The Art of Data Selection: A Survey on Data Selection for Fine-Tuning Large Language Models

Anonymous ACL submission

Abstract

Recently, Large Language Models (LLMs) have seen significant advancements, and supervised fine-tuning (SFT) plays a pivotal role in unleashing LLMs' potential to follow the users' instructions. As an emerging research field, data selection for fine-tuning LLMs aims to select a subset from a given candidate dataset for training selective-enhanced models to improve their performance and accelerate their training. Although some studies have already investigated these works, there is a lack of comprehensive analysis and comparison of them to provide potential research directions. To fill the gap, we first summarize a three-step scheme for data selection on existing works, including data preprocessing, data selector construction, and data selector evaluation, and comprehensively sort out the existing works according to this scheme. Then, we conduct an in-depth analysis of existing works from their efficiency and feasibility by making quantitative and qualitative comparisons and find that (1) the model-specific method who takes the loss output of the pending fine-tune model as an optimized goal is more effective; (2) increasing the complexity of the selector can improve the performance of the selective-enhanced model, but it needs more careful design to avoid introducing external factors. Finally, we summarize the trends in data selection and point out that the current main challenges are the lack of unified and efficient data quality measurement, as well as data selection for specific tasks and multiple turns of conversations.

1 Introduction

Large language models nowadays can generate natural and authentic human languages and complete many classic NLP challenges as well as real-world tasks (Naveed et al., 2023; Vaswani et al., 2023; Wang et al., 2022; Zhong et al., 2022). After the knowledge-based pretraining, the user-oriented supervised instruction fine-tuning endows LLMs with the most significant performance rise. With the success of LIMA (Zhou et al., 2023), data selection, that is, how to select a few high-quality samples from existing datasets to fine-tune better models in downstream tasks according to some prior indicators, has gradually become a research hotspot. It can improve fine-tuned LLMs' performance and accelerate their training simultaneously. Although recent works (Wang et al., 2024; Albalak et al., 2024) list most of the existing data selection methods for fine-tuning LLMs, there is a lack of in-depth analysis and comparison between each method for providing potential research directions.

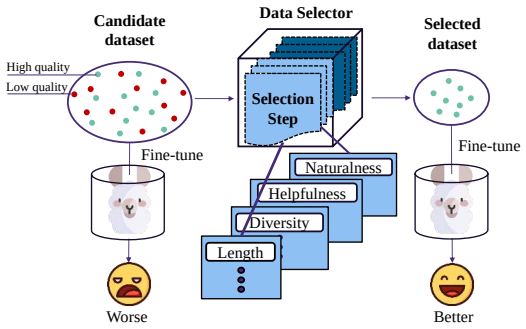

Figure 1: The Process of Data Selection.

Although recent works (Wang et al., 2024; Albalak et al., 2024) list most of the existing data selection methods for fine-tuning, there is a lack of in-depth analysis and comparison between each method.

To address these issues, we first summarize a three-step data selection scheme that can cover key parts of the entire data selection process, including data preprocessing, data selector construction, and data selector evaluation, after reviewing existing popular works. Then we conduct a comprehensive sort of the existing works based on the conversion format of the data to be selected, the indicator sources and calculation methods used by the selector, and the candidate datasets, models, and metrics used in the evaluation process.

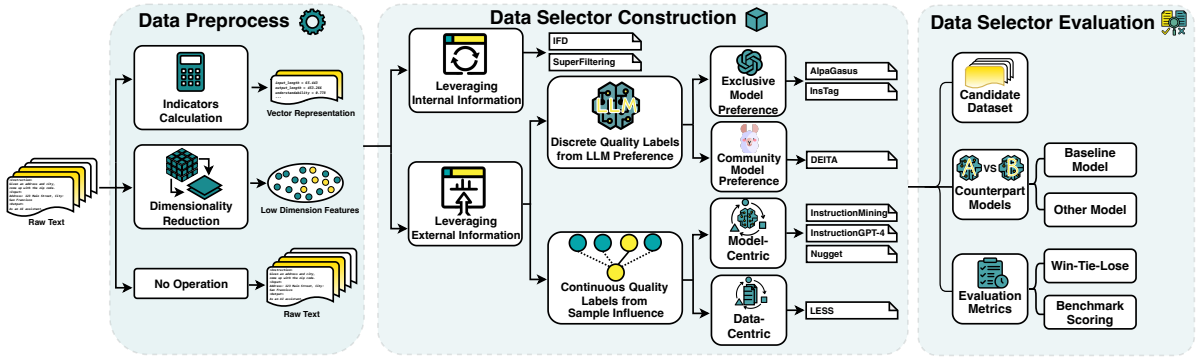

Figure 2: The Scheme of Data Selection for Fine-tuning LLMs.

Furthermore, we compared the existing work through quantitative and qualitative analysis. Specifically, we develop a unified efficiency measurement method based on the efficiency curve assumption to evaluate the performance of various models in selecting data, addressing hard comparisons due to different experimental settings. We also qualitatively evaluate their feasibility by considering simplicity and flexibility, including implementation costs and reproducibility.

Through the analysis of the above two aspects, we not only obtained the technological development path of existing work, but also identified the following two findings that can help with data selection in future work: (1) the selector taking the output of the pending fine-tune model as an optimized goal is more effective; (2) increasing the complexity of the selector can improve the performance of the selective-enhanced model, but it needs more careful design to avoid introducing external factors. Meanwhile, we point out that there are still many challenges to further research, including building unified and efficient data quality measurement and designing data selection for specific domains or multi-turn conversations.

2 Scheme of Data Selection

The data selection for fine-tuning LLM aims to select high-quality samples consisting of a subset from a given candidate dataset according to the data quality, resulting in the Selective-Enhanced Model (SEM) fine-tuned on the high-quality subset being better than the Baseline model (BM) trained on the full dataset.

Therefore, building a data selection method requires considering the following three aspects. Firstly, the form of data representation. For data selection, it is necessary to determine the selection perspectives. In addition to considering the characteristics of the text itself, linguistic features or vector representation of the text are also commonly used. After the conversion of original samples, the key to building a data selector is determining the selection measurement, which includes two aspects: the source of data quality clues and how to obtain the quality label of a sample. On the one hand, it can be obtained by calculating the statistical characteristics of the data itself. On the other hand, it can also be obtained through external information such as third-party scorer models or by comparing the performance with known good samples. Finally, it is necessary to verify the usefulness of the data selection method after obtaining a subset selected by the selector. It can pair-wise compare the response from the basic model and the selective-enhanced model directly or compare their scoring in some popular benchmarks.

By considering the above factors, we construct a full process scheme of data selection after reviewing existing popular works, which is divided into three steps: (1) data preprocessing (Section 3), (2) data selector construction (Section 4), and (3) data selector evaluation (Section 5).

3 Data Preprocessing

Data preprocessing converts raw texts into feature representations of the data for selection. According to the converted forms, it can be divided into human-preferred **explicit features** (Cao et al., 2023; Wei et al., 2023), such as the length of input, model-oriented **implicit features** (Xia et al., 2024), such as low-dimensional gradients from LoRA (Hu et al., 2021) or purely **original texts** (Li et al., 2023; Chen et al., 2024) where the raw texts can preserve the most information.

Explicit Features. Some works usually take a

series of indicators as explicit features manually based on the human's prior linguistic knowledge that can be extracted from the sample. For example, **InstructionMining** (Cao et al., 2023) converts each sample into a bag of NLP indicators (such as the length of input, the length of output, understandability, etc.). Utilizing such interpretable indicators of linguistic knowledge to represent the original sample can guide future selection with respect to these human-preferred aspects of a sample.

Implicit Features. Other works convert original texts to vector representations as implicit features of the sample. For example, **LESS** (Xia et al., 2024) randomly projects all candidate samples into low-dimensional gradient features with a warm-up LoRA, which can reflect the influence of each sample on the optimization process of the loss function. These implicit representations, though less interpretable to humans, are more objective features indicating the quality of the data.

4 Data Selector Construction

The choice of data quality measurement is the primary concern in constructing the data selector, which can be divided into two branches. One branch leverages internal information like statistical features from the candidate dataset (Li et al., 2024a). Another branch leverages the external information from LLM or datasets rather than from the SEM and candidate datasets. The leveraging external information works can be further divided into the group that uses discrete quality labels from LLM's preference, which refers to the score given by the external models like ChatGPT (Chen et al., 2024; Lu et al., 2023; Liu et al., 2023), and the group uses continuous quality label from sample influence (Cao et al., 2023; Wei et al., 2023; Li et al., 2024b; Xia et al., 2024).

4.1 Leveraging Internal Information

Leveraging internal information means using only the features of a given candidate dataset as quality clues to determine whether to select them. The pioneering work (Li et al., 2023) proposes Instruction Following Difficulty (**IFD**) as a quantified metric, which can be obtained by using only a candidate dataset and a backbone pre-trained model. A higher IFD score indicates a closer relationship between the sample's instruction and output, which means more useful information is given in the instruction and, thus, a higher quality of that sample. To obtain the IFD score, they train a LLaMa-7b as a warm-up selector on only a small portion of the candidate dataset to give the model basic instruction-following ability. Then, the IFD score can be computed by the following equation:

$$r_\theta(Q, A) = \frac{s_\theta(A|Q)}{s_\theta(A)} \quad (1)$$

where $r_\theta(Q, A)$ is the IFD score of a (Q, A) sample pair, θ means the warm-up selector model while $s_\theta(A|Q)$ and $s_\theta(A)$ are the likelihood of generating the same answer with or without giving the question as instruction.

Another work, **SuperFiltering** (Li et al., 2024a) adopts a modified version of the IFD score by replacing the likelihood function with perplexity values and selects samples with lower scores. Moreover, they use GPT-2 to train a smaller selector to determine data quality compared the previous work (Li et al., 2023).

4.2 Leveraging External Information

Leveraging external information for data quality measurement uses knowledge that is not accessible from the given candidate datasets or Pending Fine-tune Models (PFMs). PFM is the target model to be fine-tuned on the selected subset, using the same backbone model as SEM. To compute quality labels, external information is used in the form of either **discrete quality labels** from other LLMs (Chen et al., 2024; Lu et al., 2023; Liu et al., 2023) or **continuous quality labels** from sample influence (Cao et al., 2023; Wei et al., 2023; Li et al., 2024b; Xia et al., 2024), where the sample influence is reflected in model's performance gain induced by the sample.

4.2.1 Discrete Quality Labels from LLM Preference

Obtaining discrete quality labels relies on the use of external LLMs for their direct scoring or annotations on the candidate data. Such quality labels are discrete since other LLMs work as a blackbox that takes in the candidate data and outputs a response reflecting their preference. Based on whether the external LLM is trainable, one can obtain quality labels either **exclusive LLMs**, which are closed-source commercial models like GPT-4, or **community LLMs**, which are open-source trainable models like LLaMa.

Exclusive LLM Preference. Exclusive LLM's preference can be utilized as data quality labels

because many commercial models have high agreement with human annotators when evaluating the quality of data. **AlpaGasus** (Chen et al., 2024) determines data quality entirely from ChatGPT's preference reflected in its direct scoring on each sample. The score is obtained by prompting ChatGPT with a scoring template with common evaluation aspects, like helpfulness and accuracy. Then, they select higher quality samples with higher scores. Compared with AlpaGasus' straightforward prompt, **InsTag** (Lu et al., 2023) specifies clearer evaluation dimensions as tags when prompting ChatGPT. They propose a Complexity-first Diverse Sampling procedure for data selection. To obtain quality labels, their measure first sample-level complexity (average number of tags for each sample in the candidate subset) and then dataset-level diversity (the total number of distinct tags in the subset), balancing the interplay between data quality and diversity.

Community Model Preference. Community models are open-source, trainable models, which can be tailored for specific evaluation tasks after aligning them with external commercial models. **DEITA** (Liu et al., 2023) relies on the preference of a community model LLaMa to measure data quality, where the LLaMa learns from ChatGPT for scoring. To train the scorer, they utilize the ideas of evolving from WizardLM (Xu et al., 2023) to evolve a small set of sample seeds into different levels of complexity and quality and then fine-tunes a LLaMa on ChatGPT's scoring on these evoloved samples. For selection, they propose the Score-First, Diversity-Aware selection similar to that proposed in InsTag.

4.2.2 Continuous Quality Labels from Sample Influence

In search of more direct and model-specific data selection methods, this research line obtains continuous quality labels from sample influence, which is quantified by the performance improvement a model gains when fine-tuned on a sample. These improvements are gauged by continuous outputs like **model-centric** evaluation scores (Cao et al., 2023; Wei et al., 2023; Li et al., 2024b) or **data-centric** gradient similarity (Xia et al., 2024).

Model-centric. When using one sample to fine-tune the PFM, performance improvement in SEM is expexted to reflect the quality of that sample. Based on the assumption, InstructionMining (Cao et al., 2023) constructs the mapping between the 9-dimensional-indicator representations of the sample and the inference loss(Wang et al., 2023; Zheng et al., 2023). Then, they utilize BLENDSEARCH, effectively combining global and local optimizations with bayesian optimization and different local search threads, to determine the final selected dataset size. InstructionGPT-4 adopts the same logic on a multimodal model that can also process visual-caption features as input. To further avoid the fine-tuning cost, **Nugget** (Li et al., 2024b) measures the sample influence towards SEM by prompting the PFM to answer the same set of questions with or without that certain sample. Better question-answering results indicate a larger improvement the sample brings to PFM.

Data-centric. Compared with model-centric methods, data-centric approach compares the similarity between candidate data and known high-quality data's ability to improve model performance. To make comparison between data, **LESS** (Xia et al., 2024) proposes the Low-rank gradiEnt Similarity Search method. They first perform a warm-up LoRA to obtain gradient representations of the candidate dataset and then compare the similarity with high-quality dataset.

5 Data Selector Evaluation

To evaluate the usefulness of selectors, the method is to select a subset from a candidate dataset through the selector and then fine-tune a model to be the selectively enhanced model (SEM) based on this subset to compare the performance with the same model fine-tuned on full data (Baseline model, BM) or other popular oracle LLMs. Table 1 shows the detailed evaluation setting, including the choice of candidate datasets, counterpart models used in the comparison, and evaluation metrics that provide the performance.

Candidate Datasets. Most of the works (Li et al., 2024a, 2023; Liu et al., 2023) use the popular open-sourced datasets as candidate datasets to push forward better performance of fine-tuned models by selecting higher-quality samples in them. The candidate dataset is further divided into the typical group, including Alpaca, Dolly, FLAN, etc., and the advanced group developed from the typical datasets to achieve higher quality, including WizardLM, UltraChart, etc.

Counterpart Models. To objectively evaluate the performance of the SEM, most works choose BM as the counterpart model for comparison. They

Method	Candidate Datasets	Evaluating SEMs	Counterpart Models		Evaluation Metrics	
			BM	Others	Wins-ties-losses	Benchmark Scoring
AlpaGasus	Alpaca	LLaMA-2 7B	✔	✔	Vicuna, Koala, WizardLM, self-Instruct	InstructEval
Instruction-Mining	ALPACA & OPEN ASSISTANT, STACK-EXCHANGE & WIKIHOW	LLaMA 7B	✔	✘	OPENORCA & DOLLY	OPENLLM
InstructionGPT-4	MiniGPT-4	LLaMA-2	✔	✘	LLaVA-Bench	MME, VQA, MM-Bench
IFD	Alpaca & WizardLM	LLaMA-2 7B	✔	✘	Vicuna, Koala, WizardLM, self-Instruct, LIMA	OPENLLM
Superfiltering	Alpaca & Alpaca-GPT4 & WizardLM	LLaMA-2 7B/13B	✔	✘	WizardLM	OPENLLM, AlpacaEval
Nugget	Alpaca	LLaMA-2 7B	✔	✘	-	MT-Bench, AlpacaEval
LESS	FLAN V2 & CoT & DOLLY & OPEN ASSISTANT 1	LLaMA2-13B; Mistral 7B	✔	✘	-	MMLU, TYDIQA, BBH
InsTag	WizardLM & UltraChat & ShareGPT	LLaMA-1/-2	✘	✔	-	MT-Bench
DEITA	Alpaca & DOLLY & Oassit & FLAN 2022 & WizardLM & UltraChat & ShareGPT	LLaMA-1/-2 13B; Mistral 7B	✘	✔	-	OPENLLM, MT-Bench

Table 1: The candidate dataset, SEMs, counterpart models, and evaluation metrics used in each method. Some works, such as Instruction-Mining, use part of several datasets mentioned to form a candidate dataset. The "✔" under BM means the work uses the same BM as the evaluating SEM.; under Other Models, it means the work uses many models other than BM, including oracle LLM and other fine-tuned SEM; under wins-ties-losses, it means the work uses various methods to evaluate wins-ties-losses, such as AlpacEval, and directly using GPT-4 .

tend to use the popular LLaMa series (Chen et al., 2024; Lu et al., 2023) as well as Mistral (Liu et al., 2023; Xia et al., 2024) models as backbones of the SEM and BM to obtain relative improvement evaluation, which directly shows the improvement effect of the selector. Other works (Xia et al., 2024; Chen et al., 2024) compare the SEM with SOTA models (such as GPT-4, Claude, and LLaMA-Chat 7B) to obtain absolute improvement evaluation, which indicates how good SEM achieves.

Evaluation Metrics. Similar to the counterpart models, the evaluation metric adopts the relative and absolute methods to comprehensively evaluate the selector. The absolute metric uses Wins-ties-losses pairing scored by GPT-4 to indicate the direct performance difference between the SEM and counterpart model, while the absolute metric uses benchmark scoring to directly score and rank the SEM. Benchmark scoring is separated into a traditional group, which examines the loss of response on test tasks (such as MLU, TYDIQA, and Mosaic Eval Gauntle), and a group, which uses GPT-4 to score on various tasks, including MT-Bench, MM-Bench, AlpacaEval, and VicunaQA et al.

6 Analysis of Data Selection Method

To spot the common designs that lead to superior performance, we analyze the efficiency and feasibility of the existing data selection methods, distinguishing the superior and inferior work. Efficiency is quantified to examine the selection competence filtering, which is measured based on the overall consideration of the performance of SEM and the data size (selected dataset fraction), while feasibility uses a qualitative method to evaluate the difficulty of implementation, which entails both simplicity and flexibility.

6.1 Efficiency of the Selector

We manage to compare the data selection competence across different works by using efficiency, which refers to the expectation of probability in selecting the ground truth high-quality data at each bet (Appendix A.1).

Performance Improvement Ratio. The competence is reflected in the performance of SEM, where higher-quality data leads to a larger performance improvement. Performance improvement is the ratio of SEM's performance to that of the counterpart model, evaluated under various settings. Therefore, we first classify the evaluation settings into four categories and use Equation 4 to calculate the overall improvement rate for each category, and then further unify the four categories into the one that is the wins rate of SEM to BM which directly reflects the improvement effect brought by the selected subset (detailed information is in Appendix A.2).

Selected Dataset Fraction. However, the increase in data size also improves the performance (Kaplan et al., 2020). To evenly evaluate the impact of data size between different works, we use the selected dataset fraction, which refers to the fraction of the selected subset to the entire dataset. Though the size of the selected dataset is directly provided in the works, it is abandoned because it is heavily affected by the size of the entire dataset varying from 3,439 (Wei et al., 2023) to 306,044 (Lu et al., 2023).

To acquire efficiency from performance improvement, we further use the selected dataset fraction to eliminate the impact of data size. Figure 3 reflects the efficiency of the selected dataset fraction and

performance improvement ratio. In addition, as there are many pairs of dataset fractions and performances in a work, the one that achieves the optimal performance is adopted.

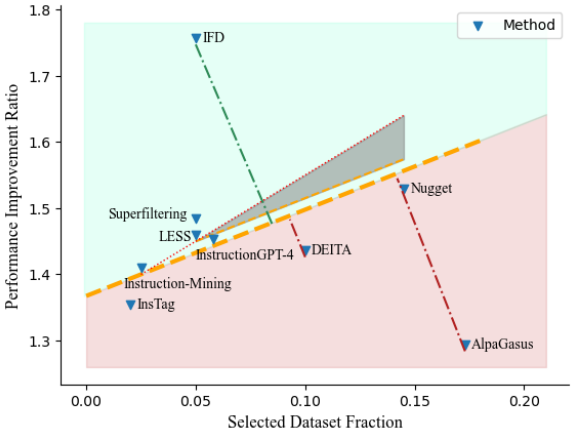

Figure 3: The comparison among popular data selection methods. The yellow line is the baseline based on Instruction-Mining and InstructionGPT-4. The signed distance between the method point and baseline is the efficiency difference between methods and baselines.

From the position of work in Figure 3, IFD has the highest efficiency because it achieves a colossal performance improvement with a small size of data. By contrast, AlpaGasus, which uses the largest selected dataset fraction but achieves the lowest performance improvement ratio, is the least efficient. Besides these two works, the superiority and inferiority of other works are elusive. Therefore, we develop the efficiency curve assumption to make the rest of the works mutually comparable (details are in Appendix A.3).

Under this assumption, the yellow dashed line represents the efficiency curve of Instruction-Mining and InstrucitonGPT-4, which is the baseline for separating the superior and inferior works. The grey area illustrates an infeasible area that every work has, where the efficiency of other work is incomparable (Appendix A.4). The red and green line, respectively indicates the superior and inferior efficiency of the work in comparison to the baseline, which is the signed distance between the baseline and the work, as calculated by the following:

$$Eff_i = \frac{Ax_i + By_i + C}{\sqrt{A^2 + B^2}} \quad (2)$$

$$l_{base} : Ax + By + C = 0 \quad (3)$$

where (x_i, y_i) is the position of work i, and l_{base} is the mathematical expression of the baseline. We present the comparison efficiency of each work in Figure 4, where the efficiency of DEITA is invisible because it is in the infeasible area of InsTag. We consider DEITA to be better than InsTag because it refines the method InsTag used.

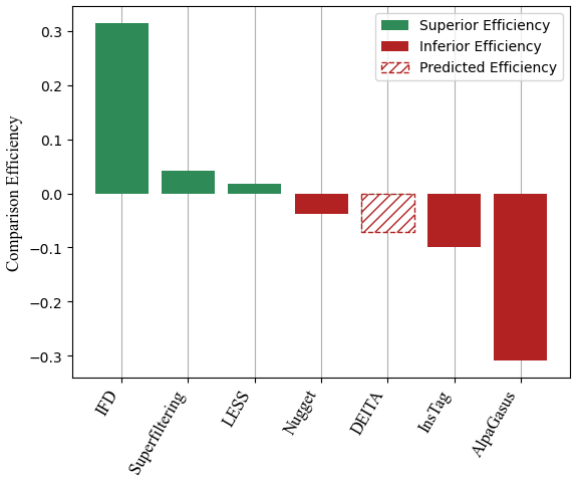

Figure 4: The comparison efficiency of each work relative to Instruction-Mining and InstructionGPT-4.

Figure 4 shows that the SEM-specific work exempted from indirect preference tends to achieve high efficiency. All the superior works use the BM LLM loss to measure the quality according to SEM preference instead of human or Oracle LLM preference, while the three least efficient works use the Oracle LLM score. Specifically, IFD achieves the highest efficiency by leveraging internal information, which is independent of all outer preferences and biases. LESS is far worse because it resorts to leveraging external information, which introduces human preferences. AlpaGasus is the worst because it solely relies on Oracle LLM to generate quality scores, whose preference deviates far from the SEM.

6.2 Feasibility of the Selector

Superior work should not only have high efficiency but also high feasibility. Feasibility employs the simplicity and the flexibility of the selector to respectively assess its implementation difficulty and competence in handling new selection tasks. We develop the feasibility rank of each work in Table 2, based on the consideration of these two aspects.

Simplicity. We evaluate the simplicity of method from its cost of implementation and reproducibility. The cost of implementing a method focuses on the training and inference cost of model,

Methods	Feasibility Rank	Simplicity	Flexibility
AlpaGasus	1	1	1
InsTag	2	3	1
Nugget	2	2	2
IFD	3	4	4
Superfilter	3	4	4
LESS	3	4	4
DEITA	4	6	3
Instruction-Mining	4	7	2
InstructionGPT-4	5	5	5

Table 2: The feasibility rank, and the corresponding simplicity and flexibility rank of each work. A smaller number indicates the work does better.

as the algorithm cost is so much smaller that it can be neglected. The cost of implementation considers the number of time involves model training and inference, and the actual number of models being trained. The reproducibility considers whether the implementation details is provided and code is open-sourced. Table 4 in Appendix A.5 shows the simplicity rank and the above considerations of each work. Specifically, the work uses oracle LLM score as quality measurement is with high simplicity, where AlpaGasus is the simpliest, who involves 1 time of model inference.

Flexibility. Flexibility evaluates the extensibility and transferability of the selector. Extensibility examines the flexibility of reforming the model used in the selector, and transferability considers whether the work is dataset-independent or model-independent when dealing with migration tasks. Table 5 in Appendix A.5 shows the flexibility rank and the two considerations of each work, which are respectively derived from two questions: (1) Whether substitute the SOTA model used in the selector into open-sourced model defunct the selector; (2). Whether the model used in the selector needs to be retrained to maintain the optimal efficiency in handling new candidate datasets or SEM. Specifically, all the works that rely on oracle LLM to acquire quality scores are model inextensible, while six works using PLM loss are model dependent(Li et al., 2024b; Xia et al., 2024; Wei et al., 2023; Cao et al., 2023; Li et al., 2023).

6.3 Overall Consideration of the Selector

According to the overall performance of the works on efficiency and feasibility, we find out that: (1) The model-specific method achieves high efficiency without cost of feasibility; (2) Complex method can improve the efficiency, if it is deliberately designed to avoid the negative effect on model-specification, but it is always accompanied with the feasibility loss.

Specifically, We divide the works into a group using oracle LLM score (Li et al., 2023, 2024a; Xia et al., 2024; Cao et al., 2023; Wei et al., 2023; Li et al., 2024b) and a group using PFM loss (Lu et al., 2023; Chen et al., 2024; Liu et al., 2023), based on the different choice of data quality measurement. The PFM loss is model-specific which examine the loss of PFM generated by data, while orcale LLM score is human-basic which leverages oracle LLM to imitate human scoring. The work in PFM loss group is always more efficient than the work in another group With the same feasibility rank, indicating that PFM loss is superior to oracle LLM score.

Moreover, these two groups have opposite relationship between efficiency and feasibility. The work using oracle LLm score has a trade-off relationship between these two aspects. AlpaGasus has the worst efficiency but highest feasibility, while DEITA and InsTag sacrifice feasibility to achieve better efficiency by using more complex methods. By contrast, in the group of BM LLM loss, the one with high efficiency typically has low feasibility. Among these works, IFD manages to achieve both high efficiency and feasibility, because it only resorts to the internal information without introducing redundant element during filtering. LESS, who leverages on an outer dataset to filter data, does worse on both aspects. The simple work tends to achieve high efficiency, because they are direct who truncate the irrelevant and detrimental information from data selection. The result shows that the complex method may be beneficial or detrimental to the efficiency, which depends on its impact on the directness of method.

7 Discussions

7.1 Trend

Figure 5 shows the trend of data selector. To achieve superior overall performance, the data selector becomes more model-specific which reflects on the goal designation and the choice of quality approximation method. Specifically, the goal is changed from selector motivation to task motivation. The early works aim to develop the selector with high transferability to handle any data selection task (Cao et al., 2023; Wei et al., 2023; Chen et al., 2024), while the later works focus on improving the performance of SEM of the specific task (Li et al., 2023; Xia et al., 2024).

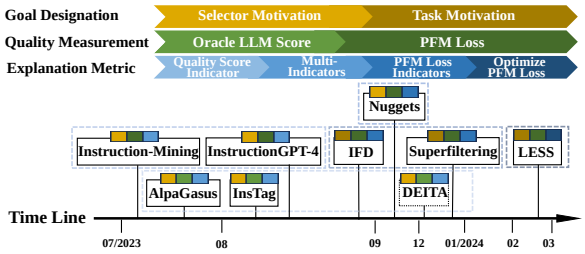

Figure 5: The timeline of the data selection methods.

The approximation method uses an explanation metric to predict the approximation object, which refers to the data quality label generated by the quality measurement. The quality measurement developed from relying on external oracle LLM score (Chen et al., 2024) to internal PFM loss (Xia et al., 2024) becomes more direct to the PFM, while the explanation metric becomes more complex from using concrete indicator to abstract indicator, and from using single to multiple indicators. The early works tend to use explicit indicators such as quality score and diversity, which introduce semantic factors to explain data quality, while the later works use abstract indicators developed from the PFM loss to reflect the data quality. On the other hand, the number of explicit indicators used in the selector increases. DEITA and InsTag employ more explicit indicators than AlpaGasus, which solely relies on quality scores from oracle LLM as quality indicators. Specifically, data diversity as the explicit indicator can largely improve the overall performance. DEITA and InsTag achieve far better overall performance than AlpaGasus because they take diversity into consideration.

7.2 Challenges

Through the above review and analysis, although there has been significant progress in data selection for fine-tuning LLMs, there are still three challenges.

(1) **Lack of unified and efficient metrics for high-quality data**. The existing data selection methods still have a vague definition of high quality. Although some methods consider explainable linguistic features or the complexity of data as well as the diversity of sampling, most of them focus on improving the performance of the model, that is, samples that can improve model performance are high-quality samples. Interestingly, recent work (Bai et al., 2024) has shown that some seemingly low-quality datasets considering their data sources also have improvement effects on LLM fine-tuning. Our analysis has shown that different methods have their own strengths, and the main challenge is to organically combine their advantages to build a unified and efficient data selection method for fine-tuning.

(2) **Lack of data selection methods for specific domains**. Most data selection methods focus on overall performance improvement, but the contribution of selected data to different domains is not the same. The existing works (Cao et al., 2023; Wei et al., 2023; Chen et al., 2024; Lu et al., 2023; Li et al., 2024b) demonstrated that selected data can bring significant improvements in writing and role-playing but minor improvements in mathematics and reasoning. Therefore, future work needs to consider how to dynamically select data based on the shortcomings of the model in a specific domain to compensate for its performance.

(3) **Lack of data selection methods for muti-turn conversations.** Most existing data selection methods are aimed at single-turn conversations because their quality is easier to measure but lacks attention to multi-turn conversation data. Although current work (Lu et al., 2023; Li et al., 2024b; Liu et al., 2023) evaluates models in static multi-turn conversation scenarios such as MT-Bench, there is a lack of suitable metrics for measuring multi-turn conversation data quality for data selection.

8 Conclusion

In this paper, we conducted an extensive survey on data selection for fine-tuning large-scale language models. We first construct a three-stage data selection scheme for the entire process and review the current research progress of data selection based on it, including data preprocessing, data selector construction, and data selector evaluation. To address the issue of incompatibility caused by different experimental settings, we propose a quantitative evaluation based on the assumption of efficiency curves to compare the existing work. We also qualitatively analyzed the feasibility of existing work, including implementation costs and reproducibility. We find that the model-specific work achieves high efficiency, whereas complex method can improve the efficiency if it is deliberately designed to avoid the negative effect on model-specification, but it is always accompanied by feasibility loss. We have summarized the existing trends and provided insights for future research.

Limitation

(1) In analyzing the efficiency of the selector, the two assumptions are proposed to make the efficiency of the method comparable. Because of the lack of information on the efficiency curve of the method, we use distance to demonstrate the efficiency of the methods, which generates the problem of infeasible area. Following the increase of work included in a comparable group, the infeasible area enlarges, which limits the number of works in the group.

(2) This paper mainly focuses on data selection for instruction fine-tuning LLMs instead of data rewriting or enhancement. Although we have already comprehensively examined the existing works, there may still be some works we neglected.

References

Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Haewon Jeong, Colin Raffel, Shiyu Chang, Tatsunori Hashimoto, and William Yang Wang. 2024. A survey on data selection for language models.

Yuelin Bai, Xinrun Du, Yiming Liang, Yonggang Jin, Ziqiang Liu, Junting Zhou, Tianyu Zheng, Xincheng Zhang, Nuo Ma, Zekun Wang, Ruibin Yuan, Haihong Wu, Hongquan Lin, Wenhao Huang, Jiajun Zhang, Wenhu Chen, Chenghua Lin, Jie Fu, Min Yang, Shiwen Ni, and Ge Zhang. 2024. Coig-cqia: Quality is all you need for chinese instruction fine-tuning.

Yihan Cao, Yanbin Kang, Chi Wang, and Lichao Sun. 2023. Instruction mining: When data mining meets large language model finetuning.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. 2024. Alpagasus: Training a better alpaca with fewer data.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

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.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Ming Li, Yong Zhang, Shwai He, Zhitao Li, Hongyu Zhao, Jianzong Wang, Ning Cheng, and Tianyi Zhou. 2024a. Superfiltering: Weak-to-strong data filtering for fast instruction-tuning.

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.

Yunshui Li, Binyuan Hui, Xiaobo Xia, Jiaxi Yang, Min Yang, Lei Zhang, Shuzheng Si, Junhao Liu, Tongliang Liu, Fei Huang, and Yongbin Li. 2024b. One shot learning as instruction data prospector for large language models.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning.

Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. 2023. instag: Instruction tagging for analyzing supervised fine-tuning of large language models.

Elizaveta Moskovskaya, Olesya Chebotareva, Valeria Efimova, and Sergey Muravyov. 2023. Predicting dataset size for neural network fine-tuning with a given quality in object detection task. *Procedia Computer Science*, 229:158–167.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. 2017. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Jiahao Wang, Bolin Zhang, Qianlong Du, Jiajun Zhang, and Dianhui Chu. 2024. A survey on data selection for llm instruction tuning.

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.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Lai Wei, Zihao Jiang, Weiran Huang, and Lichao Sun. 2023. Instructiongpt-4: A 200-instruction paradigm for fine-tuning minigpt-4.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning.

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.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

A Appendix

A.1 Efficiency and High Quality Dataset

Efficiency Definition The efficiency of work is the expectation of probability in selecting the ground truth high quality data at each bet, which is derived from the consideration on the quality structure of the selected subset. The quality structure refers to the proportion of high quality data to the size of dataset.

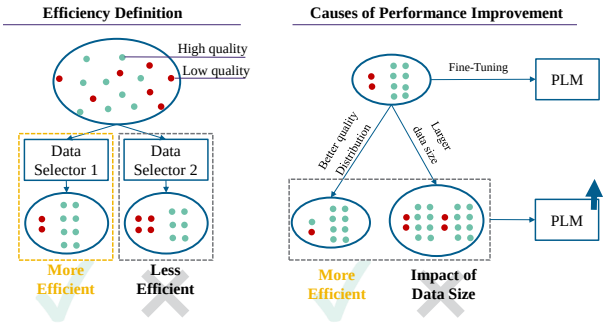

Figure 6: The left figure indicates the definition of efficiency is related to the quality distribution of selected subset. The right figure shows that the performance improvement of the corresponding fine-tuned PLM comes from both a better quality distribution and a larger data size

In Figure 6, data selector 1 and 2 selects two subsets with different quality structure from a candidate set. Subset 1 has better quality structure than subset 2, because data selector 1 has higher probability in selecting the high quality data at each bet, whose expectation is the proportion of high quality data to the size of subset.

Causes of Performance Improvement The quality of dataset decides the performance of PLM who is fine-tuned on it(Zhou et al., 2023). However, the performance also increases following the increase of data size, if the augmented data set has the same quality distribution with the original one (Chung et al., 2022).

A.2 Performance Improvement Ratio

Table 3 shows the performance improvement developed from the original performance values under 4 kinds of evaluation setting which we conclude and extract from the primitive article. Wins rate and benchmark improvement is repectively developed from the wine-ties-losses and benchmark scoring using the formula:

$$\frac{1}{n}\sum_{i=0}^{n}\frac{X_i}{Y_i} \tag{4}$$

where n is the total number of the evaluation settings using the same counterpart model with different evaluation metrics or candidate datasets, X_i and Y_i is respectively the score of the SEM and the counterpart model under the same evaluation setting i.

Then, the SEM Wins rate under Same Counterpart Model is chosen as the unified performance improvement ratio. Missing value is calculated by

Method	Same Counterpart Model (BM)		SEM	Other Counterpart Models (LLaMA chat 7B/13B)	
	Wins Rate	Benchmark Improvement		Wins Rate	Benchmark Improvement
AlpaGasus	1.284	0.949	LLaMA-2 7B	-	-
Superfiltering	1.475	0.962	LLaMA-2 7B	-	-
InsTag	1.344	-	LLaMA 13B	-	0.985
DEITA	1.467	-	LLaMA-2 13B	-	1.000
InstructionGPT-4	1.443	-	MiniGPT-4	-	-
Nuggets	1.519	-	LLaMA-2 7B	-	-
IFD	1.747	-	LLaMA-2 7B	-	-
LESS	-	0.973	LLaMA-2 13B	-	-
Instruction-Mining	-	-	LLaMA-2 7B	0.212	0.991

Table 3: The table shows the performance improvement under four evaluation settings which we deliberately choose and leverage on, to generate a unified performance improvement rate for each method. In the Other Counterpart Models, the parameter size of LLaMA chat is chosen as the same as the SEM to offset the impact of model size.

leveraging on the other method that bridges the value from other evaluation settings to the unified performance improvement ratio.

A.3 Assumption of Efficiency Curve

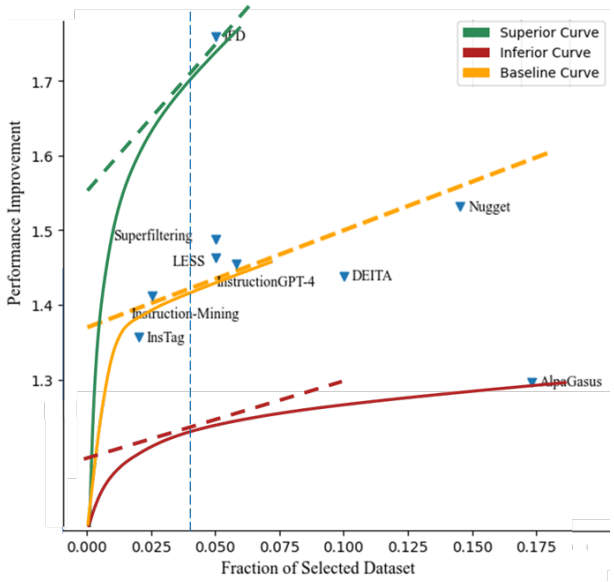

Figure 7: The demonstration of potential efficiency curve. The green, yellow, and red dashed lines represent the slopes of the methods at the same value of fraction.

We develop these assumptions to construct the efficiency curve of the work. The work on the same theoretical efficiency curve has the same efficiency, where the performance is purely affected by the impact of data size:

The first assumption. We assume for a dataset with fixed structure, its function of the performance improvement ratio and selected dataset fraction complys to the logrithem-like function which is upwarded, concaved, and approaching to linear after experiencing a rapid but short increasing.

The second assumption. We assume for two methods with different efficiency, the function's slope of the superior one is always larger than the inferior one in the whole feasible domain of the selected dataset fraction which is between 0 and 1.

Figure 7 demonstrates the potential efficiency curves of three works. As IFD is the most efficient and AlpaGasus is the least efficiency, the slope of IFD is larger than AlpaGasus at the same selected dataset fraction.

For the first assumption, many articles suggest that the impact of logarithm data size on the loss is linear, if the augmented dataset maintains the same quality structure (Kaplan et al., 2020; Sun et al., 2017; Moskovskaya et al., 2023). Assumption 1 extends this relationship to the pair of the performance improvement ratio and selected dataset fraction. Moreover, the statement of rapid but short growth complies with the fact that the slope of baseline (≈ 1.303) is far less than the slope ($= 56$) between Instruction-Mining and the original point, which implies a rapidly growth of performance at early stage.

The second assumption can be intuitively deduced from the first assumption with the fact that high quality data leads to better performance of SEM (Zhou et al., 2023). Therefore, if the methods is superior which indicate its selected dataset is with good structure, its increasing on the performance improvement must be greater than the inferior one at every point selected dataset fraction.

A.4 Infeasible Area

Because of a lack of information, each work generates an infeasible area, which is in fact the possible area of its efficiency curve. Therefore, if other work is in the infeasible area, it is incomparable with the work that generates this infeasible area. The infeasible of the inferior work and superior work is generated differently, where Figure 8 shows respectively the by using LESS and InsTag. For both superior and inferior works, the yellow boundary of the infeasible area is parallel to the baseline.

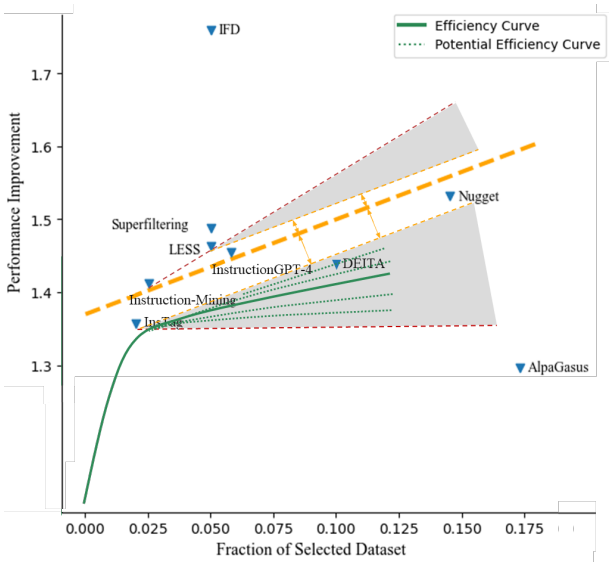

Figure 8: The demonstration of infeasible area. The green line is the efficiency curve of InsTag, where the dashed lines indicates its potential position.

For inferior work, the red boundary is horizontal because it cannot perform worse with a larger data size; For superior work, the red boundary is the line between the work and Instruction-Mining because the efficiency curve will never penetrate each other under the assumption.

A.5 Detailed Information of Feasibility

Table 4 shows the complexity rank and the corresponding considerations in structure difficulty and cost of running. Table 5 shows the flexibility rank and the corresponding considerations of each work.

Data Selection Method	Cost of Implementation			Reproducibility		Complexity Rank
	# Times involves Model Training	# Times involves Model Inference	# Models Being Trained	Implementation Details	Code Open Sourced	
AlpaGasus	0	0	1	✔	✘	1
Nuggets	0	2	0	✔	✔	2
InsTag	0	2	0	✔	✘	3
IFD	1	2	1	✔	✔	4
Superfilter	1	2	1	✔	✔	4
LESS	1	2	1	✔	✔	4
InstructionGPT-4	1	3	1	✔	✔	5
DEITA	2	3	1	✔	✔	6
Instruction-Mining	1	0	129	✔	✘	7

Table 4: The complexity rank and the corresponding considerations in structure difficulty and cost of running.

Method	Extensiblility	Transferability		Flexibility Rank
	Model Independent	Dataset Independent	Model Independent	
InsTag	✘	✔	✔	1
AlpaGasus	✘	✔	✔	1
Nuggets	✔	✔	✘	2
Instruction-Mining	✔	✔	✘	2
DEITA	✘	✘	✔	3
LESS	✔	✘	✘	4
Superfiltering	✔	✘	✘	4
IFD	✔	✘	✘	4
InstructionGPT-4	✘	✘	✘	5

Table 5: The feaibility rank and the corresponding considerations. In the real practicing, the indicator under Extensibility and Transferability has different priority. Therefore, we consider they contributes differently to the rank of feasibility, if the work has the same number of ✔. The priority from the highest to lowest is: Model Independent in Transferability, Dataset Independent, Model Independent in Extensibility.