AIM: Manifold-based Data Filtering for Representation Finetuning

Qing Li[1], **Qibin Zheng**[2*], **Yi Liu**[2], **Xingchun Diao**[1]

[1]School of Computer Science, Shanghai Jiao Tong University
[2]Advanced Institute of Big Data,Beijing
qing15@sjtu.edu.cn, zhengqb@aibd.ac.cn, liuyi@aibd.ac.cn, diaoxch640222@163.com

Abstract

Representation Finetuning (ReFT) has recently emerged as an efficient paradigm for adapting pretrained language models by editing hidden representations rather than model weights. However, our preliminary experiments reveal that ReFT is notably more sensitive to training data quality compared to traditional parameter-efficient finetuning methods, particularly to samples with incorrect labels, which can severely degrade performance. Inspired by prior work demonstrating that the hidden representations of generalizable neural networks exhibit low-dimensional manifold structures, we hypothesize that effective generalization in ReFT requires geometrically structured transformations between pre- and post-intervention representations. This implies that the intervention vectors representing these transformations should form a low-dimensional manifold, rendering the inconsistent transformations induced by label noise as detectable geometric outliers. To leverage this insight, we introduce **A**ligning **I**nterventions on a learned **M**anifold (AIM), a representation-based data filtering method for ReFT, which identifies high-quality training samples by measuring the geometric consistency of their intervention vectors with respect to a robust reference manifold derived via principal component analysis on trusted data. Extensive experiments on both commonsense and arithmetic reasoning tasks confirm the effectiveness of AIM, showing consistent improvements over strong data selection baselines across multiple model scales.

Introduction

Representation Finetuning (ReFT) is a recently proposed paradigm for adapting pretrained language models by directly intervening in hidden representations while keeping model weights frozen (Wu et al. 2024). Compared with conventional parameter-efficient finetuning (PEFT) methods, ReFT achieves stronger parameter efficiency and interpretability by directly modeling the structure of intermediate representations. The growing adoption of ReFT across various tasks highlights its potential as a promising alternative for efficient model adaptation, which also motivates further exploration of its limitations and optimization opportunities.

While ReFT demonstrates remarkable parameter efficiency, our analysis reveals that it is intrinsically more sensitive to training data quality than conventional PEFTs such as LoRA. This heightened sensitivity stems from the fundamental design of ReFT: instead of adjusting model weights, it directly optimizes intervention vectors in the representation space, where even minor deviations in supervision signals can lead to large shifts in the resulting representations (Turner et al. 2023; Geiger et al. 2024). We empirically assess the sensitivity of ReFT under varying proportions and types of training noise, including answer-level label noise, domain-irrelevant samples, and token-level perturbations. The result in Figure 1 (a) shows that the performance of ReFT degrades most rapidly when incorrect-answer samples are introduced, whereas its performance decline under token corruption or cross-domain data remains comparable to that of LoRA. These findings highlight that the performance of ReFT heavily depends on high-quality data and is particularly vulnerable to label noise, motivating the need for careful filtering of its training data.

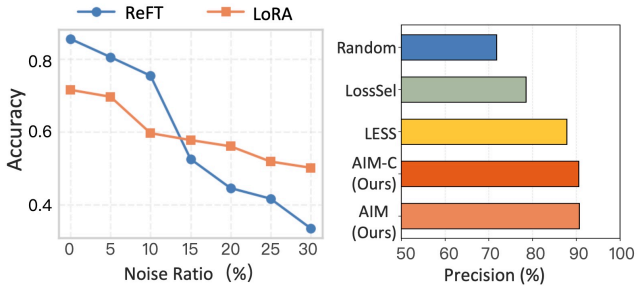

Figure 1: (a) Average accuracy of ReFT and LoRA on commonsense reasoning tasks under varying noise ratios in the training data. Experiments are conducted on LLaMA-3 8B. As the noise ratio increases, ReFT shows a more significant performance drop, indicating its higher sensitivity to data quality. (b) Comparison of data filtering methods on LLaMA-3 8B. Our proposed methods, AIM and AIM-C, achieve the best overall performance across tasks.

Various strategies have been developed to identify and exclude noisy training samples. One common approach identifies noisy samples by their persistently high loss, but often confuses them with genuinely difficult examples (Toneva et al. 2018; Han et al. 2018). Other methods improve fil-

*Corresponding author.

tering accuracy by assessing prediction consistency across either stochastic passes or training epochs (Li, Socher, and Hoi 2020; Swayamdipta et al. 2020a; Yang et al. 2024), or through adjudication by external large language models (Zhou et al. 2023; Li et al. 2024; He et al. 2025). While the former incurs considerable computational overhead, the latter introduces reliance on external models, limiting its applicability in lightweight finetuning scenarios. Motivated by these limitations, we propose a data filtering method that is both effective in distinguishing hard samples from noisy ones and computationally efficient.

We are inspired by the belief that better generalization in neural networks is closely tied to the geometric simplicity of their learned representations. Prior studies (Verma et al. 2019) have shown that when the hidden states of semantically related samples are compressed onto low-dimensional manifolds, models tend to generalize better. These findings suggest that structured and coherent hidden representations are often associated with consistent data characteristics and stable model behavior. Building on this insight, we hypothesize that the geometric structure of hidden representations can indicate data quality. In the context of ReFT, we expect that high-quality samples induce intervention vectors that align well with a shared low-dimensional subspace. In contrast, noisy samples tend to produce irregular or inconsistent directions that deviate from this common structure.

Based on this hypothesis, we propose **A**ligning **I**nterventions on a learned **M**anifold (**AIM**), a lightweight and training-decoupled data filtering method tailored for ReFT. AIM consists of two stages. In the first stage, it performs Principal Component Analysis (PCA) on the intervention vectors derived from a verified clean subset of training data, forming a low-dimensional subspace that captures their dominant directions. In the second stage, each candidate sample is assessed by projecting its intervention vector onto the orthogonal complement of this subspace. We use the projection magnitude as the filtering score. A larger value suggests that the intervention deviates more from the typical pattern found in clean data, indicating a higher risk of label noise.

We also introduce a scalable variant of our method, AIM-Classifier (AIM-C), designed to improve efficiency when applied to large-scale training datasets. Instead of computing AIM scores for every sample, AIM-C selects a small representative subset and applies the full AIM analysis to it. Based on the resulting scores, samples are labeled as positive or negative and used to train a lightweight text classifier. The trained classifier acts as a substitute for AIM scoring, allowing efficient data selection at scale.

Experimental results on both commonsense and arithmetic reasoning benchmarks (Hu et al. 2023) demonstrate that our method generally outperforms a range of strong data selection baselines across multiple model scales. As shown in Figure 1 (b), our method achieves the best overall performance among all compared baselines. Compared to the existing approach, our full method achieves up to 6.38% improvement on reasoning-intensive tasks and shows stable gains even under synthetic noise.We further ablate segment-level aggregation strategies and weighting parameters, verifying that emphasizing high-reliability representations and moderate weight scaling leads to more effective selection. The proposed method generalizes well across LLaMA-3.2-3B, LLaMA-2-7B, and LLaMA-3-8B, providing a scalable solution for robust training with noisy data.

Data Filtering with AIM

Preliminary

Representation Finetuning Representation fine-tuning (ReFT) adapts pre-trained language models by learning additive modifications on hidden representations, without updating model parameters. Given a hidden state $\mathbf{h} \in \mathbb{R}^d$, ReFT applies the following transformation:

$$\Phi_{\text{LoReFT}}(\mathbf{h}) = \mathbf{h} + \mathbf{R}^\top (\mathbf{W}\mathbf{h} + \mathbf{b} - \mathbf{R}\mathbf{h}) \tag{1}$$

where $\mathbf{W}, \mathbf{R} \in \mathbb{R}^{k \times d}$ are trainable low-rank matrices, and $\mathbf{b} \in \mathbb{R}^k$ is a bias term. This update introduces an intervention vector $\Delta\mathbf{h}$, which captures task-specific adjustments in representation space.

In ReFT, these modules are inserted at selected layers, and only $(\mathbf{W}, \mathbf{R}, \mathbf{b})$ are trained. The base model remains frozen. The resulting intervention vectors serve as the foundation for our later analysis of data quality and geometric consistency.

Geometry of Intervention Vectors and Data Quality Neural networks that generalize well often learn compact and low-dimensional representations, consistent with the inductive bias of modern architectures (Arpit et al. 2017a; Bietti and Mairal 2019). Prior studies show that reliable training examples typically produce hidden states that lie on coherent, low-rank manifolds (Verma et al. 2019; Papyan, Han, and Donoho 2020), suggesting geometric structure as a proxy for data quality.

Let $h(x) \in \mathbb{R}^d$ be the hidden state of input x, and let $\mathcal{D}$ be a set of trusted samples. Their representations are expected to span a low-dimensional subspace:

$$\dim\left(\text{span}\{h(x_i)\}_{x_i \in \mathcal{D}}\right) \ll d. \tag{2}$$

We extend this principle to intervention vectors, which reflect changes in hidden states after applying supervision. When the supervision is accurate, these shifts are aligned and form a concentrated low-dimensional structure. In contrast, noisy supervision causes scattered, inconsistent changes. This motivates our hypothesis: the geometric consistency of intervention vectors reflects supervision quality.

Let $\{v_i\}_{i=1}^n \subset \mathbb{R}^d$ be intervention vectors from high-quality data. The empirical covariance matrix is:

$$S = \frac{1}{n} \sum_{i=1}^{n} (v_i - \bar{v})(v_i - \bar{v})^\top, \tag{3}$$

where $\bar{v}$ is the mean vector. If these vectors are geometrically aligned, the eigenvalue mass of S concentrates in a few directions:

$$\frac{\sum_{j=1}^{r} \lambda_j}{\sum_{j=1}^{d} \lambda_j} \approx 1, \quad \text{for } r \ll d. \tag{4}$$

We use this low-rank property to evaluate the alignment of new samples with the reference structure, yielding a geometry-based measure of data quality.

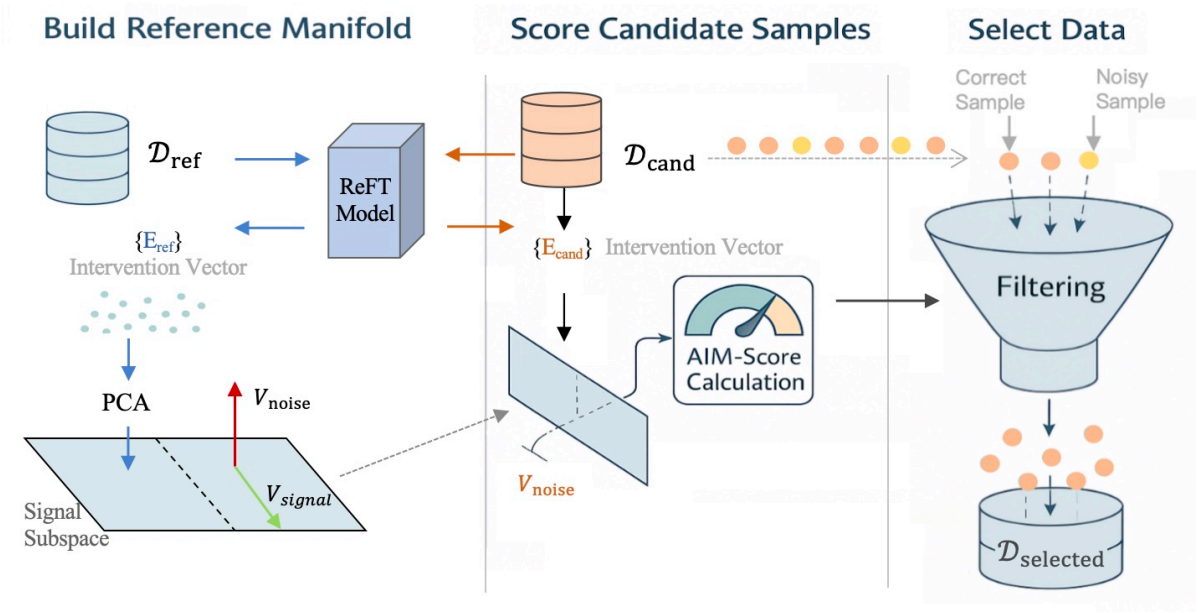

Figure 2: Overview of the proposed data selection framework. The method first computes intervention vectors from a high-quality reference dataset and constructs a reference manifold via PCA. Each candidate sample is then passed through ReFT to obtain segment-level intervention vectors, which are projected onto the reference manifold to compute AIM-Scores. Samples are scored based on their geometric alignment across early, middle, and late segments. The final AIM-Score aggregates segment-level scores with reliability weights, enabling unsupervised filtering of noisy samples. .

Intervention Consistency as a Proxy for Data Quality

We propose a geometry-based scoring method that evaluates training sample quality by measuring the alignment between their intervention vectors and a reference subspace. This unsupervised score reflects the assumption that high-quality samples induce structured, low-dimensional interventions, while noisy data exhibit larger deviations. The score relies solely on ReFT-induced internal representations and requires no task labels or predictions.

To capture structural variations across layers while maintaining efficiency, we adopt a segment-based strategy. Instead of modeling each layer separately or all layers jointly, we divide the model into early, middle, and late segments and compute intervention vectors within each segment. This design is motivated by prior findings (Chen et al. 2024; Gurnee and Tegmark 2024) showing layer specialization in Transformer models. Layers are assigned to segments proportionally, ensuring applicability across models with different depths.

Let $\mathcal{D}$ denote the full training dataset. We begin by selecting a small, high-quality subset $\mathcal{D}_{\text{ref}} \subset \mathcal{D}$, whose samples are manually verified to have correct outputs. A ReFT module is trained on $\mathcal{D}_{\text{ref}}$ to capture robust intervention patterns, from which we extract reference structures to guide scoring.

For each reference sample $x \in \mathcal{D}_{\text{ref}}$, we compute its intervention representations within each segment of the model. Specifically, let L be the total number of model layers, and divide them into three proportional segments: early ($[1, \lfloor 0.3L \rfloor]$), middle ($[\lfloor 0.3L \rfloor + 1, \lfloor 0.7L \rfloor]$), and late ($[\lfloor 0.7L \rfloor + 1, L]$). For each segment $s \in \{\text{early}, \text{middle}, \text{late}\}$, we collect all local intervention vectors $\{e^x_{l,p}\}$ computed as:

$$e^x_{l,p} = W_l h^x_{l,p} + b_l - R_l h^x_{l,p}, \tag{5}$$

where $h^x_{l,p} \in \mathbb{R}^d$ is the hidden state at layer l and token position p, and $W_l, R_l \in \mathbb{R}^{r \times d}, b_l \in \mathbb{R}^r$ are the trained ReFT parameters. The intervention vectors from all relevant layers and positions within segment s are concatenated to form a segment-level intervention vector $E^x_s \in \mathbb{R}^{d_s}$.

For each segment $s \in \{\text{early}, \text{middle}, \text{late}\}$, we aggregate the intervention vectors of all reference samples to form a segment-specific matrix:

$$\text{Pool}^{(s)}_{\text{ref}} = [E^{x_1}_s,\ E^{x_2}_s,\ \ldots,\ E^{x_n}_s] \in \mathbb{R}^{d_s \times n}, \tag{6}$$

where d_s denotes the dimensionality of the intervention vector in segment s, and $n = |\mathcal{D}_{\text{ref}}|$. We then perform PCA on $\text{Pool}^{(s)}_{\text{ref}}$ to obtain the leading subspace $V^{(s)}_{\text{signal}} \in \mathbb{R}^{d_s \times k_s}$, retaining the top-$k_s$ components such that the cumulative explained variance reaches a threshold ρ:

$$\frac{\sum_{j=1}^{k_s} \lambda^{(s)}_j}{\sum_{j=1}^{d_s} \lambda^{(s)}_j} \geq \rho. \tag{7}$$

The residual subspace $V^{(s)}_{\text{noise}}$ is defined as the orthogonal complement of $V^{(s)}_{\text{signal}}$, capturing directions not aligned with the dominant structure in segment s.

Next, for each candidate sample $x' \in \mathcal{D}_{\text{cand}}$ in candidate dataset, we use the ReFT module trained on $\mathcal{D}_{\text{ref}}$ to extract its segment-specific intervention vectors $\{E_s^{x'}\}_s$. Each vector $E_s^{x'}$ is projected onto the residual subspace $V_{\text{noise}}^{(s)}$ to measure deviation from the corresponding reference structure:

$$\text{AIM-Score}_s(x') = \frac{\left\| \text{Proj}_{V_{\text{noise}}^{(s)}}(E_s^{x'}) \right\|_2}{\|E_s^{x'}\|_2}. \tag{8}$$

To account for the varying discriminative utility across different architectural stages, we employ an adaptive weighting mechanism that scales segment-level AIM scores based on the representational complexity of the intervention subspace. For each segment s, we first determine the average projection error of the reference intervention vectors onto the corresponding residual subspace

$$\delta_s = \frac{1}{|\mathcal{D}_{\text{ref}}|} \sum_{x \in \mathcal{D}_{\text{ref}}} \left\| \text{Proj}_{V_{\text{noise}}^{(s)}}(E_s^x) \right\|_2 \tag{9}$$

where δ_s quantifies the degree of task-specific variation and informational richness within the reference set. Segments characterized by substantial projection errors are prioritized because they typically encapsulate the sophisticated reasoning structures and nuanced logical patterns essential for distinguishing high-quality samples in complex mathematical domains.

We subsequently transform these projection errors into normalized weights to emphasize the segments with the greatest discriminative potential

$$w_s = \frac{\delta_s}{\sum_{s'} \delta_{s'}} \tag{10}$$

which allocates significant importance to the segments exhibiting high representational diversity.

Finally, the overall AIM-Score for a candidate sample x' is computed as a weighted combination of its segment-level scores:

$$\text{AIM-Score}(x') = \sum_s w_s \cdot \text{AIM-Score}_s(x'). \tag{11}$$

The AIM-Score quantifies how well a candidate sample aligns with the geometric structures derived from high-quality data. A lower score indicates that the sample's intervention vectors are consistent with the reference subspaces, suggesting high data quality. A higher score reflects greater deviation, which may indicate label errors, distribution mismatch, or limited training utility.

A Scalable Variant: AIM-Classifier

To support large-scale data filtering, we introduce a scalable variant of our method, termed *AIM-Classifier*. Instead of computing AIM scores for every training sample, this variant trains a classification model on a small scored subset to approximate the selection criteria.

We first sample a subset from the full training dataset and compute AIM scores for all samples in this subset, following the procedure described earlier. Based on score ranking, we assign binary labels: samples with the lowest scores are labeled as positive, and those with the highest scores are labeled as negative. In our implementation, we use the lowest and highest 30 percent of the scores to construct the labeled set.We then fine-tune a lightweight text classifier based on DistilBERT using this labeled subset. The classifier learns to predict whether a given input sample satisfies the quality criteria defined by the AIM score. After training, the classifier is used to predict the quality of the remaining samples.

This method avoids computing explicit AIM scores for all samples, significantly reducing computational overhead. Although it may introduce approximation error compared to full scoring, it preserves the key geometric signal and enables efficient filtering for large-scale datasets.

Algorithm 1: Data Filtering with AIM

Require: Full dataset $\mathcal{D}$, trusted subset $\mathcal{D}_{\text{ref}}$, frozen base model f_θ
Ensure: Filtered dataset $\mathcal{D}_{\text{selected}}$

Step 1: Build reference manifolds
for each segment $s \in \{\text{early}, \text{middle}, \text{late}\}$ **do**
  Compute segment-level intervention vectors $\{E_s^x\}$ for all $x \in \mathcal{D}_{\text{ref}}$
  Apply PCA on $\{E_s^x\}$ to obtain subspace $V_{\text{signal}}^{(s)}$
  Compute orthogonal complement $V_{\text{noise}}^{(s)}$
  Compute segment weight w_s based on reference compactness δ_s
end for
Step 2: Score candidate samples
for each $x \in \mathcal{D}_{\text{cand}}$ **do**
  for each segment $s \in \{\text{early}, \text{middle}, \text{late}\}$ **do**
    Compute segment-level intervention vector E_s^x
    Compute normalized residual norm r_s^x
  end for
  Compute AIM-Score$(x) = \sum_s w_s \cdot r_s^x$
end for
Step 3: Select data
Rank all $x \in \mathcal{D} \setminus \mathcal{D}_{\text{ref}}$ by AIM-Score(x) in ascending order
Select top-k samples to form $\mathcal{D}_{\text{selected}}$
return $\mathcal{D}_{\text{selected}}$

The Overall Framework

We summarize the complete data filtering process in Algorithm 1. Starting from a pretrained encoder and a trusted dataset with high-quality labels, we first compute intervention vectors to construct a reference manifold that captures the geometric structure of reliable samples. Each candidate sample is then evaluated by measuring its deviation from this manifold. Samples with the lowest scores are selected for downstream fine-tuning or task-specific training. For large-scale settings, our method also supports a classifier-based variant to improve efficiency, though this is not shown in the algorithm.

Model	Baselines	Accuracy (%)								
		BoolQ	**PIQA**	**SIQA**	**HellaS.**	**WinoG.**	**ARC-e**	**ARC-c**	**OBQA**	**Avg.**
LLaMA-3.2-3B	Random	50.67	74.33	45.67	35.33	46.00	53.67	34.00	35.67	46.92
	LossSel	51.33	79.00	44.33	49.33	45.67	56.33	43.00	35.67	50.58
	LESS	45.33	77.00	**52.00**	47.33	45.33	59.33	44.67	38.67	51.21
	AIM-C	52.67	77.33	50.33	43.33	**46.67**	**65.67**	46.67	42.00	53.08
	AIM	**53.67**	**80.33**	**52.00**	**51.33**	46.00	64.33	**47.00**	**42.33**	**54.62**
LLaMA-2-7B	Random	57.67	69.67	39.67	29.67	41.67	25.67	23.67	22.33	38.75
	LossSel	59.00	73.67	46.33	41.33	42.00	**36.67**	34.00	27.00	45.00
	LESS	60.67	72.67	42.33	36.00	43.67	26.00	24.67	22.33	41.04
	AIM-C	**61.67**	77.33	46.67	**46.00**	**44.33**	25.67	**36.66**	25.67	45.50
	AIM	**61.67**	**78.00**	**49.67**	**46.00**	**44.33**	**36.67**	34.33	**28.67**	**47.42**
LLaMA-3-8B	Random	61.67	83.33	60.00	65.67	41.33	73.33	55.33	60.00	64.12
	LossSel	63.33	82.33	**64.00**	61.67	46.67	74.00	55.00	46.67	61.71
	LESS	62.00	83.00	**64.00**	64.00	46.00	73.67	54.33	44.67	61.46
	AIM-C	63.00	81.00	63.33	63.33	46.00	75.33	57.67	50.33	62.08
	AIM	**66.33**	**84.33**	**64.00**	**68.33**	**48.00**	**84.67**	**62.67**	**60.33**	**67.33**

Table 1: Accuracy comparison of ReFT with baselines on commonsense reasoning tasks under three model scales. All models are trained on Commonsense170K and evaluated on the average performance across eight tasks. AIM and AIM-C denote our proposed full and classifier-based variants. Results are averaged over three random seeds

Experiments

In this section, we first introduce the experimental setup used in our study . We then report the model performance on commonsense reasoning datasets and arithmetic reasoning datasets.

Experimental Setup

For a fair and direct comparison, we follow the experimental setup detailed in the original ReFT (Wu et al. 2024). Specifically, we conduct experiments on two representative reasoning tasks, commonsense reasoning and arithmetic reasoning, covering more than 15 datasets. Our experiments cover the LLaMA model family at multiple scales, including LLaMA-3.2-3B, LLaMA-2-7B, and LLaMA-3-8B. All experiments are conducted on a single NVIDIA A100 40G/80G.

Datasets We replicate the experimental setup in (Wu et al. 2024). The training datasets for the commonsense and arithmetic reasoning tasks are derived from the COMMONSENSE170K and MATH10K benchmarks respectively, which aggregate eight individual commonsense tasks and seven distinct arithmetic benchmarks (Hu et al. 2023). Both training sets incorporate language model generated chain-of-thought steps to provide explicit reasoning trajectories throughout the fine-tuning process.

To evaluate the efficacy of our method in identifying noisy samples, we construct synthetic noisy datasets for both reasoning tasks by corrupting 25% of the training samples. For both commonsense and arithmetic reasoning, we exclusively employ a model-driven corruption strategy where a strong language model generates incorrect responses to replace the original ground truth labels. This approach is necessitated by the requirement to maintain logical coherence between the reasoning trajectories and the final conclusions which is typically compromised by simple label flipping in tasks involving chain-of-thought steps. Consequently, we utilize the entire incorrect response including both the reasoning path and the terminal answer as the noisy version to create realistic mislabeled instances that accurately simulate the annotation noise encountered in practical scenarios.

Baseline To evaluate the effectiveness of our approach, we compare it against a range of representative data filtering baselines: (1) **Random**, which uniformly samples a subset of training data as a naive baseline. (2) **Loss-Based Selection (LossSel)**, which selects high-loss examples after a short warm-up phase, assuming high loss correlates with informativeness or label noise, though it may also retain hard but useful samples. (3) **LESS** (Xia et al. 2024), which performs Low-rank gradient Similarity Search through the construction of a low-dimensional gradient datastore to estimate data influence and selects the most influential instruction examples that maximize the performance on targeted capabilities.

Commonsense Reasoning

Table 1 demonstrates that the proposed AIM method outperforms existing baselines across various model scales. On the LLaMA-3.2-3B backbone, AIM achieves the highest average accuracy of 54.62%, which represents a 3.41% improvement over the strongest selection baseline, LESS. For LLaMA-2-7B, AIM reaches an average accuracy of 47.42% and surpasses the most competitive baseline by a margin of 6.38%. AIM continues to deliver superior results on LLaMA-3-8B with an average accuracy of 67.33%, which validates its robustness as the model size increases. The

Model	Baselines	Accuracy (%)							
		MultiArith	GSM8K	SVAMP	MAWPS	AddSub	AQuA	SingEq	Avg.
LLaMA-3.2-3B	Random	79.67	31.33	54.00	74.79	80.00	18.90	82.00	60.10
	LossSel	77.33	**34.00**	53.67	72.69	80.67	21.65	81.67	60.24
	LESS	81.67	31.33	48.67	73.21	80.00	21.26	82.67	59.83
	AIM-C	**86.33**	29.33	51.33	**79.83**	79.33	**22.44**	84.67	61.90
	AIM	81.00	33.33	**55.33**	74.37	**81.67**	21.65	**86.67**	**62.00**
LLaMA-2-7B	Random	75.00	21.67	40.67	69.79	77.67	19.69	70.33	53.54
	LossSel	78.33	23.33	41.67	72.27	79.67	21.26	72.00	55.50
	LESS	77.33	**26.67**	**44.00**	71.95	81.67	20.80	74.33	56.68
	AIM-C	79.66	24.66	42.67	**73.11**	79.33	21.26	**82.67**	**57.62**
	AIM	**82.67**	24.67	42.67	71.01	**82.00**	**22.44**	75.00	57.21
LLaMA-3-8B	Random	90.00	61.67	70.67	84.87	85.82	29.13	90.94	73.30
	LossSel	90.33	70.00	75.00	86.13	83.67	27.95	91.00	74.87
	LESS	86.33	67.33	73.33	83.39	86.33	**30.71**	91.67	63.68
	AIM-C	**95.00**	66.67	75.00	**86.97**	87.33	30.31	**93.33**	76.37
	AIM	91.67	**71.00**	**75.33**	**86.97**	**89.67**	**30.71**	91.33	**76.67**

Table 2: Accuracy comparison of ReFT with various data selection methods on eight math reasoning datasets. We conduct experiments on three backbone models. Results are averaged over three random seeds.

competitive performance of AIM-C further indicates the effectiveness of the classifier-based subspace approximation strategy. Compared to methods such as LossSel and LESS that depend on training dynamics, AIM utilizes geometric consistency to achieve higher performance gains. These experimental results suggest that the geometry-guided selection strategy is scalable and enables the fine-tuning process to focus on high-quality samples.

Arithmetic Reasoning

Table 2 demonstrates that the proposed AIM method achieves superior performance across diverse arithmetic datasets and model scales. On the LLaMA-3-8B backbone, AIM reaches an average accuracy of 76.67%, which outperforms the strongest existing baseline, LossSel, by 1.80%. In the LLaMA-2-7B configuration, AIM improves the average accuracy to 57.21% and maintains a clear advantage over prior data selection techniques. The method exhibits significant efficacy on challenging reasoning benchmarks such as SVAMP and GSM8K, which indicates its capability to identify high-quality training samples essential for reasoning tasks. The classifier-based variant AIM-C also delivers competitive results and achieves a peak average accuracy of 57.62% on the LLaMA-2-7B model. Compared to dynamic filtering strategies including LossSel and LESS, the AIM framework shows enhanced robustness and generalization. These experimental findings suggest that leveraging intervention-based manifold alignment provides a beneficial inductive bias for sample selection in ReFT fine-tuning.

Ablation Studies

Impact of Segment Aggregation on Sample Scoring We examine how different segment level aggregation strategies affect AIM score based training data selection. Five strategies are compared including uniform averaging across all segments and weighted averaging based on segment reliability as well as using the AIM score from a single semantic segment denoted as Dominant Segment-e and Dominant Segment-m and Dominant Segment-l. All methods are evaluated on the commonsense reasoning task with three backbone models which are LLaMA-3.2-3B and LLaMA-2-7B and LLaMA-3-8B. Model performance is measured by precision which represents the ratio of truly correct instances within the subset selected as high quality by our scoring mechanism.

As shown in Figure 3, uniform averaging achieves the best results for the LLaMA-3-8B model while Dominant Segment-e provides the highest precision for the LLaMA-3.2-3B model. Weighted averaging maintains competitive performance across all scales but does not consistently outperform uniform aggregation. Dominant Segment-m and Dominant Segment-l yield stable results that are generally comparable to uniform averaging for the LLaMA-2-7B model. These findings suggest that the optimal aggregation strategy varies by model architecture and that leveraging early segment information is particularly beneficial for smaller scale backbones.

Impact of Variance Threshold on Selection Precision We investigate how the cumulative explained variance threshold ρ influences the precision of the selection process as a measure of data purity. As illustrated in Figure 4, we compare the performance of three models including LLaMA-3.2-3B and LLaMA-2-7B and LLaMA-3-8B across various ρ values. In this analysis precision represents the fraction of correctly labeled samples within the selected data. The experimental curves show that the peak precision

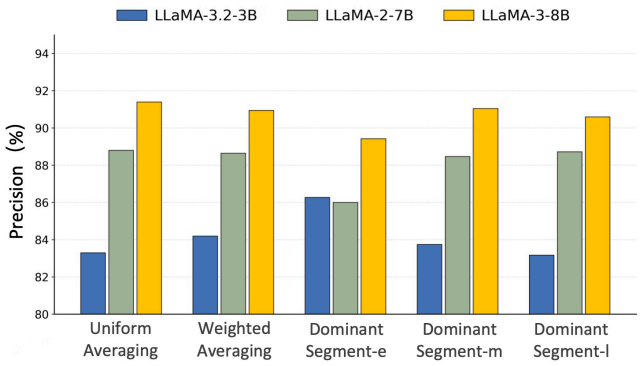

Figure 3: Precision comparison on the commonsense reasoning task across different segment-level AIM-score aggregation strategies where precision represents the proportion of accurately labeled samples in the selected subset to evaluate data purity. The comparison involves five strategies across three models including Weighted Averaging that employs adaptive weights based on informational richness. Uniform Averaging utilizes mean scores while the baseline strategies termed Dominant Segment-e, Segment-m, and Segment-l rely on the AIM-score from a specific architectural depth exclusively.

is consistently achieved at ρ equal to 0.90 for all evaluated models. While the LLaMA-3.2-3B model exhibits a slight further improvement at higher thresholds, the value of 0.90 remains a robust choice for maintaining high purity across different model scales. This observation confirms that preserving 90% of the total variance effectively captures essential informative components for data selection in ReFT.

Related Work

Learning with Noisy Labels and Data Filtering

Learning with noisy labels (LNL) aims to mitigate performance degradation caused by label errors (Song et al. 2022). A common strategy exploits training dynamics, as deep models tend to learn clean samples before memorizing noise (Arpit et al. 2017b). This led to small-loss heuristics and more principled methods like Confident Learning (Northcutt, Jiang, and Chuang 2022), Dataset Cartography (Swayamdipta et al. 2020b), and AUM (Pleiss et al. 2020), which use training behavior to score data. However, these methods often fail to distinguish noisy from difficult examples.

Model-consistency-based methods include Co-teaching (Han et al. 2018), DivideMix (Li, Socher, and Hoi 2020), and UNICON (Karim et al. 2022), which apply peer learning, loss modeling, or contrastive objectives to separate clean data. Recent approaches also use foundation models such as GPT-4 for label refinement (Gilardi, Alizadeh, and Kubli 2023; Yan et al. 2025b), but still rely heavily on loss statistics or external tools. In contrast, our method takes a geometry-based approach rooted in representation space.

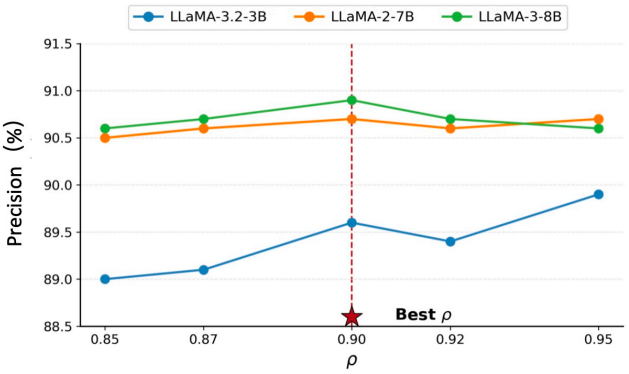

Figure 4: Precision performance across various values of the cumulative explained variance threshold ρ on a commonsense reasoning task where precision evaluates the purity of the filtered data by measuring the fraction of correctly labeled samples within the selected subset. Each individual curve represents the performance of a specific model architecture and the peak precision for all experimental configurations is consistently reached at $\rho = 0.90$.

Geometric Consistency in Representation Learning

The manifold hypothesis suggests that high-dimensional data lie on low-dimensional structures (Fefferman, Mitter, and Narayanan 2013). In deep learning, Manifold Mixup (Verma et al. 2019) and the Information Bottleneck (Tishby and Zaslavsky 2015) show that geometric regularities correlate with generalization (Wang et al. 2024b,a). This idea has been used for out-of-distribution detection (Lee et al. 2018; Yan et al. 2025a).

We extend this line by analyzing the geometric consistency of intervention vectors in ReFT (Wu et al. 2024), enabling a new paradigm for noise detection based on subspace alignment rather than task loss or predictions.

Conclusion

This paper introduces a novel data selection method tailored for representation fine-tuning, leveraging the unique structure of ReFT. By modeling intervention vectors and their alignment with reference manifolds, our approach captures geometric signals that reflect data quality. To our knowledge, this is the first work to incorporate manifold-based alignment into data filtering for fine-tuning. The method is fully unsupervised, requires no task labels, and generalizes across model scales. It is especially practical in real-world scenarios, where noisy or low-quality data are common and difficult to remove manually. Our results demonstrate that exploiting the internal structure of ReFT enables more robust and efficient training in the presence of data noise.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62332012).

References

Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; and Lacoste-Julien, S. 2017a. A Closer Look at Memorization in Deep Networks. arXiv:1706.05394.

Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; and Lacoste-Julien, S. 2017b. A Closer Look at Memorization in Deep Networks. arXiv:1706.05394.

Bietti, A.; and Mairal, J. 2019. On the Inductive Bias of Neural Tangent Kernels. arXiv:1905.12173.

Chen, N.; Wu, N.; Liang, S.; Gong, M.; Shou, L.; Zhang, D.; and Li, J. 2024. Is Bigger and Deeper Always Better? Probing LLaMA Across Scales and Layers. arXiv:2312.04333.

Fefferman, C.; Mitter, S.; and Narayanan, H. 2013. Testing the Manifold Hypothesis. arXiv:1310.0425.

Geiger, A.; Wu, Z.; Potts, C.; Icard, T.; and Goodman, N. 2024. Finding alignments between interpretable causal variables and distributed neural representations. In *Causal Learning and Reasoning*, 160–187. PMLR.

Gilardi, F.; Alizadeh, M.; and Kubli, M. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30).

Gurnee, W.; and Tegmark, M. 2024. Language Models Represent Space and Time. arXiv:2310.02207.

Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I.; and Sugiyama, M. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31.

He, J.; Fan, Z.; Kuang, S.; Xiaoqing, L.; Song, K.; Zhou, Y.; and Qiu, X. 2025. FiNE: Filtering and Improving Noisy Data Elaborately with Large Language Models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 8686–8707.

Hu, Z.; Wang, L.; Lan, Y.; Xu, W.; Lim, E.-P.; Bing, L.; Xu, X.; Poria, S.; and Lee, R. K.-W. 2023. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*.

Karim, N.; Rizve, M. N.; Rahnavard, N.; Mian, A.; and Shah, M. 2022. UNICON: Combating Label Noise Through Uniform Selection and Contrastive Learning. arXiv:2203.14542.

Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. arXiv:1807.03888.

Li, J.; Socher, R.; and Hoi, S. C. 2020. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*.

Li, M.; Zhang, Y.; He, S.; Li, Z.; Zhao, H.; Wang, J.; Cheng, N.; and Zhou, T. 2024. Superfiltering: Weak-to-strong data filtering for fast instruction-tuning. *arXiv preprint arXiv:2402.00530*.

Northcutt, C. G.; Jiang, L.; and Chuang, I. L. 2022. Confident Learning: Estimating Uncertainty in Dataset Labels. arXiv:1911.00068.

Papyan, V.; Han, X. Y.; and Donoho, D. L. 2020. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663.

Pleiss, G.; Zhang, T.; Elenberg, E. R.; and Weinberger, K. Q. 2020. Identifying Mislabeled Data using the Area Under the Margin Ranking. arXiv:2001.10528.

Song, H.; Kim, M.; Park, D.; Shin, Y.; and Lee, J.-G. 2022. Learning from Noisy Labels with Deep Neural Networks: A Survey. arXiv:2007.08199.

Swayamdipta, S.; Schwartz, R.; Lourie, N.; Wang, Y.; Hajishirzi, H.; Smith, N. A.; and Choi, Y. 2020a. Dataset cartography: Mapping and diagnosing datasets with training dynamics. *arXiv preprint arXiv:2009.10795*.

Swayamdipta, S.; Schwartz, R.; Lourie, N.; Wang, Y.; Hajishirzi, H.; Smith, N. A.; and Choi, Y. 2020b. Dataset Cartography: Mapping and Diagnosing Datasets with Training Dynamics. arXiv:2009.10795.

Tishby, N.; and Zaslavsky, N. 2015. Deep Learning and the Information Bottleneck Principle. arXiv:1503.02406.

Toneva, M.; Sordoni, A.; Combes, R. T. d.; Trischler, A.; Bengio, Y.; and Gordon, G. J. 2018. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*.

Turner, A. M.; Thiergart, L.; Leech, G.; Udell, D.; Vazquez, J. J.; Mini, U.; and MacDiarmid, M. 2023. Activation addition: Steering language models without optimization. *arXiv e-prints*, arXiv–2308.

Verma, V.; Lamb, A.; Beckham, C.; Najafi, A.; Mitliagkas, I.; Lopez-Paz, D.; and Bengio, Y. 2019. Manifold mixup: Better representations by interpolating hidden states. In *International conference on machine learning*, 6438–6447. PMLR.

Wang, J.; Cui, Z.; Wang, B.; Pan, S.; Gao, J.; Yin, B.; and Gao, W. 2024a. IME: Integrating multi-curvature shared and specific embedding for temporal knowledge graph completion. In *Proceedings of the ACM Web Conference 2024*, 1954–1962.

Wang, J.; Kai, S.; Luo, L.; Wei, W.; Hu, Y.; Liew, A. W.-C.; Pan, S.; and Yin, B. 2024b. Large language models-guided dynamic adaptation for temporal knowledge graph reasoning. *Advances in Neural Information Processing Systems*, 37: 8384–8410.

Wu, Z.; Arora, A.; Wang, Z.; Geiger, A.; Jurafsky, D.; Manning, C. D.; and Potts, C. 2024. Reft: Representation finetuning for language models. *Advances in Neural Information Processing Systems*, 37: 63908–63962.

Xia, M.; Malladi, S.; Gururangan, S.; Arora, S.; and Chen, D. 2024. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Yan, Z.; Wang, J.; Chen, J.; Li, X.; Liang, J.; Li, R.; and Pan, J. Z. 2025a. Atomic fact decomposition helps attributed question answering. *IEEE Transactions on Knowledge and Data Engineering*.

Yan, Z.; Wang, J.; Chen, J.; Wang, Y.; Tan, H.; Liang, J.; Li, X.; Li, R.; and Pan, J. Z. 2025b. Prompting large language models with partial knowledge for answering questions with unseen entities. *arXiv preprint arXiv:2508.01290*.

Yang, Y.; Mishra, S.; Chiang, J.; and Mirzasoleiman, B. 2024. Smalltolarge (s2l): Scalable data selection for fine-tuning large language models by summarizing training trajectories of small models. *Advances in Neural Information Processing Systems*, 37: 83465–83496.

Zhou, J.; Lu, T.; Mishra, S.; Brahma, S.; Basu, S.; Luan, Y.; Zhou, D.; and Hou, L. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.