with text length ranging from 20 to 2000.

Language Identification Filtering The dataset predominantly comprises English and Chinese languages, and the evaluation concentrates on English and Chinese. We preserved samples with English and Chinese with scores greater than 0.2. Recognizing that certain types of samples, such as code and mathematics, might not achieve high language scores, we set the threshold score at 0.2 to accommodate these variations.

3.3 High-level Quality Filtering

To enhance data quality filtering, we employed LLM for scoring and filtering of the data, which refered as high-level quality filtering. This approach encompasse LLM PPL, LLM IFD and LLM IFD-Vote filtering. The data distribution after high-lebel quality filtering shown in Figure3 (B).

LLM Perplexity filtering Perplexity is a key evaluation metric for language models, measuring the model's understanding of instructions. The larger the perplexity value, the lower the model's understanding of the instructions, indicating that the model needs more of this data for training. However, to prevent the model from being affected by errors in the data itself, we control the perplexity within a targeted range of 20 to 1000.

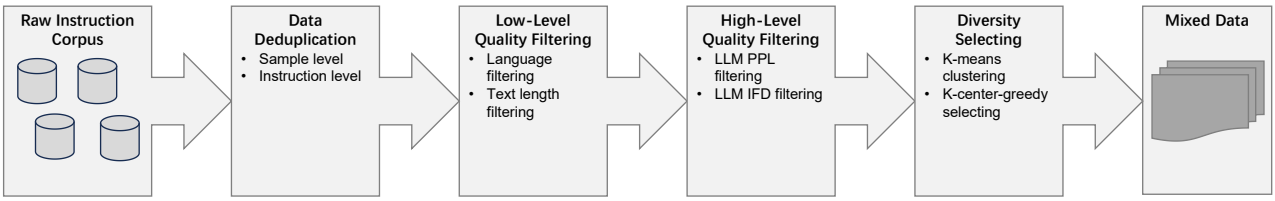


Figure 1: Overview of our competition solution.

This avoids selecting data with excessively high perplexity, thereby preventing the introduction of excessive noise into the model. We utilized the Baichuan2-7B-Base model, ensuring a consistent and reliable metric for filtering.

LLM IFD filtering IFD(Instruction Follow Difficulty) quantify the challenge each sample presents to the model, which is determined by the score of the conditional answer $S_\theta(A|Q)$ and the score of the model's direct answer $S_\theta(A)$.

$$IFD = \frac{S_\theta(A|Q)}{S_\theta(A)} \tag{3}$$

The score of the conditional answer measures the degree of consistency between the model's output and the correct answer corresponding to the instruction.

$$S_\theta(A|Q) = \frac{1}{N}\sum_{i=1}^{N} \log P(w_i^A|Q, w_1^A, w_2^A, ..., w_{i-1}^A; \theta) \tag{4}$$

The direct answer score measures the inherent difficulty brought by the answer.

$$S_\theta(A) = \frac{1}{N}\sum_{i=1}^{N} \log P(w_i^A|w_1^A, w_2^A, ..., w_{i-1}^A; \theta) \tag{5}$$

The larger IFD score, the more difficult the instruction is, suggesting the model has more to learn from this data. An IFD score too large or exceeding 1 suggests the instruction negatively impacts learning, whereas a low IFD score indicates the instruction is simple enough that the model can follow it without additional training.

Therefore, to eliminate overly simplistic or anomalous data, we maintain the IFD score within the range of 0.2 to 0.9. Adhering to the challenge's specifications, we employed the Baichuan2-7B-Base model to obtain the IFD score.

LLM IFD-Vote filtering As mentioned above, the IFD score indicating the challenge of a sample, is derived using the Baichuan2-7B-Base model in this study. However, due to the base model's limitations, the accuracy of the IFD score may vary. To enhance precision, we employed a fine-tuned version of the Baichuan2-7B-Base model as the IFD scorer. This model was fine-tuned with a mix of data processed through the previously and subsequently mentioned filtering and selection strategies.

Specifically, we calculated two IFD scores: one from the base model, another from the fine-tuned model. Samples exhibiting a variation of more than 50% between two scores were excluded.

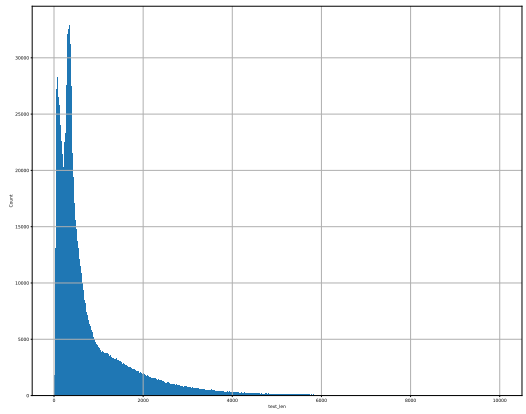
3.4 Diversity Selection

Diversity is equally critical as quality to ensure the generalization capabilities of LLMs. Meanwhile, given the constraint of training tokens capped at 10 million, corresponding to approximately 60,000 samples. We select samples based on the IFD score and the k-center greedy algorithm to satisfy token constraint and guarantee high diversity.

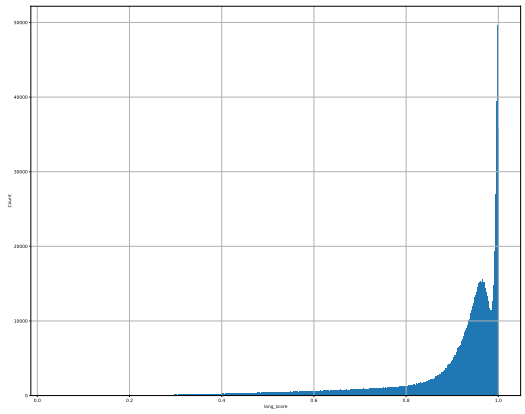
IFD based selection Specifically, we selected samples from each data source based on their average IFD score, targeting a total data size of 70,000. However, for some data sources, this target number was unmet, leading to a final collection of 60,000 samples. The details of data distribution shown in Figure 3 (C).

Language selection As illustrated in Table4, Baichuan2-7B-Base exhibits competitive performance with other similarly sized LLMs, even comparable to GPT3.5 Turbo which is a larger and more powerful LLM on Chinese benchmarks. This suggests Baichuan2-7B-Base has already achieved a high level of proficiency in Chinese, offering limited room for further improvement through continuous training.

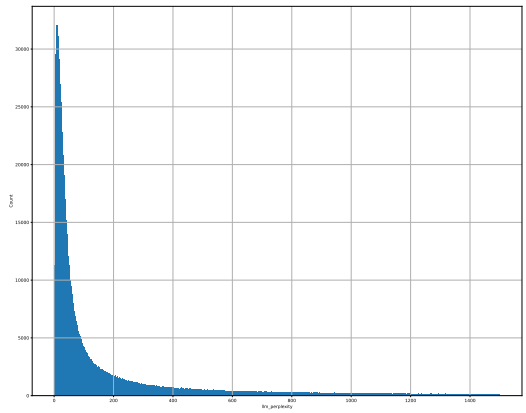
To optimize quality and diversity, we applied k-center greedy algorithm to refine our selection of Chinese samples, reducing their number from 13,000 to 9,000 without performance degrade.



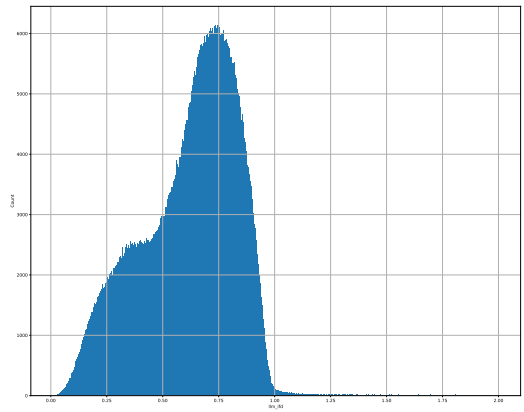
(A) Text Length Distribution



(B) Language Identification Score Distribution



(C) LLM Perplexity Distribution



(D) LLM IFD Score Distribution

Figure 2: Data statistics via histograms. (A) depicts the distribution of text lengths within the dataset. (B) shows the distribution of language identification scores. (C) presents the distribution of LLM perplexity. (D) illustrates the distribution of IFD scores.

4 Experiments

4.1 Baseline Models

The baseline model required by the organizer is Baichuan2-7B-Base(Yang et al., 2023), which was developed and released by Baichuan Intelligent Technology.

Baichuan2-7B-Base is a pre-trained model, boasting a parameter size of 7 billion and a training corpus comprising 2.6 trillion tokens.

4.2 Dataset

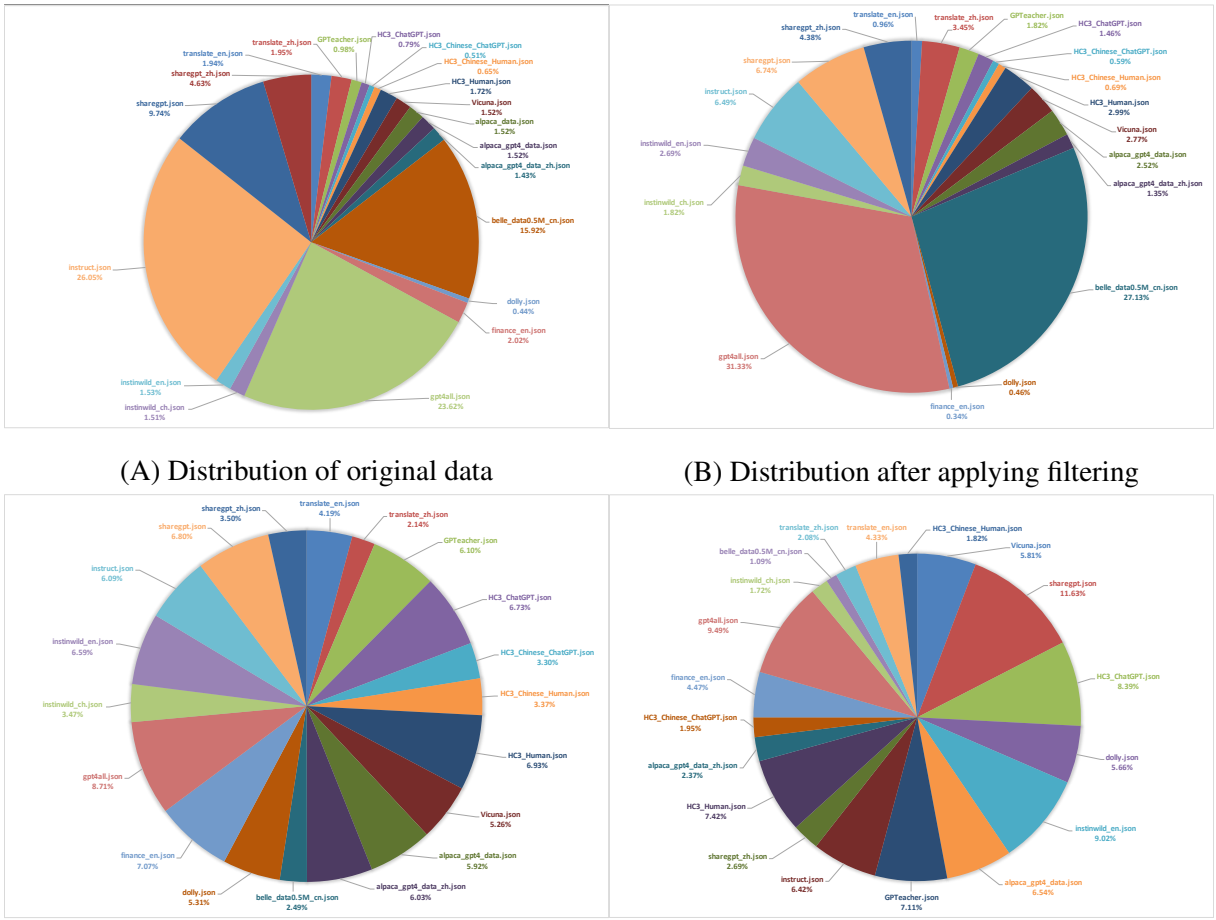
We conducted a multi-stage analysis of the dataset: the original dataset, applying low-level and high-level filtering, following IFD-based selection, and concluding the final dataset. The data distribution for each of these stages is illustrated in Figure 3.

4.3 Training Setups

Most hyper-parameter settings were determined by the organizer. The hyper-parameter settings we employed are detailed in Table 1. We chose a learning rate of 1e-3, the highest among 1e-3, 1e-4, and 1e-5.

4.4 Evaluation

Table3 presents a detailed overview of the evaluation stages for the competition, breaking down the specific categories assessed at each stage. In the preliminary stage, we outline the benchmarks for evaluating competitors' skills across a range of domains including reasoning, common sense, truthfulness, math, English knowledge, Chinese



(A) Distribution of original data

(B) Distribution after applying filtering

(C) Distribution after IFD-based selection

(D) Distribution of final mixed data

Figure 3: Data Distribution: (A) presents the original dataset distribution, (B) shows the distribution after applying low-level and high-level filtering, (C) depicts the distribution following IFD-based selection, and (D) illustrates the distribution of the final mixed data.

Table 1: Training Hyper-parameter settings

Hyper parameter	Value
Precision	bfloat16
Epochs	3
Batch size	1
Learning rate	1e-3
Warmup ratio	0.03
LR scheduler type	cosine

knowledge, and summarization. For the finals stage, while the domains remain the same, the datasets are undisclosed to participants.

4.5 Results

Table2 displays the evaluation scores, demonstrating the performance improvements achieved through various strategies. Notably, the most effective improvements were attained through low-level and high-level filtering, IFD-selection, and k-center selection. Additionally, descending ordering samples by PPL and adopting a larger learning rate of 1e-3 proved to be beneficial.

Table 2: Evaluation Scores. Scores marked with a $\star$ represent the final stage, while others correspond to the preliminary stage.These two sets of scores are not directly comparable due to differing evaluation conditions.

Exp	Strategies	Scores
Baseline	random selection	1.342
Exp 1	low-level & high-level filtering & IFD-selection	1.401
Exp 2	+ k-center selection	1.431
Exp 3	+ PPL descending order	1.443
Exp 4	+ Learning rate 1e-4 to 1e-3	1.455
Exp 5	+ IFD-Vote	1.567*

Table 3: Evaluation Detail

Category	Preliminary Stage (#Sample)	Finals Stage
Reasoning	ARC(100)	blind
Common Sense	HellaSWAG(100)	blind
Truthfulness	TruthfulQA(100)	blind
Math	GSM8K(100)	blind
English Knowledge	MMLU(100*67)	blind
Chinese Knowledge	CMMLU(100*67)	blind
Summarization	SummScreen(100)	blind

Table 4: Overall results(Yang et al., 2023) of Baichuan2-7B-Base compared with other similarly sized LLMs on general benchmarks

	C-Eval	MMLU	CMMLU	Gaokao	AGIEval	BBH	GSM8K	HumanEval
GPT-4	68.4	83.93	70.33	66.15	63.27	75.12	89.99	69.51
GPT-3.5 Turbo	51.10	68.54	54.06	47.07	46.13	61.59	57.77	52.44
LlaMA-7B	27.10	35.10	26.75	27.81	28.17	32.38	9.78	11.59
LlaMA 2-7B	28.90	45.73	31.38	25.97	26.53	39.16	16.22	12.80
MPT-7B	27.15	27.93	26.00	26.54	24.83	35.20	8.64	14.02
Falcon-7B	24.23	26.03	25.66	24.24	24.10	28.77	5.46	-
ChatGLM 2-6B	50.20	45.90	49.00	**49.44**	**45.28**	31.65	**28.89**	9.15
Baichuan 1-7B	42.80	42.30	44.02	36.34	34.44	32.48	9.17	9.20
Baichuan 2-7B-Base	**54.00**	**54.16**	**57.07**	47.47	42.73	**41.56**	24.49	**18.29**

5 Conclusions

In this report, we introduce a solution for Better Mixture challenge, securing third place in the competition. We detailed the filtering and selection strategies implemented, highlighting their contribution to our success. Looking ahead, we aim to explore model-based data mixture learning techniques,such as DOREMI(Xie et al., 2024), as a promising direction for future work.

References

[Bai et al.2023] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

[BELLEGroup2023] BELLEGroup. 2023. Belle: Be everyone's large language model engine. `https://github.com/LianjiaTech/BELLE`.

[Chen et al.2024] Daoyuan Chen, Yilun Huang, Zhijian Ma, Hesen Chen, Xuchen Pan, Ce Ge, Dawei Gao, Yuexiang Xie, Zhaoyang Liu, Jinyang Gao, Yaliang Li, Bolin Ding, and Jingren Zhou. 2024. Data-juicer: A one-stop data processing system for large language models. In *International Conference on Management of Data*.

[Hu et al.2021] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

[Li et al.2023] Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2023. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*.

[OpenAI2023] OpenAI. 2023. Chatgpt: Optimizing language models for dialogue. Blog post.

[Xie et al.2024] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. 2024. Doremi: Optimizing data mixtures speeds up language model pretraining. *Advances in Neural Information Processing Systems*, 36.

[Yang et al.2023] Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

[Zeng et al.2022] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. Glm-130b: An open bilingual pre-trained model.