FIRE🔥: Flexible Integration of Data Quality Ratings for Effective Pretraining

Liangyu Xu[4*], **Xuemiao Zhang**[1,4*], **Feiyu Duan**[2,4*], **Sirui Wang**[3,4†], **Rongxiang Weng**[4], **Jingang Wang**[4], **Xunliang Cai**[4]

[1] Peking University [2] Beihang University [3] Tsinghua University [4] Meituan

{xuliangyu02, wangsirui, wangjingang02, caixunliang}@meituan.com
zhangxuemiao@pku.edu.cn duanfeiyu@buaa.edu.cn

Abstract

Selecting high-quality data can improve the pretraining efficiency of large language models (LLMs). Existing methods generally rely on heuristic techniques or single quality signals, limiting their ability to evaluate data quality comprehensively. In this work, we propose FIRE, a flexible and scalable framework for integrating multiple data quality raters, which allows for a comprehensive assessment of data quality across various dimensions. FIRE aligns multiple quality signals into a unified space, and integrates diverse data quality raters to provide a comprehensive quality signal for each data point. Further, we introduce a progressive data selection scheme based on FIRE that iteratively refines the selection of high-quality data points. Extensive experiments show that FIRE outperforms other data selection methods and significantly boosts pretrained model performance across a wide range of downstream tasks, while requiring less than 37.5% tokens needed by the *Random* baseline to reach the target performance.

1 Introduction

Large language models (LLMs) have demonstrated remarkable performance by utilizing large-scale Transformers to pretrain on trillions of tokens. However, due to the constraints imposed by scaling laws (Kaplan et al., 2020), LLMs are quickly nearing their capacity and data limits. As a result, efforts to improve LLM performance have increasingly concentrated on optimizing the quality of pretraining data.

Numerous studies indicate that effective data selection can significantly enhance the convergence speed and generalization capability of LLMs (Engstrom et al., 2024; Wettig et al., 2024; Gao et al., 2025). Traditional methods predominantly rely on heuristic techniques, such as rule-based filtering (Rae et al., 2021; Raffel et al., 2020), deduplication (Abbas et al., 2023; Tirumala et al., 2024), and assessing proximity to high-quality corpora (Xie et al., 2023). Additionally, some work has focused on improving the evaluation of pretraining data quality by querying authoritative LLMs to determine whether the texts meet specific criteria (Wettig et al., 2024; Sachdeva et al., 2024).

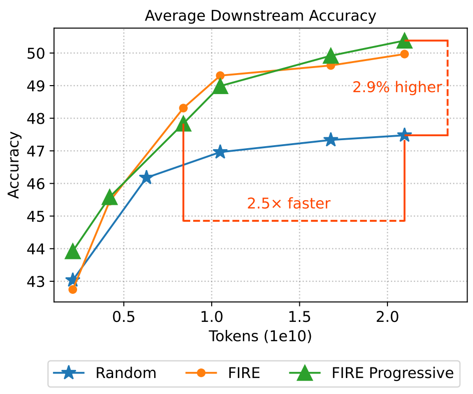


Figure 1: Downstream accuracy with respect to pretraining tokens for Random, FIRE, and FIRE Progressive.

Intuitively, assessing the quality of a text involves analyzing it across multiple dimensions. Nevertheless, the methods mentioned above evaluate data quality based on individual aspects, lacking a comprehensive assessment of the data's overall quality. Building on this limitation, the success of querying LLMs (Sachdeva et al., 2024) has inspired the straightforward idea of merging various quality rating criteria into a single prompt to collect comprehensive quality signals from authoritative LLMs. However, experimental findings show that this approach considerably weakens the performance of LLMs, as the excessive number of rules makes it challenging for LLMs to follow (further details can be found in Appendix E.2). The challenge of adhering to multiple rules underscores the necessity for a more sophisticated strategy to integrate various

*Equal contribution.
†Corresponding author.

quality signals effectively.

In this paper, we propose FIRE, a **F**lexible and scalable framework for **I**ntegrating multiple data quality **R**aters, designed to enable **E**ffective pretraining of LLMs. Initially, we introduce an alignment method to tackle the issue of inconsistent ratings from multiple raters. This method involves ranking the data based on the scores from the original raters and then partitioning it into quantiles. We assess the probability that the data within each quantile is of higher quality (win rate) compared to a reference subset, using this probability as the aligned rating. By fitting a win-rate-quantile curve for each rater, we effectively map the ratings from multiple raters into a unified rating space. Subsequently, to derive a comprehensive signal representing the overall data quality, we integrate the aligned ratings of multiple raters, considering both the intrinsic reliability and orthogonality of the raters. Further, we introduce a progressive data selection scheme based on FIRE that iteratively refines the selection of high-quality data points, balancing computational complexity with the refinement of orthogonality.

Extensive experiments demonstrate that by applying integrated ratings from multiple raters, our method achieves superior results across a variety of downstream tasks. Figure 1 illustrates that FIRE significantly enhances the pretrained model. We summarize our main contributions as follows:

(1) We propose FIRE, a flexible and scalable framework for integrating multiple data quality raters. FIRE aligns ratings from multiple raters into a unified space and integrates them to provide a comprehensive quality signal for each data point.

(2) We introduce a progressive data selection scheme based on FIRE that iteratively refines the selection of high-quality data points. It achieves a balance between computational complexity and the refinement of orthogonality.

(3) Extensive experiments demonstrate that FIRE enhances the pretrained model's performance by an average of 2.9%, while requiring less than 37.5% of the training data needed by the *Random* baseline to reach the target performance.

2 FIRE: Flexible Integration of Quality Ratings

2.1 Overview of the Method

We propose FIRE, a method that flexibly integrates multiple raters to comprehensively evaluate data quality. It involves two key processes: **(a) Rating Alignment** and **(b) Rater Integration**. Figure 2 illustrates the overall framework of FIRE. Many off-the-shelf raters exist in practice. To be integrated by FIRE, a rater must provide a scalar score for each data point and be empirically validated for effectiveness.

First, we propose an alignment method to map ratings from multiple raters into a unified rating space. Specifically, we involve the probability that the data in each quantile is of higher quality (win rate) compared to a reference subset as the aligned rating. By fitting a win-rate-quantile curve for each rater, we effectively map the ratings from multiple raters into a unified rating space. It's worth noting that the alignment process allows us to quantify the intrinsic reliability of each rater, defined by the win rate of the best data subset selected by the rater relative to the reference subset, thereby reflecting the rater's performance.

We then integrate the aligned ratings of multiple raters, considering both the intrinsic reliability and orthogonality of the raters. We construct an orthogonality graph and calculate centrality through PageRank(Page et al., 1999) to quantify the independence among raters. The integrated rating is given by:

$$I(x) = \mathbf{A}(x)^T(\mathbf{o} \odot \boldsymbol{\gamma}) \tag{1}$$

where $\mathbf{A}(x) \in \mathbb{R}^n$ is the vector of aligned ratings for data point x from n raters, $\boldsymbol{\gamma} \in \mathbb{R}^n$ is the vector of intrinsic reliability scores, $\mathbf{o} \in \mathbb{R}^n$ denotes the overall orthogonality scores, and $\odot$ represents the element-wise (Hadamard) product. The result $I(x) \in \mathbb{R}$ is a scalar that reflects the final integrated quality score of the data point.

2.2 Rating Alignment

Aligning ratings from multiple data quality raters is crucial for achieving a credible integrated rater. This process involves standardizing ratings to a consistent scale and eliminating the significant differences in raters' high-quality thresholds, which are the score thresholds that distinguish data contributing positively to pretraining. Appendix C.1 offers further analysis on the importance of rating alignment.

Given a pretraining dataset $\mathcal{D}_t$ and multiple raters $R_1, R_2, \ldots, R_n$, we propose a method to consolidate these ratings into a unified rating space:

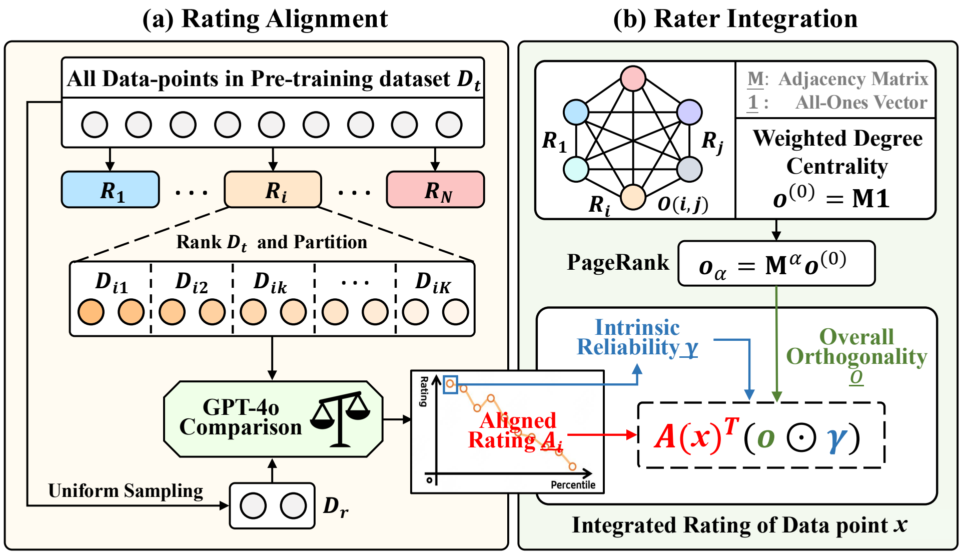


Figure 2: Overview of the proposed FIRE framework, which consists of two main components: (a) **Rating Alignment**, which maps scores from different quality raters into a unified rating space using win-rate estimation against a reference subset; (b) **Rater Integration**, which combines aligned ratings by weighting them with both the *intrinsic reliability* (γ) and *orthogonality* ($\mathbf{o}$) of each rater.

Step 1: Sample reference subset. Uniformly sample a subset $\mathcal{D}_r$ from the pretraining data $\mathcal{D}_t$. It is worth noting that the data quality distributions of $\mathcal{D}_r$ and $\mathcal{D}_t$ are consistent. This consistency allows us to use $\mathcal{D}_r$ as a representative reference set: if a subset of data is of higher quality than data in $\mathcal{D}_r$, it can be considered to exceed the typical quality of $\mathcal{D}_t$, and vice versa.

Step 2: Sort and partition data. Sort $\mathcal{D}_t$ according to R_i, and partition the sorted data into k intervals.

Step 3: Compare and calculate win rates. Randomly sample a subset $\mathcal{D}_{ij}$ from each interval, ensuring that $|\mathcal{D}_{ij}| = |\mathcal{D}_r|$. Use GPT-4o (Islam and Moushi, 2024) to evaluate how the $\mathcal{D}_{ij}$ samples impact pretraining in comparison to the reference dataset $\mathcal{D}_r$. Then, calculate the win rate w_{ij} for each interval j:

$$w_{ij} = \frac{|\{x \in \mathcal{D}_{ij} \mid \text{GPT-4o: } x > y,\ y \sim \mathcal{D}_r\}|}{|\mathcal{D}_{ij}|} \tag{2}$$

where $\mathcal{D}_{ij}$ signifies the subset sampled from the j-th rating interval for rater R_i. The win rate w_{ij} is the proportion of samples x in $\mathcal{D}_{ij}$ that GPT-4o determines have higher quality than the comparison sample y from $\mathcal{D}_r$. Calculating win rates relative to $\mathcal{D}_r$ makes ratings from multiple raters comparable. Moreover, since we sample only 1,000 data points per interval for comparison, the computational cost of this process remains low. The prompt for GPT-4o is detailed in Appendix C.2. We demonstrate the reliability of using GPT-4o for quality comparison in Appendix C.3.

Step 4: Fitting a win-rate-percentile function. Employ w_{ij} as the aligned rating for the midpoint of j-th rating interval of R_i, denoted by (p_j, w_{ij}). Construct a continuous win rate-percentile function from these coordinates using polynomial spline interpolation (detailed in Appendix C.4).

For any data point, we can find the aligned rating from a specific rater by applying the rater's win-rate-percentile function to its original rating and percentile. We apply the alignment method to the ratings of 4 raters on the SlimPajama dataset. The win rates in different percentiles and the fitted functions are provided in Appendix C.5. It is worth noting that since w_{i0} represents the win rate of the best data subset selected by rater i relative to the reference subset, it reflects the performance of rater i to a certain extent. Therefore, we can use w_{i0} as the intrinsic reliability of rater i, i.e., $\gamma_i = W_{i0}$.

2.3 Rater Integration

Suppose for raters $R_1, R_2, \ldots, R_n$, each rater corresponds to a standard basis vector $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ in the quality space. The integrated quality vector

$\mathbf{q}(x)$ of data point x can be expressed as:

$$\mathbf{q}(x) = \sum_{j=1}^{n} \gamma_j A_j(x) \mathbf{v}_j \tag{3}$$

where γ_j denotes the intrinsic reliability of rater j, $A_k(x)$ denotes the rating for data point x from rater R_k after alignment. Ideally, if $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ form an orthogonal basis, it is reasonable to measure the overall quality of the data using the L1 norm of $\mathbf{q}(x)$, as it represents the sum of the scores of data point x across various orthogonal quality dimensions. However, $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$ are not necessarily completely independent. There may be raters R_i and R_j with a correlation coefficient $\rho > 0$ and directly adding the corresponding aligned ratings would increase the weight of a particular quality dimension. To mitigate this issue, we define $O(i, j)$ to quantify the orthogonality of two raters i and j (the formalization of $O(i, j)$ can be found in Appendix C.6). For a rater R_i, we apply the sum of its orthogonality with all other raters to weight its rating. If a rater is highly correlated with others, we use a lower orthogonality to penalize. So the integrated rating for data point x can expressed as

$$I(x) = \sum_{j=1}^{n} \gamma_j o_j A_j(x) \tag{4}$$

where $o_j = \sum_{\substack{k=1 \\ k \neq j}}^{n} O(j, k)$ is a quantification of the overall orthogonality R_j with other raters.

Inspired by Equation (4), we find that integration of ratings weighted by orthogonality can be formalized to the centrality problem of graph theory. Formally, we define orthogonality graph of a rater as follows:

Definition 1 (Orthogonality Graph). *An orthogonality graph is a complete graph where the vertices V_i represent the raters R_i. The weight of the edge between two vertices is the orthogonality $O(i, j)$ between the two raters.*

Based on Definition 1, we provide the following theorem:

Theorem 1. *The overall orthogonality o_i of a rater R_i with other raters can be quantified as **weighted degree centrality** of the corresponding vertex V_i in the orthogonality graph.*

We give Theorem 1's proof in Appendix B. Let $\mathbf{o} = [o_1, o_2, \dots, o_n]^T$ denote the overall orthogonality vector, where o_i represents the overall orthogonality of rater R_i. Define $\mathbf{M}$ as the adjacency matrix of the orthogonality graph, such that $\mathbf{M}_{ij} = O(i, j)$. Additionally, let $\mathbf{1}$ be the all-ones vector. We can then derive the following:

$$\mathbf{o}^{(0)} = \mathbf{M}\mathbf{1} \tag{5}$$

Considering that in a multi-rater setting, the independence of R_i might be affected by the orthogonality between different raters R_j and R_k, we propose an iterative formula, analogous to PageRank (Page et al., 1999):

$$\mathbf{o}^{(k+1)} = d\mathbf{M}\mathbf{o}^{(k)} + (1 - d)\mathbf{1} \tag{6}$$

where d is the damping factor and k denotes the k-th iteration. In PageRank, node centrality depends on edge weights. Since FIRE defines edge weights via orthogonality, the resulting scores reflect each node's overall independence. Since the introduction of the damping factor aims to address the rank sinks problem, which is not present in our graph, we find it reasonable to set $d = 1$. Assuming the number of iterations is α, and normalizing the final result, our final formula becomes:

$$\mathbf{o}_\alpha = \mathbf{M}^\alpha \mathbf{o}^{(0)} \tag{7}$$

$$\mathbf{o} = \frac{\mathbf{o}_\alpha}{\|\mathbf{o}_\alpha\|_2} \tag{8}$$

where $\|\cdot\|_2$ denotes the Euclidean Norm. In this paper we set $\alpha = 50$ since $\mathbf{o}_\alpha$ tends to stabilize after 50 iterations. The justification for employing Equation (7) and (8) to quantify the overall orthogonality is provided by Theorem 3 in Appendix B.[1]

Final version of the integrated rating for data point x can be expressed as:

$$I(x) = \mathbf{A}(x)^T (\mathbf{o} \odot \boldsymbol{\gamma}) \tag{9}$$

where $\mathbf{A}(x) = [A_1(x), A_2(x), \dots, A_n(x)]^T$ is the vector of aligned ratings for data point x from all raters, $\mathbf{o}$ is the overall orthogonality vector, $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \dots, \gamma_n]^T$ represents the intrinsic reliability vector, and $\odot$ denotes the Hadamard product.

3 Progressive Data Selection via FIRE

The most intuitive data selection method involves ranking the integrated ratings based on FIRE for the pretraining dataset, and then selecting the top k highest-rated data points. Nonetheless, our analysis shows that after ranking $\mathcal{D}_t$ based on the integrated

[1]When there's a complete correlation among some Raters, the orthogonality drops to zero. In such a case, we don't proceed with the rater integration and consider these multiple raters as one.

Algorithm 1 Progressive Data Selection Scheme

```
1:  Input: The entire dataset $\mathcal{D}_t$; Decay factor $\eta$; Initial number of parts $n$; Part multiplication factor $\beta$; Maximum number of parts $n_{\max}$; Desired dataset size $k$
2:  Output: $\mathcal{D}_s$: Selected data subset
3:  Calculate and sort integrated ratings for $\mathcal{D}_t$
4:  Reduce the data to $\eta\%$ of its original size using decay factor $\eta$
5:  while size of $\mathcal{D}_t \geq k$ do
6:      Divide the data into $n$ parts based on the quantiles of the integrated ratings
7:      for each part $P_i$ do
8:          Calculate the overall orthogonality in $P_i$
9:          Derive refined integrated ratings $\mathcal{S}_{P_i}$
10:     end for
11:     Sort the data based on the new integrated ratings $\mathcal{S}_{P_i}$
12:     Select the top $\eta\%$ data according to $\mathcal{S}_{P_i}$
13:     $n \leftarrow \min(n \times \beta, n_{\max})$
14: end while
15: $\mathcal{D}_s \leftarrow$ top k elements from $\mathcal{D}_t$
```

ratings, there is a change in the overall orthogonality **o** calculated from data subsets in different quantiles (Figure 10). The phenomenon arises because the top data better reflects the quality emphasized by the raters, while the tail data often contains more noise and low-quality information, leading to changed orthogonality among the raters. To refine the data selection process and mitigate the coarseness introduced by computing orthogonality on the entire dataset, we propose a progressive data selection scheme based on FIRE.

Specifically, as shown in Algorithm 1, we first calculate the integrated ratings for the data points in $\mathcal{D}_t$ based on FIRE, then sort the data points, but we don't select them right away based on these ratings. It is reduced to $\eta\%$ of its original size by selecting the top $\eta\%$ of data based on integrated ratings. Then, the data is partitioned into n segments based on integrated ratings' quantiles. Orthogonality is computed within each segment to determine refined integrated ratings. After sorting the data according to new ratings, it's further reduced to $\eta\%$ of its initial size. The number of segments is increased by a factor of β, unless it reaches the maximum threshold $n_{\max}$, in which case the data is divided into $n_{\max}$ segments. The iterative process of calculating orthogonality within progressively smaller subsets continues before the subsequent reduction leaves less than k data points for selection.

4 Experiments

4.1 Experimental setup

Setup We use SlimPajama (Soboleva et al., 2023) as the selection pool, with a total scale of 627B. And we employ the Llama (Touvron et al., 2023) tokenizer to divide the entire dataset into sequences of length 1024. During the data selection process, we select the top portion of data with the highest rating. For integrating different raters, we carry out experiments based on the four single raters of QuRating (Wettig et al., 2024): Writing Style, Facts and Trivia, Educational Value, and Required Expertise. For progressive data selection (FIRE Progressive), we set $\eta = 60, \beta = 20$. For model training, we train a model with 1.3B parameters for 10,000 steps (equivalent to 20B tokens) and a 3B model for 200B tokens, with bfloat16 format during both training and testing. The detailed model configurations are provided in Table 6 of Appendix D.1.

Evaluation We utilize lm-evaluation-harness (Gao et al., 2021) to assess the models' performance across eight downstream tasks: ARC-E (Clark et al., 2018), ARC-C (Clark et al., 2018), SciQ (Welbl et al., 2017), LogiQA (Liu et al., 2020), BoolQ (Clark et al., 2019), HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020), and WinoGrande (Sakaguchi et al., 2021). We employ in-context learning for the evaluation, selecting enough examples to fill the window length of 1024 tokens for each task. Standard accuracy metrics are reported for all tasks.

Baselines In addition to comparing FIRE with four single raters from Qurating, we also compare it with the following methods: (1) *Random*: randomly selecting data from the original training corpus. (2) *DSIR* (Xie et al., 2023): utilizing importance sampling for data selection, and we chose Wikipedia and Books as target domains. (3) *Density* (Sachdeva et al., 2024): using KDE to estimate data density in the training corpus and employing inverse sampling. (4) *ASK-LLM* (Sachdeva et al., 2024): using a comprehensive prompt to label high-quality data, and train a T5-based classifier.

Furthermore, we compare several basic rating integration methods: (1) *Comprehensive Rater*: integrating multiple single rater criteria into a single prompt to obtain annotations from GPT-4o, then

Method		ARC-E	ARC-C	SciQ	LogiQA	BoolQ	HellaSw.	PIQA	W.G.	AVG.
Baseline	Random	48.2	22.3	84.5	19.7	60.8	32.1	63.5	49.2	47.5
	DSIR with Book	36.2	19.5	73.4	21.4	61.8	29.5	62.5	**53.6**	44.7
	DSIR with Wiki	37.2	18.0	76.4	23.0	58.0	27.9	57.3	51.1	43.6
	Density	47.2	20.0	81.7	20.3	61.5	31.4	**66.3**	51.4	47.5
	ASK-LLM	52.6	24.8	80.2	22.1	**62.2**	28.9	59.5	50.2	47.6
Baseline (1 Rater)	QuRating (W.S.)	47.5	21.4	81.8	21.3	61.3	31.3	62.7	52.5	47.5
	QuRating (R.E.)	50.6	23.2	83.9	22.6	61.4	30.2	59.8	49.8	47.7
	QuRating (F.T.)	54.1	23.0	83.5	22.0	60.9	30.4	59.5	51.7	48.1
	QuRating (E.V.)	50.1	21.6	84.4	20.9	**62.2**	31.9	61.2	48.8	47.6
Baseline (Integration method, 4 Raters)	Comp. Rater	52.9	24.3	81.2	22.0	62.0	30.9	59.2	50.1	47.8
	Max Criteria	54.1	22.6	83.3	22.7	61.3	30.8	59.8	48.9	47.9
	Average	48.7	23.5	83.4	21.4	59.8	30.1	58.8	51.1	47.1
	Mix Criteria†	49.6	22.1	83.6	25.7	61.8	29.7	58.6	50.4	47.7
Raters Integration	FIRE (2 Raters)	55.4	24.9	83.6	21.3	60.0	31.6	60.1	50.4	48.4
	FIRE (3 Raters)	58.4	25.7	85.1	**23.1**	59.8	32.3	61.2	51.2	49.6
	FIRE (4 Raters)	59.1	26.4	86.0	21.0	61.8	32.9	59.7	52.8	50.0
	FIRE (4 Raters) Prg.	**59.2**	**27.0**	**86.9**	23.0	60.2	**33.0**	62.4	51.6	**50.4**

Table 1: Downstream tasks results for different rating integration method. We report accuracy for each task, and the best performances are marked in bold. For rater integration, we report the average score of all the combinations. Detailed results can be found in the Appendix E.1. Abbreviations: HellaSw. = HellaSwag, W.G. = WinoGrande, AVG. = Average, W.S. = Writing Style, R.E. = Required Expertise, F.T. = Facts and Trivia, E.V. = Educational Value, Prg = Progressive, Comp. = Comprehensive. †: We implement the method from QuRating(Wettig et al., 2024).

training a comprehensive quality rater. (2) *Max Criteria*: aligning ratings and selecting the highest value in each dimension as the final rating. (3) *Average*: arithmetic mean integration of normalized ratings from each rater. (4) *Mix Criteria*: we follow QuRating(Wettig et al., 2024) to merge and deduplicate the top data selected by each single rater, followed by random sampling. These four methods are applied to the integration of four raters. More details can be found in Appendix D.2.

4.2 Main Results

Table 1 show our main results. We find that:

FIRE is superior to other integration methods. FIRE demonstrates greater effectiveness than existing integration methods, while introducing minimal additional computational cost (see Appendix D.3 for a detailed analysis of computational cost). As shown in Table 1, *Comprehensive Rater* with a multi-dimension prompt results in an average score that is even lower than the single-dimension rater *Facts and Trivia*. This suggests that GPT-4o still falls short in assessing data quality from a broad perspective. Both *Mix Criteria* and *Max Criteria* are inferior to FIRE, indicating that a comprehensive evaluation of FIRE is more beneficial. *Average* simply calculates the mean of all ratings and the experimental results of FIRE demonstrate an improvement over *Average*. To demonstrate the robustness of FIRE, we conduct additional experiments integrating other raters. Detailed results are provided in Appendix D.4.

FIRE outperforms the single raters and other data selection methods. Comparing FIRE to the individual raters, it demonstrates significant improvements, with an average score increase of up to 1.9% over the best single rater and 2.9% over random selection. From a data efficiency perspective, FIRE achieves comparable performance using less than 37.5% of the data required by the *Random* baseline. Additionally, our method outperforms *QuRating* and other data selection methods, validating the high quality of the data selected by FIRE.

Adding more raters can lead to better performance. We observe that as the number of integrated raters increases, the overall effect gradually improves. The average score of FIRE with three raters surpasses that of the integration of two raters, and further increases when four raters are integrated. This indicates that our rater integration method is scalable: incorporating a broader range of raters not only provides a more comprehensive evaluation of the samples but also allows for a better understanding of the importance of each metric.

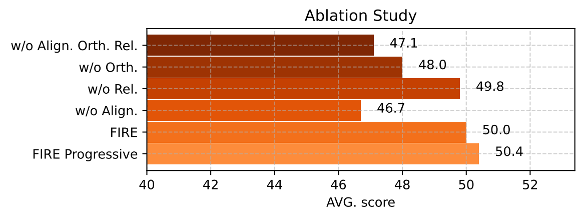


Figure 3: Ablation experiments on the impact of different rating integration strategies in FIRE.

Progressive selection further improves FIRE After integrating progressive selection, we observe a notable improvement in the model's performance on downstream tasks. Compared to FIRE (4 Raters), the average score of FIRE (4 Raters) Prg. increases by 0.4%. The most significant improvement is seen in PIQA, with an absolute score boost of 2.7%. These results validate the effectiveness of the progressive selection method in choosing high-quality data.

4.3 Analysis

Ablation Study We integrate four single raters, and subsequently remove *Rating Alignment (Align.)*, *Intrinsic Reliability (Rel.)*, and *Orthogonality (Orth.)*, as well as remove all (directly averaging on the rating post-normalization), in order to investigate the impact of each component in FIRE.

From Figure 3, we can find that: (1) Rating alignment is a crucial step. We note that without rating alignment, the score drops by 3.3%, even falling below the direct average. As we previously detailed, the alignment allows for better comparability between the ratings from different raters, making their integration more reasonable. (2) Both orthogonality and intrinsic reliability can further enhance model performance, with the impact of orthogonality being relatively significant (a drop of 2% w/o Orth.), while the improvement from Intrinsic Reliability is rather subtle (a drop of 0.2% w/o Rel.) (3) The combination of all components yields the best results. This implies that these methods of integrating the ratings are complementary. By superimposing both methods, we can achieve a rating that more accurately reflects the actual contribution of the sample to the pretraining.

Training Efficiency Figures 1 and 4 show how the model's performance on downstream tasks evolves with the pretraining tokens. In terms of average score, our method outperforms the random baseline by 2.9%. From the training efficiency perspective, our method reduces training tokens to achieve a certain performance level by more than half. In addition, our method shows a significant advantage in the ARC-E/C and SciQ tasks, consistently scoring higher than the random baseline. However, on the HellaSwag task, our method's performance is similar to the random baseline and does not consistently surpass it. One possible explanation is that HellaSwag is an especially challenging dataset, which makes it difficult to discern performance differences on 1.3B model.

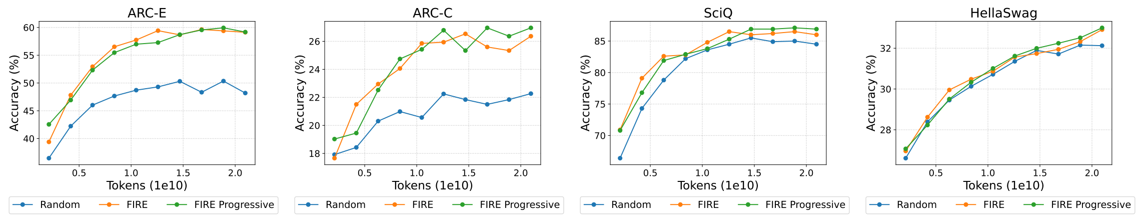
Method	FLOPS	ARC-C	HellaSwag	AVG.
Model Size = 1.3B				
Random	32.2×10^{19}	23.6	**34.4**	48.6
FIRE	16.1×10^{19}	**26.4**	32.9	**50.0**
Model Size = 3B				
Random	377.5×10^{19}	26.8	49.4	54.1
FIRE	377.5×10^{19}	**28.8**	**51.7**	**55.7**

Table 2: Results on larger models and datasets.

Larger datasets and models To validate our method on larger models and datasets, we conduct several additional experiments: (1) Training a 1.3B parameter model for 40B tokens using randomly sampled data; (2) Training a 3B parameter model for 200B tokens for both random sampling and FIRE (four raters integration). As illustrated in Table 2, the results indicate that the FIRE method outperforms *Random* with fewer training FLOPS in the 1.3B parameter model setting. Furthermore, in the 3B parameter model setting, FIRE exceeds *Random* by an average of 1.6%, demonstrating the robustness and scalability of our method with larger models and training datasets.

Ablation Study for Progressive Selection We conduct an ablation study on the partition multiplier factor β for the FIRE Progressive approach, with the outcomes shown in Figure 5. The results show that for a majority of β values, FIRE Progressive scores surpass those of FIRE and Random, suggesting that the progressive selection method contributes to a consistent enhancement of the FIRE framework.

Case study We extract 1M samples from the corpus and compute the pairwise Pearson correlation of the ratings across all dimensions. As illustrated in Figure 6, the FIRE rating exhibits a strong correlation with all other ratings, confirming the effectiveness of the FIRE framework in consolidating



(a) ARC-E (b) ARC-C (c) SciQ (d) HellaSwag

Figure 4: The in-context learning results with respect to pretraining tokens on four downstream tasks: ARC-E, ARC-C, SciQ, and HellaSwag.

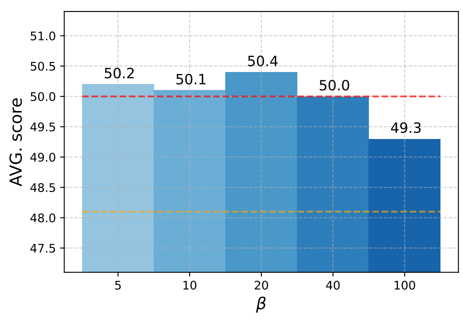
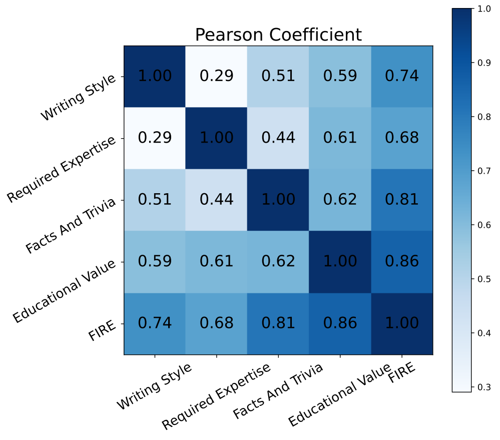


Figure 5: The impact of the partition multiplier factor β on FIRE Progressive performance. The red and orange dashed lines respectively represent the scores of FIRE and Random.



	Writing Style	Required Expertise	Facts And Trivia	Educational Value	FIRE
Writing Style	1.00	0.29	0.51	0.59	0.74
Required Expertise	0.29	1.00	0.44	0.61	0.68
Facts And Trivia	0.51	0.44	1.00	0.62	0.81
Educational Value	0.59	0.61	0.62	1.00	0.86
FIRE	0.74	0.68	0.81	0.86	1.00

Figure 6: Pearson correlation between different raters.

ratings across multiple dimensions. By employing the FIRE framework, we can effectively select data that exhibits high ratings across all dimensions.

Moreover, we analyze the data properties that FIRE focuses on compared to other raters. Based on the four dimensions in QuRating and a comprehensive dimension, we determine the percentage of high-quality data selected by each rater in each dimension. We pick 1000 data points at random from the top 20B data that are selected by each method. Then we use GPT-4o to assess each dimension six

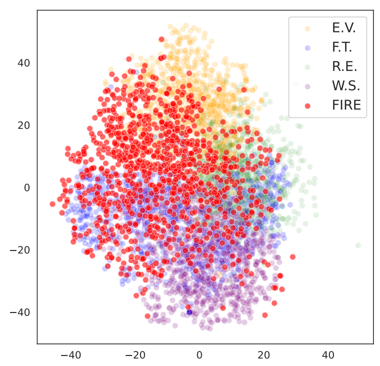


Figure 7: Illustration of top samples rated by each method. We use Sentence-T5 to encode each text, and employ t-SNE to perform dimensionality reduction on dense vectors.

Rater/Dimension	W.S.	R.E.	F.T.	E.V.	Comp.
W.S.	**94.0**	53.5	62.9	25.6	84.1
R.E.	39.0	85.8	93.8	**97.1**	98.0
F.T.	46.0	**97.3**	88.7	42.2	98.3
E.V.	50.8	76.3	<u>96.5</u>	47.1	99.1
FIRE	<u>58.2</u>	<u>90.5</u>	**97.3**	<u>54.1</u>	**99.5**

Table 3: The percentage of high-quality data across various dimensions, for top data selected by each rater. Underlined scores indicate the second highest. Comp. = Comprehensive.

times, taking the average as the final evaluation result. Refer to the Appendix D.5 for more details. As shown in Table 3: (1) From a comprehensive perspective, FIRE shows the best results, as evidenced by our performance in downstream tasks. (2) In terms of each dimension, FIRE consistently achieves relatively high accuracy, demonstrating that the data selected by this method maintains high quality across all dimensions.

To assess whether FIRE selects more diverse samples than single raters, we extract the top 1000 texts from each method and encode them using Sentence-T5 (Ni et al., 2022). We then apply t-SNE (Van der Maaten and Hinton, 2008) for dimensionality reduction and visualize the results in Figure 7. In the latent semantic space, samples selected by single raters appear more clustered, while those selected by FIRE are more broadly distributed, indicating higher diversity. Combined with prior case studies showing high quality, FIRE demonstrates a strong balance between diversity and quality.

Computational Cost We further analyze the computational cost (measured in FLOPs) of different data rating and integration methods, focusing on the overhead introduced by applying raters to the pretraining data (excluding the main model's pretraining cost). As shown in Table 4, FIRE achieves more effective integration without introducing significant extra cost beyond the base raters. In particular, FIRE and QuRating (mix of criteria) have similar FLOPs, both of which are several times higher than the single-rater setting, but FIRE provides substantially better downstream performance at comparable cost. The detailed derivation of the computational cost is provided in Appendix D.3.

Method	FLOPs
QuRating (single rater)	$\approx 8.17 \times 10^{20}$
QuRating (mix of criteria, 4 raters)	$\approx 3.26 \times 10^{21}$
FIRE (mix of criteria, 4 raters)	$\approx 3.26 \times 10^{21}$

Table 4: Comparison of computational cost (FLOPs) across different rating methods.

5 Related Works

When pretraining language models, a large amount of text corpus is often crawled from the internet. However, several studies (Li et al., 2023; Zhou et al., 2024; Duan et al., 2025) suggest that high-quality data is more beneficial to the model's performance. To select high-quality data, a common strategy involves utilizing rules crafted by experts (Raffel et al., 2020; Rae et al., 2021; Laurençon et al., 2022; Computer, 2023; Penedo et al., 2024) and removing duplicate sentences (Lee et al., 2022; Sorscher et al., 2022; Abbas et al., 2023; Soboleva et al., 2023; Tirumala et al., 2024). However, they often fall short in effectively selecting high-quality data based on semantic content. An alternative approach involves utilizing a target data source or proxy model (Wenzek et al., 2020; Xie et al., 2023; Marion et al., 2023; Thakkar et al., 2023; Engstrom et al., 2024; Yu et al., 2024).

Training a classifier is a more straightforward method (Du et al., 2022; Gururangan et al., 2022; Zhang et al., 2024; Wettig et al., 2024; Sachdeva et al., 2024). Du et al. (2022) implemented a logistic regression binary classifier to score the data, while some studies train more complex scorers (Zhang et al., 2024; Sachdeva et al., 2024). Additionally, QuRating (Wettig et al., 2024) trains multiple raters with a finer-grained approach to analyze the contribution of data to model performance improvement from different dimensions. Other studies(Zhang et al., 2025b,a) explore methods to boost pretraining efficiency by curriculum learning.

Prior methods mainly select data from a single perspective. Although QuRating introduces multidimensional raters, it does not systematically address their integration—a challenge our work explicitly tackles.

6 Conclusion

We propose FIRE, a flexible and scalable framework that integrates multiple data quality raters for comprehensive, multi-dimensional data assessment. First, ratings from different dimensions are aligned into a unified space. Then, orthogonality is introduced to adjust rater weights. To handle orthogonality variations across rating ranks, we adopt a progressive approach for fine-grained data selection. Experiments on the SlimPajama dataset show that FIRE outperforms other selection methods, substantially improving pretrained model performance across diverse downstream tasks.

7 Limitations and Future Works

Linear assumption for orthogonality integration We hypothesize our integration on a linear additive relation. While this assumption simplifies the computations, it might limit our ability to capture complex interactions between different dimensions. Future research could explore incorporating nonlinear systems to adjust rater weights, potentially boosting performance.

Number of raters We've tested our method with four different raters and the results have been promising. To make our integration method more robust and reliable, future tests could include more raters from diverse dimensions, which would ultimately help us build a more resilient and versatile rating integration system.

References

Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S Morcos. 2023. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

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

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Together Computer. 2023. Redpajama: an open dataset for training large language models.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR.

Feiyu Duan, Xuemiao Zhang, Sirui Wang, Haoran Que, Yuqi Liu, Wenge Rong, and Xunliang Cai. 2025. Enhancing llms via high-knowledge data selection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(22):23832–23840.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. 2024. Dsdm: Model-aware dataset selection with datamodels. *arXiv preprint arXiv:2401.12926*.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, et al. 2021. A framework for few-shot language model evaluation. *Version v0. 0.1. Sept*, 10:8–9.

Tianyu Gao, Alexander Wettig, Luxi He, Yihe Dong, Sadhika Malladi, and Danqi Chen. 2025. Metadata conditioning accelerates language model pre-training. *Preprint*, arXiv:2501.01956.

Suchin Gururangan, Dallas Card, Sarah Dreier, Emily Gade, Leroy Wang, Zeyu Wang, Luke Zettlemoyer, and Noah A Smith. 2022. Whose language counts as high quality? measuring language ideologies in text data selection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2562–2580.

Raisa Islam and Owana Marzia Moushi. 2024. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*.

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*.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, et al. 2022. The bigscience roots corpus: A 1.6 tb composite multilingual dataset. *Advances in Neural Information Processing Systems*, 35:31809–31826.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint arXiv:2007.08124*.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. 2023. When less is more: Investigating data pruning for pretraining llms at scale. *arXiv preprint arXiv:2309.04564*.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford infolab.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Noveen Sachdeva, Benjamin Coleman, Wang-Cheng Kang, Jianmo Ni, Lichan Hong, Ed H Chi, James Caverlee, Julian McAuley, and Derek Zhiyuan Cheng. 2024. How to train data-efficient llms. *arXiv preprint arXiv:2402.09668*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. 2023. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. 2022. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536.

Megh Thakkar, Tolga Bolukbasi, Sriram Ganapathy, Shikhar Vashishth, Sarath Chandar, and Partha Talukdar. 2023. Self-influence guided data reweighting for language model pre-training. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2033–2045.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. 2024. D4: Improving llm pretraining via document de-duplication and diversification. *Advances in Neural Information Processing Systems*, 36.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Johannes Welbl, Nelson F Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Édouard Grave. 2020. Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012.

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. 2024. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. 2023. Sheared llama: Accelerating language model pre-training via structured pruning. In *The Twelfth International Conference on Learning Representations*.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. 2023. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227.

Zichun Yu, Spandan Das, and Chenyan Xiong. 2024. Mates: Model-aware data selection for efficient pretraining with data influence models. *arXiv preprint arXiv:2406.06046*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*.

Xuemiao Zhang, Feiyu Duan, Liangyu Xu, Yongwei Zhou, Sirui Wang, Rongxiang Weng, Jingang Wang, and Xunliang Cai. 2025a. Frame: Boosting llms with a four-quadrant multi-stage pretraining strategy. *Preprint*, arXiv:2502.05551.

Xuemiao Zhang, Liangyu Xu, Feiyu Duan, Yongwei Zhou, Sirui Wang, Rongxiang Weng, Jingang Wang, and Xunliang Cai. 2025b. Preference curriculum: Llms should always be pretrained on their preferred data. *Preprint*, arXiv:2501.13126.

Yifan Zhang, Yifan Luo, Yang Yuan, and Andrew C Yao. 2024. Autonomous data selection with language models for mathematical texts. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

A Ethical Considerations

Training large language models (LLMs) demands a substantial amount of electrical power, resulting in significant carbon emissions. To address this issue, we aim to develop efficient data selection methods that reduce the computational resources required for model training, thereby mitigating environmental impact. Furthermore, by meticulously curating high-quality data, we can enhance model performance and minimize the occurrence of hallucinations. This not only improves the reliability of the models but also helps curb the spread of fake news and misinformation, addressing critical societal concerns.

B Proof of theorems

In this section, we present the proofs for the two previously mentioned theorems. Theorem 2 facilitates the transformation of orthogonality calculations into a centrality problem within a graph. Meanwhile, Theorem 3 rigorously demonstrates the convergence of our framework. Specifically, it establishes that after several iterations, the vector $\frac{\mathbf{o}}{||\mathbf{o}||_2}$ will assuredly converge to a fixed vector, thus precluding divergence.

Theorem 2. *The overall orthogonality o_i of a rater R_i with other raters can be quantified as **weighted degree centrality** of the corresponding vertex V_i in the orthogonality graph.*

Proof. Consider that the weighted degree centrality of vertex V_i is the sum of the weights of the edges connecting V_i to all vertices in the set of adjacent vertices A_i. Since the graph of orthogonality is a complete graph, we have

$$C(V_i) = \sum_{V_j \in A_i} O(i,j) = \sum_{\substack{j=1 \\ j \neq i}}^{n} O(i,j)$$

This is consistent with the definition of the overall orthogonality in Equation (4) of the main text. □

Theorem 3. *As $\alpha \to +\infty$, $\mathbf{o}$ will eventually converge to a fixed unit vector.*

Proof. Since the Graph of orthogonality is an undirected graph, the adjacency matrix $\mathbf{M}$ is a symmetric matrix. According to the Spectral Theorem for Symmetric Matrices, $\mathbf{M}$ can be diagonalized, and all corresponding eigenvectors can form an orthogonal basis.

Since for any i, j, $O(i,j) \geq 0$, we can deduce that $\mathbf{M}$ is a non-negative matrix. Additionally, since the Orthogonality Graph is a complete graph, $\mathbf{M}$ is an irreducible matrix. By the Perron-Frobenius Theorem, we obtain that:

$$\begin{aligned} &\exists \lambda \in \mathbb{R}, \lambda > 0 \\ &\text{s.t. } \lambda = \max\{\mu \mid \mu \in \sigma(\mathbf{M})\}, \end{aligned}$$

where $\sigma(\mathbf{M})$ denotes the set of eigenvalues of $\mathbf{M}$. Assume $\mathbf{M}$ has m eigenvalues $\lambda_1, \lambda_2, ..., \lambda_m$ arranged in descending order, where $\lambda_1 > 0$. Each eigenvalue λ_i corresponds to the eigenvectors $\mathbf{v}_{i1}, ..., \mathbf{v}_{ip_i}$, where p_i is the algebraic multiplicity of λ_i. We can decompose $\mathbf{o}^{(0)}$ into each eigenvector

$$\mathbf{o}^{(0)} = \sum_{i=1}^{m} \sum_{j=1}^{p_i} c_{ij} \mathbf{v}_{ij}$$

Therefore, we have

$$\begin{aligned} \mathbf{M}^{\alpha} \mathbf{o}^{(0)} &= \mathbf{M}^{\alpha} \sum_{i=1}^{m} \sum_{j=1}^{p_i} c_{ij} \mathbf{v}_{ij} \\ &= \sum_{i=1}^{m} \sum_{j=1}^{p_i} c_{ij} \mathbf{M}^{\alpha} \mathbf{v}_{ij} \\ &= \sum_{i=1}^{m} \sum_{j=1}^{p_i} c_{ij} {\lambda_i}^{\alpha} \mathbf{v}_{ij} \\ &= {\lambda_1}^{\alpha} \sum_{i=1}^{m} \sum_{j=1}^{p_i} c_{ij} (\frac{\lambda_i}{\lambda_1})^{\alpha} \mathbf{v}_{ij} \end{aligned}$$

Given that $\forall i \neq 1$, $|\frac{\lambda_i}{\lambda_1}| < 1$, thus

$$\lim_{\alpha \to +\infty} \left(\frac{\lambda_i}{\lambda_1}\right)^{\alpha} = 0$$

We obtain

$$\mathbf{o} = \lim_{\alpha \to +\infty} \mathbf{M}^{\alpha} \mathbf{o}^{(0)} = {\lambda_1}^{\alpha} \sum_{j=1}^{p_1} c_{1j} \mathbf{v}_{1j}$$

Thus

$$\begin{aligned} \frac{\mathbf{o}}{||\mathbf{o}||_2} &= \frac{{\lambda_1}^{\alpha} \sum_{j=1}^{p_1} c_{1j} \mathbf{v}_{1j}}{||{\lambda_1}^{\alpha} \sum_{j=1}^{p_1} c_{1j} \mathbf{v}_{1j}||_2} \\ &= \frac{\sum_{j=1}^{p_1} c_{1j} \mathbf{v}_{1j}}{|| \sum_{j=1}^{p_1} c_{1j} \mathbf{v}_{1j}||_2} \end{aligned}$$

The right-hand side of the formula is a fixed unit vector. Therefore, the theorem is proven. □

C FIRE Analysis

C.1 Analysis of the necessity of rating alignment

Multiple raters may employ different scales and criteria for assessing data quality, which can cause substantial problems if their ratings are integrated without appropriate standardization. For example, some raters may prioritize grammatical accuracy using a numerical scale, while others might assess semantic relevance using a percentage scale. Moreover, the rating thresholds distinguishing data that positively contribute to pretraining can significantly differ among raters. Without rating alignment, the integrated ratings can be misleading, inaccurately reflecting the actual quality of the data point. The subsequent examples and analyses underscore the importance of rating alignment for a reasonable data quality evaluation:

- Different Scales. Suppose we have Rater A, who assesses data quality on a 1 to 10 scale based on grammatical accuracy, and Rater B, who evaluates semantic relevance on a 0% to 100% scale. Let's say a particular data point receives an 8 from Rater A and 85% from Rater B. If we naively average these ratings, we get: $\frac{8+85}{2} = 46.5$. This score does not genuinely reflect the data quality as the scales used are inherently different. Hence, it is crucial to standardize the ratings onto a common scale to facilitate meaningful comparisons.

- Different Quality Thresholds. Even with ratings standardized to a common scale, we face the problem of varying quality thresholds distinguishing data that positively contribute to pretraining. For example, Rater A may deem ratings above 5 as high-quality, whereas Rater B may view ratings above 80% as high-quality. Suppose we standardize both ratings to a 0-1 scale, turning an 8 from Rater A into 0.8 and 85% from Rater B into 0.85. Despite this standardization, direct comparison of the two raters' scores remains impractical due to their differing threshold values for differentiating data quality.

C.2 Prompt for GPT-4o to compare data quality

Prompt for GPT-4o to compare data quality

Compare two text excerpts and choose the text which contain more informative signal for pretraining a large–language model.

An informative datapoint should be well–formatted, contain some usable knowledge of the world, and strictly NOT have any harmful, racist, sexist, etc. content.
Aspects that should NOT influence your judgement:
1. The length of the text
2. The order in which the texts are presented

Note that the texts are cut off, so you have to infer their contexts. The texts might have similar quality, but you should still make a relative judgement and choose the label of the preferred text.

[Option A]
... {text a} ...
[Option B]
... {text b} ...

Now you have to choose between either A or B. Respond only with a single word.

C.3 Reliability Analysis of GPT-4o

QuRating(Wettig et al., 2024) points out that GPT is more effective at comparing the relative quality between two data samples than performing absolute quality evaluation. To further assess the reliability of GPT-4o, we use QuRating (Educational Value) as a case study and conduct a win-rate evaluation involving human experts. Specifically, we compare the win rates assigned by human annotators and GPT-4o across different rating percentiles, with results presented in descending order of percentile rating in Table 5. The two sets of win rates exhibit a Pearson correlation of 0.99, indicating strong agreement and suggesting that GPT-4o does not introduce significant bias in the annotation process. This high consistency supports the reliability of using GPT-4o for quality assessment. Furthermore, GPT-4o is used only to estimate win rates between rater-selected samples and random subsets, rather than to assign absolute scores, which further reduces bias.

C.4 Polynomial spline interpolation for win-rate-percentile function

To obtain the aligned rating for each data point, we consider the win rate w_{ij} of rater i in the j-th rating interval as the rating of the midpoint of that interval, denoted by coordinates (p_j, w_{ij}). We then complete the rater's win-rate-percentile function using polynomial spline interpolation to derive a continuous and smooth win-rate-percentile function. The polynomial spline interpolation function $S(p)$ is defined as follows:

$$\begin{aligned} S(p) =& a_k(p-p_k)^n + b_k(p-p_k)^{n-1} + \cdots \\ &+ y_k(p-p_k)^2 + z_k(p-p_k) + d_k, \\ &\qquad p_k \le p < p_{k+1} \end{aligned} \tag{10}$$

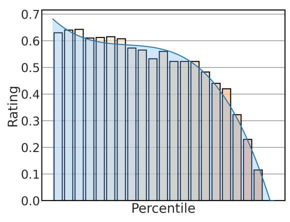
where p denotes the percentile, p_k and p_{k+1} are the boundaries of the k-th interval, n is the degree of the polynomial, and a_k, b_k, $\cdots$, y_k, z_k, and d_k are the coefficients determined through the spline interpolation process.

C.5 Ratings distribution illustration

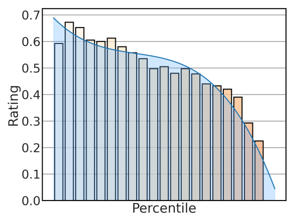
Figure 8 shows the win rates of samples in different percentile intervals and the fitted Rating-Percentile curves for 4 raters. The original ratings provided by the four raters exhibit significant differences, as illustrated in Figure 4 of QuRating(Wettig et al., 2024). The alignment introduces a random subset for comparison, which makes the ratings from different raters comparable, mapping the ratings into a similar range. This forms the foundation for the subsequent weighted integration of the raters, which explains the poor performance without alignment. However, even after alignment, there are still noticeable differences in the rating distributions of different raters. For instance, in Figure 8a and 8c, there are significant differences in the win rate of the first quartile and the middle section of the curve.

C.6 Formalization of Orthogonality.

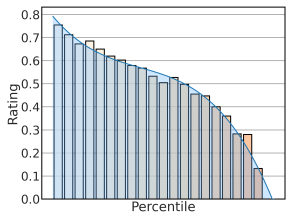
Empirical determination of the boundary conditions. For convenience and without losing rationality, we use the initial version of the integrated



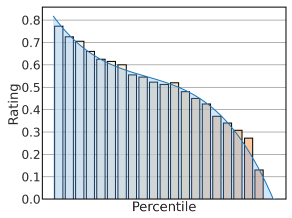
(a) Writing Style



(b) Required Expertise



(c) Facts and Trivia



(d) Educational Value

Figure 8: The win rates of samples in different percentile intervals and the fitted Rating-Percentile curves for 4 raters.

rating calculation formula (without PageRank optimization) to determine the boundary values of orthogonality. The integrated rating of data point i can be expressed as

$$I(i) = \sum_{j=1}^{n} \gamma_j o_j A_j(i)$$

where $o_j = \sum_{\substack{k=1 \\ k\neq j}}^{n} O(j,k)$ is a quantification of the overall orthogonality R_j with other raters. We ignore the impact of raters' reliability and set $\gamma_j = 1$. By substituting the definition of the overall orthogonality, we obtain

$$\begin{aligned} I(i) &= \sum_{j=1}^{n} \left(\sum_{\substack{k=1 \\ k\neq j}}^{n} O(j,k) \right) A_j(i) \\ &= \frac{1}{2} \sum_{j=1}^{n} \sum_{\substack{k=1 \\ k\neq j}}^{n} O(j,k) \left(A_j(i) + A_k(i) \right) \end{aligned} \tag{11}$$

The final expression, after rearrangement, can be seen as pairwise addition of raters, with their integrated ratings weighted by the orthogonality of the two raters. When two raters are perfectly correlated, it means their ratings are identical across all data points. In this scenario, the information provided by one rater is entirely redundant with respect to the other. Therefore, the orthogonality between these raters should be set to 0, indicating no additional information is gained by considering both ratings. Conversely, when two raters are

Percentile	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
GPT-4o	0.773	0.705	0.625	0.600	0.545	0.513	0.480	0.425	0.340	0.273
Human Expert	0.797	0.698	0.657	0.616	0.533	0.501	0.469	0.433	0.317	0.300

Table 5: Win-rate comparison between GPT-4o and human experts for QuRating (Educational Value) across different rating percentiles.

perfectly orthogonal, it signifies that their ratings are completely independent of each other. Each rater offers unique and complementary perspectives, leading to a comprehensive evaluation when combined. In such cases, the orthogonality should be set to 0.5, reflecting that each rater contributes equally distinct information to the overall rating.

Orthogonality function. The most intuitive approach to determine the orthogonality between two raters is to use the correlation coefficient between their rating distributions on a pretraining dataset. The orthogonality-correlation function needs to satisfy two key conditions:

- **Monotonicity:** The stronger the correlation, the lower the orthogonality.
- **Boundary Conditions:** Empirically, when two raters are completely uncorrelated, the orthogonality between them is 0.5; when they are fully correlated, the orthogonality is 0. And the empirical value has been described above.

In fact, we choose the orthogonality between two raters to be 0 when they are completely correlated, as we want to avoid duplication issues during rater integration. For instance, consider three raters, where Rater 1 and Rater 2 are completely correlated (correlation coefficient of 1) and are completely uncorrelated with Rater 3 (correlation coefficient of 0). Considering only one iteration, the integration result of the three raters is $R = O_{12}(R_1+R_2)+O_{13}(R_1+R_3)+O_{23}(R_2+R_3) = (O_l + O_h)R_1 + (O_l + O_h)R_2 + 2O_hR_3$, where $O_l = O_{12}$, $O_h = O_{13} = O_{23}$. Given that Rater 1 and Rater 2 are completely correlated, which means $R_1 = R_2$, we obtain $R = 2(O_l+O_h)R_1+2O_hR_3$. If $O_l \neq 0$, R_1 will be assigned an additional weight. However, one would naturally assume that R_1 and R_3 should carry the same weight when they are completely uncorrelated.

Considering the scenario where the correlation coefficients among all raters are zero, the calculation of orthogonality would completely degenerate to zero. Therefore, in such extreme cases, we treat all perfectly correlated raters as a single rater and do not proceed with orthogonality-related integration.

Based on the conditions, we explored three functional forms that satisfy the criteria: a linear function, a Gaussian function, and a symmetrically processed Gaussian function. The three functional forms explored for orthogonality are as follows:

(1) **Linear Function:**

$$O_L(i,j) = \frac{1}{2} \cdot (1 - |r(i,j)|) \tag{12}$$

(2) **Gaussian Function:**

$$O_G(i,j) = \exp\left(-\frac{r(i,j)^2}{2c^2}\right) - \frac{1}{2} \tag{13}$$

(3) **Symmetrically Processed Gaussian Function:**

$$O_{S.G}(i,j) = \left(\frac{3}{2} - |r(i,j)|\right) - \exp\left(-\frac{r(i,j)^2}{2c^2}\right) \tag{14}$$

where $r(i,j)$ represents the correlation measure between raters i and j. Specifically, it is quantified using the Pearson correlation coefficient between the score distributions of the two raters over a common dataset. In both Gaussian-based functions, the constant $c = \sqrt{\frac{1}{2\ln 2}}$ is determined by the boundary conditions. Figure 9 illustrates the 3 forms of orthogonality functions with respect to correlation coefficient.

It's worth noting that the experiments in Appendix E.3 demonstrate that the symmetrically processed gaussian function exhibits the best performance. Therefore, in the main experiments of this paper, we use the symmetrically processed gaussian function, i.e., Equation (14). Additionally, when integrating the ratings from two raters, their weights based on orthogonality are the same. Therefore, we only use the intrinsic reliability of the raters to weight their ratings.

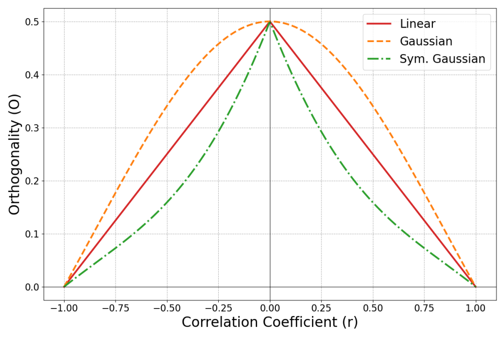


Figure 9: Trends of 3 forms of orthogonality functions with the change in the correlation coefficient.

Hyperparameter	Value	Hyperparameter	Value
Attention heads	16	Precision	bfloat16
Layers	24	Vocab size	32,000
Hidden size	2048	Window length	1024
Intermediate size	5504	Tied embedding	False
Position embedding	ROPE	Activation	SwiGLU

Table 6: The hyperparameters of model structure.

C.7 Orthogonality-Percentile curve

Figure 10 shows the orthogonality calculated based on data subsets from different quantiles. It is evident that the orthogonality calculated from data in different quantiles varies.

D Experimental Details

D.1 Model and training

We train a model with 1.3B parameters similar to the Llama architecture, the model structure is detailed in the Table 6. We train the language model from scratch and randomly initialize the model parameters. We set the batch size to 2048 and the learning rate to 5e-4, using a cosine learning schedule. To accelerate the training and inference processes, we use the bfloat16 format during both training and testing. The training is based on the Megatron framework and utilizes flash attention. The entire model is trained on 16 A100 GPUs for a total of 10,000 steps. For FIRE orthogonality calculation, we use the symmetrically processed gaussian function.

D.2 Integration Baselines

We compare our rating integration method with other baseline methods. Here we give more details about the baselines.

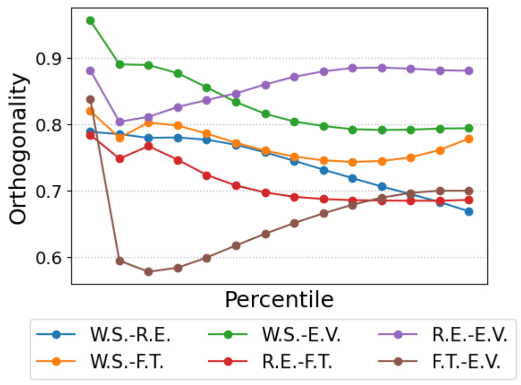


Figure 10: Trends of orthogonality functions with the change in percentile of different pairs of raters.

Prompts for Comprehensive Rater

Compare two text excerpts and choose the text which
1. has a more polished and beautiful writing style.
2. contains more facts and trivia. Prefer specific facts and obscure trivia over more common knowledge.
3. requires greater expertise and prerequisite knowledge to understand it.
4. has more educational value, e.g., it includes clear explanations, step–by–step reasoning, or questions and answers.
Aspects that should NOT influence your judgement:
1. Which language the text is written in
2. The length of the text
3. The order in which the texts are presented

Note that the texts are cut off, so you have to infer their contexts. The texts might have similar quality, but you should still make a relative judgement and choose the label of the preferred text.

[Option A]
{text1}
[Option B]
{text2}

Now you have to choose between either A or B. Respond only with a single word.

Comprehensive Rater Following QuRating (Wettig et al., 2024), we collect pairwise comparison data and train a reward model based on Sheared-Llama 1.3B (Xia et al., 2023). We merge all the evaluation criteria into a single prompt to guide GPT-4o comparison. Above is an illustration of the prompt.

Mix Criteria We utilize each rater to select the top 20B tokens, which are then merged and duplicates removed. Following this, we randomly pick out another 20B tokens from this merged set. As each sample only requires to excel in one rater's evaluation to be considered for selection, this approach emphasizes the dimension in which the sample performs best amongst all dimensions.

Max Criteria Once the scores are aligned, we directly select the dimension with the highest rating to represent the integration result. This method is analogous to performing max-pooling across all dimensions, straightforwardly highlighting the dimension within the sample that has the highest rating.

Average For multi-rater integration, the most straightforward approach is to assume that each rater contributes equally to the overall quality of the data. Accordingly, we normalize the ratings provided by each rater for the samples. Following normalization, the average of these ratings is computed to obtain the final integrated rating.

D.3 Analysis of Computational Cost

This section analyzes the computational cost (measured in FLOPs) for various data rating and integration methods. The focus is on the cost introduced by applying raters to the pre-training data, excluding the main model's training cost. FIRE achieves more effective integration without introducing significant extra cost beyond the base raters. The FLOPs comparison is summarized in Table 4.

The analysis includes:

1. Rater training (if applicable),
2. Rater inference over the entire dataset,
3. Additional computation for FIRE's win-rate-based integration.

QuRating (Single Rater). Each QuRating model is a 1.3B-parameter transformer. It is first fine-tuned on 500K examples (512 tokens each). The total FLOPs for training this rater is approximately:

$$\text{FLOPs}_{\text{train}} \approx 6 \cdot N \cdot d^2 \cdot L \cdot T \approx 4.2 \times 10^{19}$$

Rater inference over the entire 627B-token dataset consumes:

$$\text{FLOPs}_{\text{infer}} \approx 6 \cdot 627 \times 10^9 \cdot d^2 \cdot L \approx 8.13 \times 10^{20}$$

$$\text{FLOPs}_{\text{total}} \approx 8.17 \times 10^{20}$$

QuRating (Mix of Criteria). Integrating four QuRating raters (each covering a distinct aspect) requires four forward passes across the full dataset. Since retraining is not needed, total cost is:

$$\text{FLOPs}_{\text{mix}} \approx 4 \times \text{FLOPs}_{\text{infer}} \approx 3.26 \times 10^{21}$$

FIRE. FIRE uses the same four QuRating raters as input, and hence shares the same inference cost. In addition, it estimates the win-rate-percentile mapping using only 20K pairwise comparisons per rater. The additional cost of this step is negligible compared to inference over 627B tokens.

D.4 Integration of Other Raters

FIRE has extremely high scalability: our approach is not directly related to the specific attributes of the raters. Therefore, as long as the raters can provide ratings for the samples and are not completely related to each other, they can be integrated using the FIRE method. We also attempt to integrate another raters: QuRating(Required Expertise), QuRating(Facts and Trivia), DSIR-Book, and DSIR-Wiki. Tabel 7 shows that the average performance of FIRE, after integrating these four raters, is better than each individual rater and the Random baseline, indicating that our integration method is still effective on other raters.

Method	ARC-E	ARC-C	SciQ	LogiQA	BoolQ	HellaSw.	PIQA	W.G.	AVG.
Random	48.2	22.3	84.5	19.7	60.8	32.1	63.5	49.2	47.5
DSIR with Book	36.2	19.5	73.4	21.4	61.8	29.5	62.5	53.6	44.7
DSIR with Wiki	37.2	18.0	76.4	23.0	58.0	27.9	57.3	51.1	43.6
QuRating (R.E.)	50.6	23.2	83.9	22.6	61.4	30.2	59.8	49.8	47.7
QuRating (F.T.)	54.1	23.0	83.5	22.0	60.9	30.4	59.5	51.7	48.1
FIRE	**57.1**	**25.9**	**84.9**	**20.8**	**61.5**	**31.5**	**60.1**	**50.3**	**49.0**

Table 7: Performance comparison by applying FIRE to integrate four raters: QuRating (Required Expertise), QuRating (Facts and Trivia), DSIR-Book, and DSIR-Wiki.

D.5 Prompt for Multi-dimension Analysis

Prompts for GPT-4o evaluation

You are a data annotation expert. You should judge that {condition}

Aspects that should NOT influence your judgement:
1. Which language the text is written in
2. The length of the text
3. The order in which the texts are presented

Note that the texts are cut off, so you have to infer their contexts.
Here is the text:
[TEXT BEGIN]
{text}
[TEXT END]

Please follow the question order to respond. For answer, only respond yes or no.
Return the results for each question in the following json format:
[{
"quesion": "Is this text has a polished and beautiful writing style ?",
"reason": "Fill in the reason for the judgment here",
"answer": "yes/no"
},
...]

We instruct the GPT-4o to evaluate the data selected through the raters, and here we show the prompts. For dimension, we consider four individual dimensions, the same as QuRating Wettig et al. (2024); and a comprehensive dimension:

- Does this text have a polished and beautiful writing style?
- Does this text contain many facts and trivia? Prefer specific facts and obscure trivia over more common knowledge.
- Does this text have much educational value? E.g., it includes clear explanations, step-by-step reasoning, or questions and answers.
- Does this text require a lot of expertise and prerequisite knowledge to understand it?
- Does this text contain an informative signal for pretraining a large-language model? An informative data point should be well-formatted, contain some usable knowledge of the world, and strictly NOT have any harmful, racist, sexist, etc. content.

E Further Analysis of Experiments

E.1 Results of different combinations

We present the results of different rater combinations in Table 8. The average scores of FIRE (W.S.+R.E.+F.T.) and FIRE (W.S.+R.E.+E.V.) both surpass the results obtained from combining any two of their individual raters. Furthermore, FIRE (W.S.+R.E.+F.T.+E.V.) achieves even better performance.

E.2 Prompt merge effect

We investigate whether combining multiple rules within a single prompt can effectively meet the evaluation standards of each rule. We randomly selected 3,000 data points and guided GPT-4's evaluation using a prompt that integrates multiple rules, alongside conducting individual rule-guided evaluations and separate human annotator evaluations

Method	**ARC-E**	**ARC-C**	**SciQ**	**LogiQA**	**BoolQ**	**HellaSw.**	**PIQA**	**W.G.**	**AVG.**
				Two Raters Integration					
FIRE (W.S.+R.E.)	54.5	24.6	85.6	19.7	61.3	31.3	60.4	51.8	48.7
FIRE (W.S.+F.T.)	56.1	25.0	81.5	22.7	55.1	31.7	61.2	49.7	47.9
FIRE (R.E.+F.T.)	56.9	26.1	84.7	20.7	62.0	30.5	59.4	50.0	48.8
FIRE (W.S.+E.V.)	56.7	24.2	82.2	21.0	60.3	**33.0**	61.0	51.0	48.7
FIRE (R.E.+E.V.)	53.0	24.7	84.1	22.4	61.2	31.3	58.7	49.7	48.1
				Three Raters Integration					
FIRE (W.S.+R.E.+F.T.)	59.0	25.4	85.6	**25.5**	57.5	31.9	61.8	51.1	49.7
FIRE (W.S.+R.E.+E.V.)	57.8	25.9	84.5	20.6	62.0	32.7	60.5	51.3	49.4
				Four Raters Integration					
FIRE (W.S.+R.E.+F.T.+E.V.)	59.1	26.4	86.0	21.0	61.8	32.9	59.7	52.8	50.0

Table 8: Downstream tasks results for different rater combinations. We report accuracy for each task, and the best performances are marked in bold. Abbreviations: HellaSw. = HellaSwag, W.G. = WinoGrande, AVG. = Average, W.S. = Writing Style, R.E. = Required Expertise, F.T. = Facts and Trivia, E.V. = Educational Value

	W.S.	F.T.	E.V.	R.E.
Sing.(Human)	53.3	69.7	85.1	92.2
Sing.(GPT4)	52.1	69.7	85.5	91.8
Comp.(GPT4)	57.8	80.5	86.9	50.3
$\rho_{Human-Sin.(GPT4)}$	0.81	0.85	0.86	0.77
$\rho_{Human-Com.(GPT4)}$	0.72	0.64	0.72	0.32

Table 9: Results of GPT-4 and human annotation for 3,000 samples. Sing. = Single, Comp. = Comprehensive.

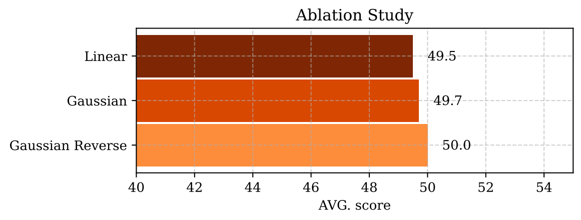


Figure 11: Ablation experiments evaluating the impact of different orthogonality functions on model performance.

for each criterion. To understand how GPT-4's holistic evaluation aligns with each individual dimension, we employed CoT (Wei et al., 2022) approach: the model first evaluates each rule separately, and then provides an overall evaluation. Each rule is assessed with a binary yes/no question, and after six evaluations, we average the results to obtain the final score (for human evaluations, this entails averaging the scores given by six annotators).

Table 9 displays the percentage of data with a relatively high degree in each dimension, along with the correlation coefficients between GPT-4's and human evaluations. From the proportions of high-quality data in each dimension, it is evident that GPT-4's scores approximate human scores more closely when evaluated individually, with differences within a margin of 0.2 at most. However, GPT-4's scores exhibit greater variability when multiple rules are integrated. In terms of correlation coefficients, there is a strong correlation between GPT-4's individual scoring and human scoring, but this correlation significantly diminishes when it comes to comprehensive scoring. Particularly, in the *Require Expertise* aspect, the correlation is only 0.32. This suggests that GPT-4's current capability to adhere to all rules in a single prompt is still inadequate.

E.3 Orthogonality Function

There are various methods to calculate orthogonality, and we aim to identify the most effective one. We compare three functions for orthogonality calculation: *linear*, *gaussian*, and *sym. gaussian* (the symmetrically processed gaussian function). As illustrated in Figure 11, the three functions achieve comparable results, all surpassing the configuration without orthogonality, which demonstrates the effectiveness of incorporating orthogonality. Besides, the sym. gaussian function outperforms the other two. The gaussian function smooths the correlation coefficient around zero, while the sym. gaussian function amplifies changes in orthogonality near zero. The linear function, however, strikes a balance between these two. Our experimental results confirm that enhancing the rate of change around zero is more efficient, emphasizing the role of highly orthogonal raters, and intensifying the penalties for raters with high correlation.

E.4 Effect of Sample/top-K

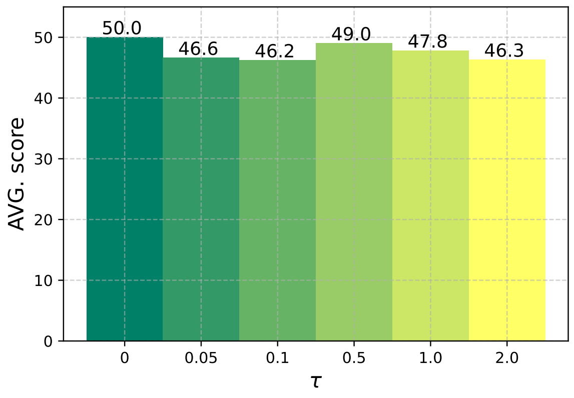


Figure 12: The impact of the sample temperature coefficient on model performance. τ is the temperature coefficient, and when $\tau = 0$, it refers to top-K selection.

Qurating (Wettig et al., 2024) suggest that sampling is more effective than directly selecting the top-K data. In this experiment, we integrate four different raters to rigorously investigate the impact of sampling on the model's final performance. Specifically, we calculate the sampling probability for all rated samples using the following softmax formula:

$$P(x) = \frac{exp(I(x)/\tau)}{\sum exp(I(x)/\tau)} \tag{15}$$

where τ is the temperature parameter. We train a model of the same size as in the previous experiments. From the results presented in Figure 12, several key observations can be made:

(1) Our findings reveal that direct top-K selection outperforms sampling, further affirming the efficacy of our integration method. Additionally, Wettig et al. (2024) posits that sampling enhances data diversity, which is beneficial for model learning. Our results indicate that the top-K scores are higher than the sampling scores, demonstrating that the top-ranked data according to our integration rating exhibit a broader distribution rather than being concentrated in a single domain.

(2) Contrary to the optimal value of 2 suggested by Wettig et al. (2024), our analysis indicates that a relatively smaller τ value yields optimal results. A smaller τ accentuates the impact of ratings, suggesting that our method effectively selects higher-quality data. Furthermore, our multi-dimensional approach also accounts for the diversity of data types, ensuring a more comprehensive evaluation.

F Selected Data Cases

We show the document cases rated by the single raters and FIRE in the Table 10. For each method, we show the best/middle/worst sample.

Method	Best	Middle	Worst
Writing Style	... is the very thing that makes each person inimitable, the thing that allows us finally to see and celebrate one another's distinct natures. Once it is understood that this expanse must always exist, each person is free to become whatever it is they will become, unburdened by the need to shape themselves to fit their partner. And this individuation need not be a growing apart. For if each partner can remember the beauty and necessity of the expanse, then they can come to appreciate fully the ...	... him some of that history," she said. Near the beginning of the session panelist Don Lemon of CNN played video of his story on what he called "a picture of racial unity," Sherrod's reunion with the elderly white farmers whose farm she helped save, after first not making a full effort at a non-profit where she worked 24 years ago. The author of legislation that would require natural-gas companies to disclose hydraulic-fracturing fluids says she feels betrayed by industry groups that have spoken ...	... - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D Stat Scoring Average (Actual) - 2006 Stat Scoring Average (Actual) - 2006 Y-T-D St ...
Required Expertise	... induced climate change have used instrumental records to study how quickly climate is warming across different parts of the world. Our study uses a collection of natural archives that preserve information about past temperatures over a much longer period, spanning the last 500 years, to ask the question: "When did the sustained warming trends that we've seen in the 20th and 21st Centuries first begin?" ...	... (by R.M. Butler?) in 'Folder 20' seen by Nick Sheaff, 1970s; Liam Swords, Achonry and its churches (Strasbourg: Éditions du Signe, 2007), 78(illus.). Nature: Additions, for Lady Fitzgerald Arnott. contractor: Michael O'Brien, Dun Laoghaire. Refs: IB 59, 27 Oct 1917, 551; MS letter in IAA (Acc. 88/118) from Rev.P. Kilkenny to Butler, 6 Jun 1919, states that he cannot resist 'revolutionary, ...	... more. Awesome. : 1138 This is one awesome article.Thanks Again. Really Cool. : 1136 Appreciate you sharing, great article. : 1135 I cannot thank you enough for the blog post.Much thanks again. Great. : 1134 I think this is a real great post.Much thanks again. Cool.: 1133 A big thank you for your blog article.Really looking forward to read more. Great. : 1132 Great, thanks for sharing this blog post.Much thanks again. ...
Facts and Trivia	... Native American to carry the United States flag at the opening ceremony of the Olympic Games: Clarence "Taffy" Abel (Chippewa). 1926 First Native American in the NHL New York Rangers November 16, 1926: Clarence "Taffy" Abel (Chippewa). First Native American woman to hold state office in Oklahoma: Jessie Elizabeth Randolph Moore (Chickasaw). First national reform group with only Native American membership: National Congress of American Indians (NCAI) by Zitkala-Sa ...	... I probably only understand two-thirds of what's going on, honestly, but it's still damn good stuff. (Audiobook) Being Mortal by Atul Gawande: Heard raves about this non-fiction about elder and end-of-life care from many a Rioter, and it's ringing all my Books That Make Me Want To Change the World buttons. (audiobook) Half-Resurrection Blues by Daniel José Older: Loved Older's Salsa Nocturna, so I speedily picked up this novel about a half-alive, half-dead sort-of-secret-agent who works for ...	... T'AIME JE T'AIME ...
Educational Value	... Learn what "big history" is and how scholars apply this approach to the story of humanity. Gain new understanding of the complete sweep of human history, across all civilizations and around the world. Use the lens of history to find out what makes us human, why the world exists as it does today, and where we might be going in the future. See how the environment, population growth, social complexity, and more have driven the rise and fall of civilizations over ...	... a similar set of models as Fig. 2, this time displaying the C-star fractions for models with varying f_{CE} between 0.008 and 0.4 at the base of the convective envelope. In this instance, there is a far more straightforward interpretation, with an increase in f_{CE} producing an increased C-star fraction, in almost all cases. Furthermore, there is more readily acceptable agreement with the observed C-star fractions than was the case for the ...	... 05:17:33 https://cse.google.com.bo/url?sa=t url=https://toppornsites.mobi 2023-01-27 05:17:33 https://cse.google.com.br/url?sa=t url=https://toppornsites.mobi 2023-01-27 05:17:33 https://cse.google.com.by/url?sa=t url=https://toppornsites.mobi 2023-01-27 05:17:33 https://cse.google.com.bz/url?sa=t url=https://toppornsites.mobi ...
FIRE	In mathematics, the differential geometry of surfaces deals with the differential geometry of smooth surfaces with various additional structures, most often, a Riemannian metric. Surfaces have been extensively studied from various perspectives: extrinsically, relating to their embedding in Euclidean space and intrinsically, reflecting their properties determined solely by the distance within the surface as measured along curves on the surface. One of the fundamental concepts investigated is ...	... Everett, Washington: Charlotte Murray, 2010. Second Edition of 10. 7.5 x 5.25"; 56 pages. Images captured using three digital cameras, a Nikon Coolpix 5700, a Nikon D70, and a Nikon D80. Printed with Epson Photo Stylus R2880 printer with UltraChrome K3 pigment inks on Epson Premium Presentation Paper Matte. Papyrus font. Coil binding with green see through cover and lightweight cardboard back. Colophon: "The Dead Tree Scrolls first edition was created in 2005. This second edition was ...	... HBP - MILLER; MORALES. SF - WILKERSON(2). SB - EPPS(9); HORAN(3); PINDER(4). CS - EPPS(3). Clemson IP H R ER BB SO AB BF Justin Sarratt...... 6.1 6 3 3 1 8 24 26 Alex Frederick...... 0.1 0 0 0 0 0 1 1 Joseph Moorefield... 0.1 0 0 0 0 1 1 1 Matt Campbell....... 2.0 0 0 0 3 3 6 9 Virginia Tech IP H R ER BB SO AB BF Marc Zecchino....... 7.1 7 5 4 1 7 27 29 Jake Atwell......... 0.2 3 3 3 0 0 5 5 Sean McDermott ...

Table 10: Data cases are randomly selected from the top, middle, and bottom 0.01% of training samples, representing the best, middle, and worst cases for each method.