From Insight to Action: A Novel Framework for Interpretability-Guided Data Selection in Large Language Models

Ling Shi[1], **Xinwei Wu**[1,2], **Xiaohu Zhao**[2], **Hao Wang**[2], **Heng Liu**[2], **Yangyang Liu**[2], **Linlong Xu**[2], **Longyue Wang**[2], **Deyi Xiong**[1]*, **Weihua Luo**[2],
[1]TJUNLP Lab, School of Computer Science and Technology, Tianjin University, China
[2]Alibaba Group, China
{shiling_100, wuxw2021, dyxiong}@tju.edu.cn

Abstract

While mechanistic interpretability tools like Sparse Autoencoders (SAEs) can uncover meaningful features within Large Language Models (LLMs), a critical gap remains in transforming these insights into practical actions for model optimization. We bridge this gap with the hypothesis that data selection guided by a model's internal task features is a effective training strategy. Inspired by this, we propose Interpretability-Guided Data Selection (IGDS), a framework that first identifies these causal task features through frequency recall and interventional filtering, then selects "Feature-Resonant Data" that maximally activates task features for fine-tuning. We validate IGDS on mathematical reasoning, summarization, and translation tasks within Gemma-2, LLaMA-3.1, and Qwen3 models. Our experiments demonstrate exceptional data efficiency: on the Math task, IGDS surpasses full-dataset fine-tuning by a remarkable **17.4%** on Gemma-2-2B while using only 50% of the data, and outperforms established baselines focused on data quality and diversity. Analysis confirms a strong positive correlation between feature amplification and task performance improvement. IGDS thus provides a direct and effective framework to enhance LLMs by leveraging their internal mechanisms, validating our core hypothesis.

1 Introduction

Large Language models (LLMs) have demonstrated increasingly superior performance across diverse downstream tasks (Liu et al., 2025b; Yang et al., 2025; Zhang and Xiong, 2025; Liu et al., 2024; Peng et al., 2025; Shi et al., 2026). Recent research in mechanistic interpretability has revealed that LLMs are not entirely black boxes; instead, they contain disentangled, human-understandable components (Gao et al., 2024a; Arditi et al., 2024). Discoveries such as steering vectors for factual knowledge (Ferrando et al., 2024) and sparse features for cross-modal entities (Lou et al., 2025) have provided invaluable *insights* into model mechanisms. However, while these insights are powerful for analysis, a critical gap remains in translating them into practical *actions* for model optimization (Rai et al., 2024; Sharkey et al., 2025).

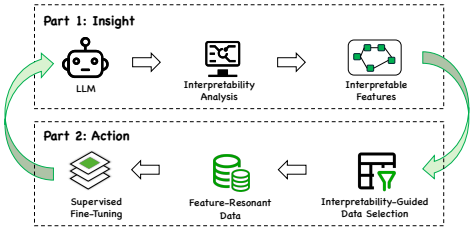


Figure 1: Conceptual illustration of the IGDS paradigm. The diagram depicts the closed loop from internal insight to optimization action, showing how model features are leveraged to guide data selection.

We bridge this gap with the hypothesis that data selection guided by a model's internal, causally-validated task features is a highly effective training strategy. To operationalize this, we propose **Interpretability-Guided Data Selection (IGDS)**, a framework that transforms interpretability insights into a tangible optimization pipeline. This conceptual loop, which we term *Insight2Action*, is visualized in Figure 1. Our core idea is to first identify the model's beneficial internal mechanisms and then select data that maximally activates them, which is called "Feature-Resonant Data", to reinforce these mechanisms through fine-tuning.

Specifically, IGDS operationalizes this vision through a two-stage process. The first stage, **Task Feature Identification**, moves from a broad *high-frequency recall* of candidate features using an SAE to a rigorous *causal intervention filtering* step, which isolates the potent subset directly impacting task performance. The second stage, **Feature-Based Data Scoring**, then uses this validated set of causal features to construct a highly effective utility

* Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6258–6272
July 2-7, 2026

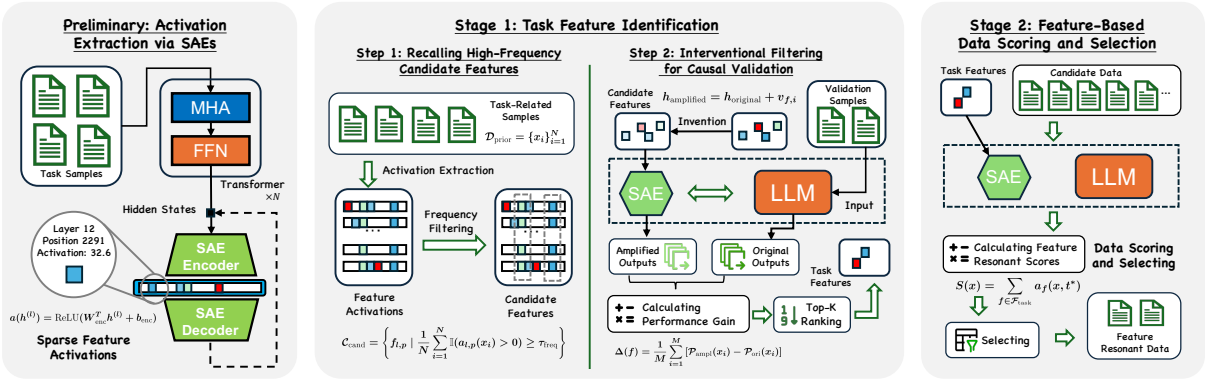


Figure 2: An overview of our Interpretability-Guided Data Selection (IGDS) framework.

function, selecting data that "resonates" with and reinforces the model's internal problem-solving structure.

We validate IGDS on mathematical reasoning, summarization, and translation tasks across Gemma-2, LLaMA-3.1, and Qwen3 models. Our experiments show that our method achieves the highest data efficiency across all settings. Notably, on the Math task, IGDS surpasses full-dataset fine-tuning by a remarkable **17.4%** on the Gemma-2-2B model while using only 50% of the data, and consistently outperforms established baselines focused on data quality and diversity. Furthermore, our analysis confirms a strong positive correlation between the targeted amplification of these internal features and the improvements in downstream task performance, providing strong mechanistic evidence for our method's success.

In summary, our key contributions are:

- We propose a novel optimization strategy that leverages a model's causally validated internal features to guide data selection, offering a direct path to efficient capability enhancement.
- We introduce IGDS, a general and practical framework that transforms descriptive interpretability insights into a prescriptive pipeline for identifying and selecting high-utility training data, effectively closing the loop from analysis to optimization.
- We provide extensive empirical validation across multiple models and tasks, demonstrating that IGDS surpasses both competitive baselines and full-dataset fine-tuning while utilizing only a fraction of the data.

2 Related Work

Mechanistic Interpretability Mechanistic interpretability (MI) aims to reverse-engineer Large Language Models (LLMs) to uncover the causal mechanisms behind their behaviors (Dunefsky et al., 2024; Sharkey et al., 2025). Recent breakthroughs have revealed that fine-grained semantic information is encoded within model representations, from neurons corresponding to specific concepts (Niu et al., 2024; Fang et al., 2024), to steering vectors that control high-level attributes like factuality and safety (Ferrando et al., 2024; Yi et al., 2025), and sparse features that relevant to downstream tasks (Lu et al., 2025; Shu et al., 2025). However, most MI research currently stops at the explanatory level, further exploration into how these insights can proactively build better models is often neglected (Rai et al., 2024; Sharkey et al., 2025). While prior research has begun to explore applications, such as controlling model behavior via inference-time interventions (Wu et al., 2024; Ghosh et al., 2025) or refining reward models specifically for safety alignment (Zhang et al., 2025; Liu et al., 2025a), a general pipeline for leveraging these insights to guide the broader training process remains absent. This work aims to bridge this gap, establishing a framework that turns descriptive interpretability findings into a prescriptive guide for model optimization.

Sparse Autoencoders Sparse Autoencoders (SAEs) have emerged as a pivotal tool in MI for addressing the superposition hypothesis (Sharkey et al., 2025). This premise suggests that densely parameterized models frequently superimpose multiple independent semantic signals onto individual neuronal activations. Such polysemanticity obscures interpretability at both the single-unit

and embedding-vector scales (Elhage et al., 2022; Gurnee et al., 2023). By untangling these entangled representations into a sparse basis of semantically coherent dimensions, SAEs markedly improve architectural transparency (Cunningham et al., 2023; Gao et al., 2024b). Current applications of SAEs typically focus on identifying task-specific features to enable precise, post-hoc interventions on model behavior (Templeton et al., 2024; Farrell et al., 2024; Wu et al., 2026). While effective for changing outputs, such interventional approaches can suffer from high latency and instability in practical applications. Departing from this paradigm, we leverage SAE-identified features to guide the data selection process, establishing a new pathway from internal insight to optimization action.

Data Selection Training data remains a critical determinant of model efficacy (Sun et al., 2024; Yang et al., 2026; Pan et al., 2025). The paradigm for post-training data selection has shifted from prioritizing quantity to emphasizing quality (Zhou et al., 2023), giving rise to automated methods that score data based on **data quality** using external models (Chen et al., 2024) or self-assessed metrics (Li et al., 2024; Xia et al., 2024a). Beyond quality, **data diversity** is recognized as crucial for model robustness, with frameworks like *ZIP* proposed to balance this trade-off (Bukharin et al., 2024; Yin et al., 2024). However, the efficacy of these methods is scrutinized at scale, as recent studies suggest they often yield negligible gains over simple random selection (Xia et al., 2024b). Departing from prior approaches that rely on external signals and treat the model as a black box, we posit that a direct and more potent signal for data utility resides within the model itself.

3 Methodology

In this section, we detail the proposed Interpretability-Guided Data Selection (IGDS) framework, which operationalizes interpretability insights into a rigorous data selection strategy. As illustrated in Figure 2, the framework proceeds in two main stages: (1) **Task Feature Identification**, where we isolate features that are not merely correlated with, but causally linked to task performance; and (2) **Feature-Based Data Scoring**, where we leverage these validated features to quantify data utility and curate high-quality subsets for supervised fine-tuning.

3.1 Preliminaries: Activation Extraction via SAEs

Our framework leverages Sparse Autoencoders (SAEs) to transform compact neural activations into a sparse, expansive representation where individual dimensions carry distinct semantic meaning. Given an input sequence, we first extract the hidden state $\boldsymbol{h}^{(l)} \in \mathbb{R}^{d_{\text{model}}}$ from a specific layer l via a forward pass. The SAE then projects this dense representation into a sparse feature activation vector $\boldsymbol{a}(\boldsymbol{h}^{(l)}) \in \mathbb{R}^{d_{\text{sae}}}$ using an encoder parameterized by the weight matrix $\boldsymbol{W}_{\text{enc}}$:

$$\boldsymbol{a}(\boldsymbol{h}^{(l)}) = \text{ReLU}(\boldsymbol{W}_{\text{enc}}^T \boldsymbol{h}^{(l)} + \boldsymbol{b}_{\text{enc}}), \quad (1)$$

where $d_{\text{sae}} \gg d_{\text{model}}$. Each dimension p in this vector corresponds to a feature $f_{l,p}$, with its value indicating the feature's activation magnitude. Our goal is to navigate this vast feature space and pinpoint the precise subset of features that exert a causal influence on the target task.

3.2 Stage 1: Task Feature Identification

To identify these features, we employ a coarse-to-fine filtering strategy that distills the vast feature space through two sequential steps: high-frequency recalling and interventional filtering.

Recalling High-Frequency Candidate Features A feature fundamental to a specific task is expected to activate consistently during task execution (Geiger et al., 2024). Based on this principle, we begin by identifying candidate features that exhibit a strong correlation with the target task. Utilizing a small corpus of N prior task-related samples, denoted as $\mathcal{D}_{\text{prior}} = \{x_i\}_{i=1}^N$, we monitor feature activations at the critical token position (e.g., the final token of the prompt), with specific positions for each task listed in Appendix 9. Let $a_{l,p}(x_i)$ denote the activation magnitude of feature $f_{l,p}$ for sample x_i at this step.

A feature $f_{l,p}$ is selected as a candidate if its activation frequency, defined as the proportion of samples in $\mathcal{D}_{\text{prior}}$ where the feature is active, exceeds a predefined threshold τ_{freq} (e.g., 80%). Formally, we define the set of candidate features, $\mathcal{C}_{\text{cand}}$, as follows:

$$\mathcal{C}_{\text{cand}} = \left\{ f_{l,p} \mid \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(a_{l,p}(x_i) > 0) \geq \tau_{\text{freq}} \right\}, \quad (2)$$

where $\mathbb{I}(\cdot)$ is the indicator function, returning 1 if the condition holds and 0 otherwise. This initial

step effectively filters out the vast majority of irrelevant features, yielding a manageable candidate set for subsequent rigorous validation.

Interventional Filtering for Causal Validation High activation frequency implies correlation, but not necessarily causation. A feature that consistently activates on task data might merely capture general linguistic patterns rather than task-specific mechanisms (Deng et al., 2025). To isolate features with genuine causal efficacy, we perform a rigorous filtering step via targeted interventions on a small validation set of M samples, $\mathcal{D}_{\text{val}} = \{x_i\}_{i=1}^{M}$.

For each candidate feature $f \in \mathcal{C}_{\text{cand}}$, we quantify its causal impact by measuring how amplifying its activation affects the model's performance. This is achieved by adding the feature's influence vector to the model's residual stream and evaluating the resulting change in task-specific performance. Specifically, for each validation sample x_i, we compute the feature's influence vector $\boldsymbol{v}_{f,i}$ as the product of its current activation and its corresponding SAE decoder weight:

$$\boldsymbol{v}_{f,i} = a(h_i) \cdot \boldsymbol{W}_{\text{dec},f}, \tag{3}$$

where $a(h_i)$ denotes the scalar activation of feature f on the hidden state h_i, and $\boldsymbol{W}_{\text{dec},f}$ represents the decoder weight vector for feature f. We then generate two outputs: one from the original model, and one from an "amplified" counterpart where the feature's influence vector is directly added to the residual stream:

$$\boldsymbol{h}_{\text{ampl}} = \boldsymbol{h}_{\text{ori}} + \boldsymbol{v}_{f,i}, \tag{4}$$

Let $\mathcal{P}_{\text{ori}}(x_i)$ and $\mathcal{P}_{\text{ampl}}(x_i)$ denote the task performance scores (e.g., COMET for translation) for the original and amplified outputs, respectively. The causal impact of feature f is defined as the average performance gain across the validation set:

$$\Delta(f) = \frac{1}{M} \sum_{i=1}^{M} \left[\mathcal{P}_{\text{ampl}}(x_i) - \mathcal{P}_{\text{ori}}(x_i) \right], \tag{5}$$

A significantly positive $\Delta(f)$ indicates that amplifying the feature consistently enhances the model's task capability, validating the feature as a positive causal driver. Finally, we rank all candidates by $\Delta(f)$ and select the top-K features to form the final validated set of task features, $\mathcal{F}_{\text{task}}$. This interventional filtering ensures the retention of features that are demonstrably beneficial for the target task.

3.3 Stage 2: Feature-Based Data Scoring and Selection

With the causally-validated set of task-relevant features $\mathcal{F}_{\text{task}}$ identified, we leverage this subset as an intrinsic lens to quantify the utility of each candidate data point from a large pool $\mathcal{D}_{\text{pool}}$. Our central hypothesis posits that the most valuable data for fine-tuning are those that maximally activate the model's internal causal mechanisms associated with the task.

To operationalize this, we introduce the **Feature-Resonant Score (FRS)**, denoted as $S(x)$. This score is computed for each data point x by aggregating the activation magnitudes of all task features at the same critical token position, t^*, consistent with the identification phase in Stage 1. Formally, the score is defined as:

$$S(x) = \sum_{f \in \mathcal{F}_{\text{task}}} a_f(x, t^*), \tag{6}$$

where $a_f(x, t^*)$ represents the activation of task feature f at position t^* for input x. By design, this formulation directly prioritizes data that elicits a strong, collective response from the specific features underpinning the desired capability.

Finally, we rank all data points in $\mathcal{D}_{\text{pool}}$ by their FRS and select a subset based on a predefined ratio. This process yields our final training dataset, $\mathcal{D}_{\text{SFT}}$, a high-potency subset of the original corpus, densely packed with targeted, task-relevant signals.

4 Experiments

To validate the effectiveness and robustness of our Interpretability-Guided Data Selection (IGDS) framework, we conducted a comprehensive set of experiments across diverse tasks, models, and competitive baselines.

4.1 Tasks and Evaluation

For each task, we strictly enforced data separation to prevent information leakage. Specifically, we distinctively defined three subsets: the task-related set for feature identification (Stage 1), a large candidate pool for selection (Stage 2), and a held-out test set for final evaluation. Detailed prompt templates for these stages are provided in Appendix A.3. The specific setup for each task is detailed below:

Mathematical Reasoning The selection pool ($\mathcal{D}_{\text{pool}}$) comprises 93.7K samples from the `OpenR1-Math-220k` dataset. For feature identification, we

Table 1: Results of the task feature identification. For each model and task, we report the proportion of high-frequency candidates in basis points (*Recalled (‱)*, where 100‱= 1%), the specific feature with the top-1 positive impact (*Feature*), and its corresponding performance gain (Δ).

	Math			Summarization			Translation		
Model	Recalled (‱)	Feature	Δ(ACC)	Recalled (‱)	Feature	Δ(ROUGE-1)	Recalled (‱)	Feature	Δ(COMET)
Gemma-2-2B-it	9.20	114_p11575	+12	8.89	125_p3017	+0.025	4.82	111_p892	+1.12
LLaMA-3.1-8B-it	2.76	119_p16897	+1.5	3.40	131_p15962	+0.022	4.91	18_p2083	+4.98
Qwen3-8B	1.33	116_p36564	+1.8	0.35	118_p61304	+0.018	2.94	112_p43296	+8.34

utilized the training set of gsm8k (Zeng et al., 2023). Final model performance was evaluated on the MATH-500 benchmark.

Summarization We utilized the DialogSum dataset (Chen et al., 2021). The official training split (12.5K samples) served as the selection pool ($\mathcal{D}_{\text{pool}}$), while the validation split (500 samples) was employed for feature identification. Performance was measured on the official test split (1.5k samples).

Machine Translation The selection pool ($\mathcal{D}_{\text{pool}}$) consists of 10K English-to-Chinese pairs randomly sampled from the WMT24 dataset (Kocmi et al., 2024). From the same source, a separate set of 500 samples was randomly held out to serve as the set for feature identification. Evaluation was conducted on the WMT24++ test set (Deutsch et al., 2025) (997 samples).

Evaluation Metrics We report performance using standard, task-specific metrics. For Mathematical Reasoning, we employ pass@8 accuracy. For Summarization, we report ROUGE-1 scores in the main text, with full ROUGE-1/2/L results provided in the Appendix A.2. For Machine Translation, we utilize the COMET score (Rei et al., 2020).

4.2 Models and SAEs

Our experiments encompass three model families: Gemma-2 (Team, 2024a), LLaMA-3.1 (Team, 2024b), and Qwen3 (Team, 2025). We adopted a strategic two-stage setup: task features were identified using the publicly available instruction-tuned versions, as these models possess the requisite task awareness for meaningful feature discovery (Xia et al., 2024a). The subsequent fine-tuning was then performed on the corresponding base models to cleanly evaluate the impact of the selected data. This setup ensures that any observed performance gains are attributed directly to the quality of the selected data, rather than confounding factors from the initial instruction tuning.

For feature extraction, we utilized SAEs specific to models. For Gemma-2 and LLaMA-3.1, we leveraged publicly available pre-trained SAEs (Lieberum et al., 2024; He et al., 2024). In the absence of publicly available SAEs for Qwen3, we trained a custom instance using the JumpReLU architecture (He et al., 2024). Full training details for this custom SAE are provided in Appendix A.1.

4.3 Baselines

To comprehensively evaluate the IGDS framework, we benchmarked it against a spectrum of data selection baselines and standard controls. We compared IGDS against three distinct selection strategies: (1) **Quality-based** methods, including *IFD* (Li et al., 2024) targeting instruction-following difficulty, and *Loss*, a perplexity filter prioritization samples with the lowest Cross-Entropy; (2) A **Diversity-based** approach, *ZIP* (Yin et al., 2024), designed to maximize semantic coverage; and (3) *Random* selection, which serves as a robust, data-agnostic baseline to validate the necessity of intelligent selection metrics. To contextualize performance, we establish two standard controls: *Original*, evaluating the zero-shot capabilities of the base model without SFT, and *Full*, utilizing the entire training pool as a reference for data efficiency.

4.4 Main Results

Results of Task-Specific Features Identification We initiate our analysis by examining the efficacy of the feature identification stage, with key statistics summarized in Table 1. The process begins by applying a frequency filter to the vast search space comprising millions of potential SAE features. This initial step proves highly selective, drastically reducing the vast search space. As shown in the *Recalled* column, high-frequency candidates represent a minute fraction of the total, often amounting to just a few basis points (‱).

From this condensed pool, our causal validation step consistently identifies task features whose amplification yields a positive performance impact

Table 2: Performance comparison across three tasks: Math (pass@8), Summarization (ROUGE-1), and Translation (COMET). The best result among all fine-tuning methods for each model-task pair is in **bold**. The subscript denotes the gap relative to the performance of full SFT.

		Standard Controls		Data Selection Baselines				
Task	Model	Original	Full	Random	Loss	IFD	ZIP	**IGDS**
Math	Gemma-2-2B	11.2	32.2	$29.2_{-9.3\%}$	$24.6_{-23.6\%}$	$30.0_{-6.8\%}$	$29.4_{-8.7\%}$	$\mathbf{37.8}_{+17.4\%}$
	Llama-3.1-8B	26.6	45.0	$40.8_{-9.3\%}$	$37.2_{-17.3\%}$	$36.8_{-18.2\%}$	$35.4_{-21.3\%}$	$\mathbf{45.8}_{+1.8\%}$
	Qwen3-8B-Base	55.7	60.8	$58.7_{-3.5\%}$	$59.4_{-2.3\%}$	$60.5_{-0.5\%}$	$60.3_{-0.8\%}$	$\mathbf{61.1}_{+0.4\%}$
Sum	Gemma-2-2B	0.220	0.450	$0.439_{-2.4\%}$	$0.448_{-0.6\%}$	$0.440_{-2.2\%}$	$0.449_{-1.2\%}$	$\mathbf{0.452}_{+0.4\%}$
	Llama-3.1-8B	0.011	0.260	$0.254_{-2.3\%}$	$0.254_{-2.3\%}$	$0.258_{-0.8\%}$	$0.245_{-5.8\%}$	$\mathbf{0.261}_{+0.4\%}$
	Qwen3-8B-Base	0.312	0.489	$0.482_{-1.4\%}$	$0.474_{-3.1\%}$	$0.488_{-0.2\%}$	$0.485_{-0.8\%}$	$\mathbf{0.490}_{+0.2\%}$
Trans	Gemma-2-2B	26.92	65.93	$62.92_{-4.6\%}$	$55.31_{-16.1\%}$	$64.19_{-2.6\%}$	$63.37_{-3.9\%}$	$\mathbf{68.14}_{+3.4\%}$
	Llama-3.1-8B	31.11	77.68	$70.78_{-8.9\%}$	$72.29_{-6.9\%}$	$76.82_{-1.1\%}$	$74.49_{-4.1\%}$	$\mathbf{78.45}_{+1.0\%}$
	Qwen3-8B-Base	81.30	82.71	$82.61_{-0.1\%}$	$82.66_{-0.1\%}$	$82.69_{-0.0\%}$	$82.57_{-0.2\%}$	$\mathbf{83.07}_{+0.4\%}$

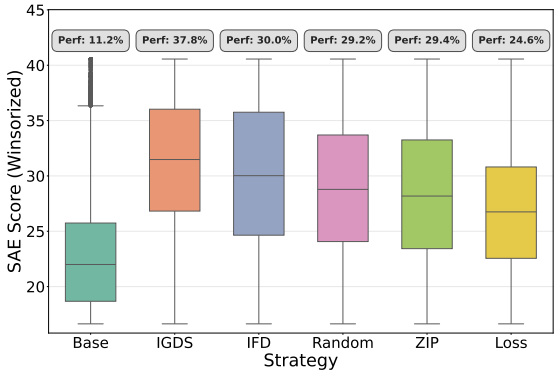


Figure 3: Correlation between feature activation and task performance on the Math task for Gemma-2-2B.

on the validation set. The magnitude of this impact, reported in the Δ column, is often substantial. We observe particularly striking efficacy in specific cases: for instance, `Gemma-2-2B` achieves a performance increase of 12 points on the Math task, while `Qwen3-8B` sees a massive gain of +8.34 on Translation. The consistent discovery of such high-impact features across diverse models and tasks provides compelling evidence that our framework reliably identifies features that are not merely correlated with, but causally instrumental to, the model's task-solving capabilities.

Performance of IGDS framework Utilizing the identified features, we selected the top 50% of the candidate pool to fine-tune base models. Table 2 provides a comprehensive comparison against all baselines and controls.

The results show IGDS consistently and substantially outperforms all other data selection methods across every tested model and task. Furthermore, our method surpasses the performance of full-data fine-tuning in multiple scenarios. As indicated by the positive subscripts in Table 2, IGDS even achieves a striking +17.4% relative gain for Gemma-2-2B on Math. This phenomenon strongly validates our core hypothesis: selecting data through the lens of the model's own causally-validated mechanisms is a effective strategy for targeted model improvement.

4.5 Validating the Role of Task Features

To corroborate the link between our identified features and downstream task proficiency, we analyzed the post-fine-tuning activation distributions of a key task feature. Figure 3 shows the results for the top-ranked Math feature, `114_p11575`, in the `Gemma-2-2B` model. We plot its activation distribution across the training set for the base model and for models fine-tuned with various data selection strategies, alongside the final task performance.

Two key observations emerge from our analysis. First, even baseline strategies that do not explicitly target this feature (e.g., *Random*, *IFD*) implicitly enhance its activation levels compared to the base model. This suggests that the feature is intrinsically aligned with the underlying optimization mechanics of the Math task. Second, and most crucially, there is a strong positive correlation between elevated activation magnitude and optimized task performance. The IGDS-trained model not only exhibits the highest median feature activation but also achieves the superior performance score of 37.8. Furthermore, a consistent trend is evident across all methods: the hierarchy of median feature activations closely mirrors the ranking of final

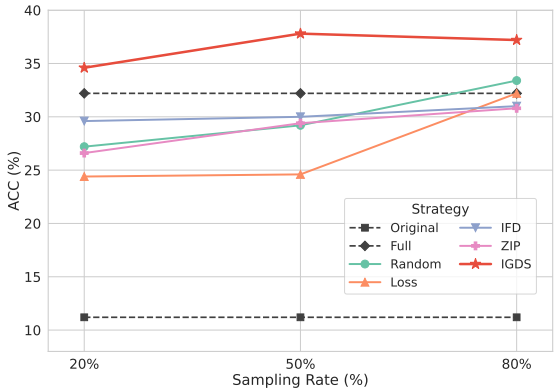


Figure 4: Performance comparison of different data selection strategies under varying sampling rates (20%, 50%, 80%) with `Gemma-2-2B` model.

Table 3: Ablation study of the IGDS framework on the `Gemma-2-2B` model.

Method	Math	Sum	Trans
Full IGDS (k=1)	**37.8**	**0.452**	**68.14**
Ablation on Feature Identification Stage			
w/o Frequency Recalling	29.2	0.439	63.19
w/o Causal Filtering	33.0	0.434	64.52
Ablation on Data Selection Stage (Varying k)			
$k = 3$	37.2	0.447	66.81
$k = 5$	31.8	0.435	62.83

performance. Collectively, these findings provide compelling evidence that the features identified by our framework are causally instrumental to the model's task-solving capabilities.

5 Robustness Analysis

Beyond standard performance benchmarks, we further investigate the practical viability of the IGDS framework. This section demonstrates the method's consistent superiority across varying data budgets, validates the contribution of its core components via ablation studies, and confirms its computational efficiency, solidifying its standing as a practical and reliable solution.

5.1 Effect of Different Sampling Ratios

To assess the robustness of our method under varying data budgets, we conducted experiments using sampling ratios of 20%, 50%, and 80% of the full training set, specifically on the Math task with `Gemma-2-2B`. We benchmarked the proposed IGDS against all competitive baselines.

As illustrated in Figure 4, our IGDS exhibits overwhelming superiority across all tested sampling ratios. Notably, at every data budget, IGDS not only outperforms all other selection baselines by a significant margin but also consistently surpasses the performance of fine-tuning on the *Full* dataset. These findings underscore the superior data efficiency of IGDS, demonstrating its capability to curate smaller, yet higher-utility subsets for effective and economical model training.

Table 4: Time cost comparison of data selection strategies on `LLaMA-3.1-8B`. The raw time costs (in hours) are normalized relative to the `Loss` strategy (100%).

Method	Math	Sum	Trans
Loss	3.0	0.9	0.5
IDF	6.5 (+167%)	1.2 (+33%)	0.7 (+40%)
ZIP	1.6 (-47%)	0.4 (-56%)	0.2 (-60%)
IGDS	2.5 (-17%)	0.7 (-22%)	0.4 (-20%)

5.2 Ablation Study

To dissect the contributions of our framework's key components, we conducted an ablation study on the `Gemma-2-2B` model, with results presented in Table 3. First, we first assess the impact of the feature identification pipeline. Replacing our frequency-based recalling with random selection (*w/o Frequency Recalling*) leads to a sharp performance decline, underscoring the necessity of pre-screening the vast search space for relevant candidate features. Similarly, removing the causal filtering step (*w/o Causal Filtering*) and instead using all recalled features for scoring also significantly degrades performance. This confirms that causal validation is indispensable for distinguishing genuine task features from merely correlated noise.

Next, we examined sensitivity to k, the number of top features used for scoring. While our default setting ($k = 1$) yields the best results, performance remains robust at $k = 3$. However, increasing k further to 5 results in a noticeable drop. This observation suggests that a highly focused feature set is preferable, as including less impactful features likely introduces introduce noise and dilutes the selection quality.

5.3 Time Cost of Data Selection

In addition to model performance, the computational efficiency of a data selection strategy is a critical determinant for its practical application. To evaluate this, we conducted a runtime analysis on

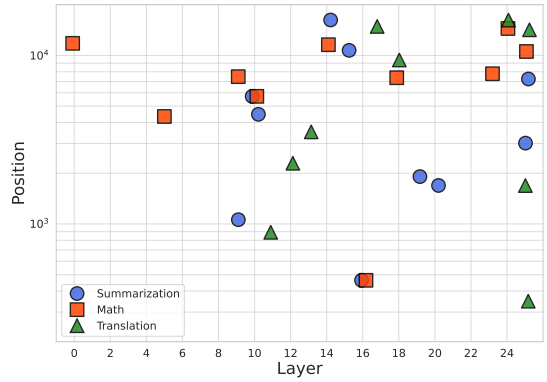


Figure 5: Distribution of positive features across layers and positions for `Gemma-2-2B` model. Each point represents a feature, plotted by its layer (x-axis) and position (y-axis, log scale).

the `LLaMA-3.1-8B` model by sampling 50% of the data across three distinct tasks: Math, Summarization, and Translation. We measured the time cost of each strategy and normalized it relative to the `Loss` strategy, which serves as our 100% baseline as it requires a single forward pass per sample.

As shown in Table 4, the model-free `ZIP` strategy is naturally the most efficient (almost -60%), whereas `IFD` incurs substantial overhead (+167%) due to its need for extra inference in Math task. Our IGDS method demonstrates high efficiency, reducing the computational cost by approximately 20% overhead. This is because its computation is integrated directly into the single forward pass required for gradient calculation. This result confirms that IGDS provides a practical solution for data selection without compromising on efficiency.

6 Interpretability Analysis

In this section, we analyze the structural distribution of the identified task-specific features and demonstrate their stability, confirming that they represent robust and intrinsic model properties.

6.1 Distribution of Task-Specific Features

To investigate where task-specific knowledge is encoded within the model, we visualize the distribution of identified features for Math, Summarization, and Translation tasks on the `Gemma-2-2B` model. As shown in Figure 5, we plot each feature's layer against its position. Due to the vastness of the SAE's feature space, the position axis is presented on a logarithmic scale. The results reveal distinct, task-dependent structural preferences. Specifically, Math features are widely dispersed across all layers, indicating that mathematical reasoning is a full-stack capability engaging the entire model depth. Conversely, Summarization and Translation features are concentrated in the middle-to-late layers, indicating a reliance on deep semantic processing. This differential distribution highlights that distinct cognitive abilities are localized in different structural regions of the model, underscoring the value of our fine-grained, task-specific analysis.

Table 5: Stability of identified task features for Math on `Gemma-2-2B-it` using different prior datasets. The top-5 features with the highest positive impact are listed.

Dataset	GSM8K	Math500	OpenR1
Top-1	$F_{14,11575}$	$F_{14,11575}$	$F_{14,11575}$
Top-2	$F_{18,4651}$	$F_{24,14448}$	$F_{18,4651}$
Top-3	$F_{23,7783}$	$F_{10,5717}$	$F_{24,14448}$
Top-4	$F_{24,14448}$	$F_{25,10550}$	$F_{10,5717}$
Top-5	$F_{10,5717}$	$F_{11,892}$	$F_{5,4341}$

6.2 Stability of Task-Specific Features

A crucial question for interpretability is whether identified features represent intrinsic model capabilities or are merely dataset-specific biases derived from the discovery source. To rigorously evaluate this stability, we performed independent feature identification runs on the Math task with `Gemma-2-2B-it`, utilizing three distinct datasets, GSM8K, Math500, and OpenR1, as separate identification sources. The results, presented in Table 5, reveal a remarkable degree of consistency across these diverse contexts.

Most strikingly, the top-ranked feature, $F_{14,11575}$, consistently emerges as the most impactful driver for mathematical reasoning, irrespective of the source dataset. Furthermore, we observe a significant overlap within the top-5 ranks, with three specific features ($F_{14,11575}$, $F_{24,14448}$, and $F_{10,5717}$) persisting across all three settings. This high degree of stability across varying data distributions provides compelling evidence that our framework is not simply finding patterns specific to one dataset's style or content. Instead, it successfully identifies features that are fundamental and intrinsic to the model's core mechanism for mathematical reasoning.

(a) Translation Task (b) Summarization Task (c) Math Task

Figure 6: Word clouds showing the terms with the most significant frequency increase after amplifying the top-ranked task feature for each task. The size of a word corresponds to the magnitude of its frequency change.

6.3 Correlation Analysis of Task Feature Identification

To better elucidate the semantic roles of the identified features, we qualitatively examine lexical shifts in the model's output when amplifying the top-ranked feature for each task. Figure 6 visualizes the vocabulary exhibiting the most pronounced frequency increases, revealing a clear alignment with the underlying task semantics. For instance, amplifying the **Translation** feature consistently elevates language-specific terms such as "english" and "chinese" (Figure 6a). Likewise, strengthening the **Summarization** feature drives up the prominence of structural cues like "summary" and "topic" (Figure 6b), whereas activating the **Math** feature yields a marked increase in solution-oriented vocabulary such as "answer" and "final" (Figure 6c). This qualitative evidence appears consistent with our quantitative findings, supporting the notion that our data selection strategy is guided by features that encode meaningful, task-relevant concepts.

7 Conclusion

In this work, we have established and validated a new principle for LLM optimization: the most potent signals for data selection reside within the model's own internal causal mechanisms. We have presented Interpretability-Guided Data Selection (IGDS), a practical framework that operationalizes this principle by first identifying causally-validated task features, and subsequently selecting "Feature-Resonant Data" that maximally activates them. Our empirical results are compelling: IGDS not only consistently outperforms competitive baselines but can surpass full-dataset fine-tuning, achieving a remarkable **17.4%** gain on the Math task using only half the data. Ultimately, IGDS pioneers a new class of optimization techniques that leverage interpretability, effectively bridging the gap between mechanistic understanding and practical model improvement.

Limitations

While IGDS presents a promising paradigm for data selection, we acknowledge that its performance is inherently linked to the quality of the underlying Sparse Autoencoders (SAEs). This dependency is empirically reflected in our results presented in Table 2, where model performance correlates strongly with the maturity of the available SAE ecosystems.

Specifically, for `Gemma-2-2B`, we utilized the high-quality, officially released Gemma-Scope SAEs. This resulted in the most significant performance improvements, with a particularly striking gain of $+\mathbf{17.4}\%$ on the Math task. In contrast, due to computational constraints, our custom SAE training for `Qwen3-8B` was restricted to a limited subset of layers. We hypothesize that this partial coverage of the model's layers is the primary reason for the more modest gains observed for Qwen3-8B, which were among the lowest across all tasks.

This performance disparity underscores that the quality and comprehensiveness of the SAEs are critical factors that directly influence the efficacy of our method. To address this bottleneck and to contribute to the broader research community, we plan to train SAEs for the remaining layers of `Qwen3-8B` in our future work. We are committed to open-sourcing these artifacts upon completion to facilitate further research in this area.

Acknowledgments

The present research was supported by the National Key Research and Development Program of China (Grant No. 2023YFE0116400). We would like to thank the anonymous reviewers for their insightful comments.

References

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083.

Alexander Bukharin, Shiyang Li, Zhengyang Wang, Jingfeng Yang, Bing Yin, Xian Li, Chao Zhang, Tuo Zhao, and Haoming Jiang. 2024. Data diversity matters for robust instruction tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3411–3425.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. 2024. AlpaGasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Ruixuan Deng, Xiaoyang Hu, Miles Gilberti, Shane Storks, Aman Taxali, Mike Angstadt, Chandra Sripada, and Joyce Chai. 2025. Sparse feature coactivation reveals composable semantic modules in large language models. *arXiv preprint arXiv:2506.18141*.

Daniel Deutsch, Eleftheria Briakou, Isaac Rayburn Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. WMT24++: expanding the language coverage of WMT24 to 55 languages & dialects. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Findings of ACL, pages 12257–12284. Association for Computational Linguistics.

Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. 2024. Transcoders find interpretable llm feature circuits. *Advances in Neural Information Processing Systems*, 37:24375–24410.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger B. Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. 2022. Toy models of superposition. *CoRR*, abs/2209.10652.

Junfeng Fang, Zac Bi, Ruipeng Wang, Houcheng Jiang, Yuan Gao, Kun Wang, An Zhang, Jie Shi, Xiang Wang, and Tat-Seng Chua. 2024. Towards neuron attributions in multi-modal large language models. *Advances in Neural Information Processing Systems*, 37:122867–122890.

Eoin Farrell, Yeu-Tong Lau, and Arthur Conmy. 2024. Applying sparse autoencoders to unlearn knowledge in language models. In *Neurips Safe Generative AI Workshop 2024*.

Javier Ferrando, Oscar Balcells Obeso, Senthooran Rajamanoharan, and Neel Nanda. 2024. Do i know this entity? knowledge awareness and hallucinations in language models. In *The Thirteenth International Conference on Learning Representations*.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024a. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024b. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah Goodman. 2024. Finding alignments between interpretable causal variables and distributed neural representations. In *Causal Learning and Reasoning*, pages 160–187. PMLR.

Shaona Ghosh, Amrita Bhattacharjee, Yftah Ziser, and Christopher Parisien. 2025. A simple yet effective method for non-refusing context relevant fine-grained safety steering in llms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 35116–35136.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, Yu-Gang Jiang, and Xipeng Qiu. 2024. Llama scope: Extracting millions of features from llama-3.1-8b with sparse autoencoders. *CoRR*, abs/2410.20526.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondrej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena

Karpinska, Philipp Koehn, Benjamin Marie, Christof Monz, Kenton Murray, Masaaki Nagata, Martin Popel, Maja Popovic, and 3 others. 2024. Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet. In *Proceedings of the Ninth Conference on Machine Translation, WMT 2024, Miami, FL, USA, November 15-16, 2024*, pages 1–46. Association for Computational Linguistics.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2024. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 7595–7628. Association for Computational Linguistics.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. 2024. Gemma scope: Open sparse autoencoders everywhere all at once on gemma 2. *arXiv preprint arXiv:2408.05147*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Chuang Liu, Linhao Yu, Jiaxuan Li, Renren Jin, Yufei Huang, Ling Shi, Junhui Zhang, Xinmeng Ji, Tingting Cui, Tao Liu, Jinwang Song, Hongying Zan, Sun Li, and Deyi Xiong. 2024. OpenEval: Benchmarking chinese llms across capability, alignment and safety. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 190–210. Association for Computational Linguistics.

Dengcan Liu, Jiahao Li, Zheren Fu, Yi Tu, Jiajun Li, Zhendong Mao, and Yongdong Zhang. 2025a. SparseRM: A lightweight preference modeling with sparse autoencoder. *arXiv preprint arXiv:2511.07896*.

Yan Liu, Renren Jin, Ling Shi, Zheng Yao, and Deyi Xiong. 2025b. FineMath: A fine-grained mathematical evaluation benchmark for chinese large language models. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 24(12):139:1–139:15.

Hantao Lou, Changye Li, Jiaming Ji, and Yaodong Yang. 2025. SAE-V: Interpreting multimodal models for enhanced alignment. In *Forty-second International Conference on Machine Learning*.

Yin Lu, Xuening Zhu, Tong He, and David Wipf. 2025. Sparse autoencoders, again? In *Forty-second International Conference on Machine Learning*.

Jingcheng Niu, Andrew Liu, Zining Zhu, and Gerald Penn. 2024. What does the knowledge neuron thesis have to do with knowledge? In *The Twelfth International Conference on Learning Representations*.

Leiyu Pan, Bojian Xiong, Lei Yang, Renren Jin, Shaowei Zhang, Yue Chen, Ling Shi, Jiang Zhou, Junru Wu, Zhen Wang, Jianxiang Peng, Juesi Xiao, Tianyu Dong, Zhuowen Han, Zhuo Chen, Yuqi Ren, and Deyi Xiong. 2025. Advancing large language models for tibetan with curated data and continual pre-training. *CoRR*, abs/2507.09205.

Jianxiang Peng, Ling Shi, Xinwei Wu, Hanwen Zhang, Fujiang Liu, Haocheng Lyu, and Deyi Xiong. 2025. Diplomacyagent: Do llms balance interests and ethical principles in international events? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 13721–13739. Association for Computational Linguistics.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2024. A practical review of mechanistic interpretability for transformer-based language models. *arXiv preprint arXiv:2407.02646*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, William Saunders, and 10 others. 2025. Open problems in mechanistic interpretability. *Trans. Mach. Learn. Res.*, 2025.

Ling Shi, Yuqin Dai, Ziyin Wang, Ning Gao, Wei Zhang, Chaozheng Wang, Yujie Wang, Wei He, Jinpeng Wang, and Deiyi Xiong. 2026. SAGE: A service agent graph-guided evaluation benchmark. *Preprint*, arXiv:2604.09285.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. 2025. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. *arXiv preprint arXiv:2503.05613*.

Haoran Sun, Renren Jin, Shaoyang Xu, Leiyu Pan, Supryadi, Menglong Cui, Jiangcun Du, Yikun Lei, Lei Yang, Ling Shi, Juesi Xiao, Shaolin Zhu, and Deyi Xiong. 2024. FuxiTranyu: A multilingual large language model trained with balanced data. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: EMNLP 2024 - Industry Track, Miami, Florida, USA, November 12-16, 2024*, pages 1499–1522. Association for Computational Linguistics.

Gemma Team. 2024a. Gemma 2: Improving open language models at a practical size. *CoRR*, abs/2408.00118.

Llama Team. 2024b. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Qwen Team. 2025. Qwen3 technical report. *CoRR*, abs/2505.09388.

A. Templeton, T. Conerly, J. Marcus, J. Lindsey, T. Bricken, B. Chen, A. Pearce, C. Citro, E. Ameisen, A. Jones, H. Cunningham, N. L. Turner, C. McDougall, M. MacDiarmid, C. D. Freeman, T. R. Sumers, E. Rees, J. Batson, A. Jermyn, and 3 others. 2024. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. Transformer Circuits Thread.

Xinwei Wu, Weilong Dong, Shaoyang Xu, and Deyi Xiong. 2024. Mitigating privacy seesaw in large language models: Augmented privacy neuron editing via activation patching. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 5319–5332.

Xinwei Wu, Heng Liu, Xiaohu Zhao, Yuqi Ren, Linlong Xu, Longyue Wang, Deyi Xiong, Weihua Luo, and Kaifu Zhang. 2026. Finding the translation switch: Discovering and exploiting the task-initiation features in llms. In *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 33971–33979. AAAI Press.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024a. LESS: selecting influential data for targeted instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 54104–54132.

Tingyu Xia, Bowen Yu, Kai Dang, An Yang, Yuan Wu, Yuan Tian, Yi Chang, and Junyang Lin. 2024b. Rethinking data selection at scale: Random selection is almost all you need. *arXiv preprint arXiv:2410.09335*.

Lei Yang, Renren Jin, Ling Shi, Jianxiang Peng, Yue Chen, and Deyi Xiong. 2025. ProBench: Benchmarking large language models in competitive programming. *CoRR*, abs/2502.20868.

Lei Yang, Leiyu Pan, Bojian Xiong, Renren Jin, Shaowei Zhang, Yue chen, Jiang Zhou Ling Shi, Junru Wu, Zhen Wang, Jianxiang Peng, Juesi Xiao, Tianyu Dong, Zhuowen Han, Zhuo Chen, Yuqi Ren, and Deyi Xion. 2026. From curated data to scalable models: Continual pre-training of dense and moe large language models for tibetan. In *The 64th Annual Meeting of the Association for Computational Linguistics*.

Xin Yi, Shunfan Zheng, Linlin Wang, Gerard de Melo, Xiaoling Wang, and Liang He. 2025. NLSR: Neuron-level safety realignment of large language models against harmful fine-tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25706–25714.

Mingjia Yin, Chuhan Wu, Yufei Wang, Hao Wang, Wei Guo, Yasheng Wang, Yong Liu, Ruiming Tang, Defu Lian, and Enhong Chen. 2024. Entropy law: The story behind data compression and llm performance. *arXiv e-prints*, pages arXiv–2407.

Zhongshen Zeng, Pengguang Chen, Haiyun Jiang, and Jiaya Jia. 2023. Challenge llms to reason about reasoning: A benchmark to unveil cognitive depth in llms.

Shaowei Zhang and Deyi Xiong. 2025. BackMATH: Towards backward reasoning for solving math problems step by step. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025 - Industry Track, Abu Dhabi, UAE, January 19-24, 2025*, pages 466–482. Association for Computational Linguistics.

Shuyi Zhang, Wei Shi, Sihang Li, Jiayi Liao, Tao Liang, Hengxing Cai, and Xiang Wang. 2025. Interpretable reward model via sparse autoencoder. *arXiv preprint arXiv:2508.08746*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: less is more for alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

A Appendix

A.1 Implementation Details

This section provides additional details regarding our experimental setup to ensure the reproducibility of our work.

All experiments, including the training of our custom SAE for Qwen3 and all model fine-tuning runs, were conducted on a server equipped with 8 NVIDIA H100 GPUs. The implementation was based on the PyTorch framework, leveraging the `sae_lens` and `LLaMA-Factory` libraries for efficient model handling and training. The use of this powerful hardware enabled the comprehensive evaluation of our framework across multiple models, tasks, and baselines.

Model Parameters To ensure the reproducibility of our work, we provide detailed specifications for the Large Language Models (LLMs) and their corresponding Sparse Autoencoders (SAEs) in Table 6. This includes architectural details of the base models and the configuration of the SAEs applied to them.

Supervised Fine-tuning All models were fully fine-tuned using the `LLaMA-Factory` framework on a high-performance computing cluster equipped with 8 NVIDIA H100 GPUs. For each data subset selected by a given method, we trained the corresponding base model for one full epoch. We used a consistent learning rate of 2×10^{-5} and a global batch size of 64 across all experiments. The computational efficiency of our IGDS framework in comparison to these baselines will be analyzed in a subsequent section.

Training SAEs As no pre-trained SAE was publicly available for the `Qwen3-8B` model, we trained the classical SAE to facilitate our analysis. The training was implemented using the `sae_lens` library (v6.6.0), employing a JumpReLU architecture on the residual stream outputs of layers 12 through 18. The dictionary size was set to 65,536, corresponding to an **16x expansion factor** over the model's hidden size. Key training hyperparameters included a context size of 512 tokens, a batch size of 2048, a learning rate of 5×10^{-5}, and a carefully tuned L1 coefficient for each layer to balance reconstruction loss and sparsity. The process was resource-intensive, requiring approximately **7 days** of computation on a single NVIDIA H100 GPU to train the SAE for one layer.

Table 6: Detailed parameters of the LLMs and their corresponding SAEs. Model names in the header are abbreviated for space.

Parameter	Gemma-2-2B	LLaMA-3.1-8B	Qwen3-8B
LLM Parameters			
Total Layers	26	32	36
Hidden Size (d_{model})	2,304	4,096	4,096
SAE Parameters			
SAE Source	Gemma-scope	Llama-scope	Self-trained
Dict. Size (d_{sae})	18,432	32,768	65,536
Exp. Factor	8x	8x	16x
Applied Layers	[0–25]	[0–31]	[12–18]

Table 7: Detailed ROUGE-1/2/L results for the summarization task across different models.

Model	Method	Rouge-1	Rouge-2	Rouge-L
Gemma-2-2B	Base	0.2203	0.0731	0.1698
	Full	0.4501	0.2038	**0.3630**
	Random	0.4394	0.1954	0.3494
	Loss	0.4480	0.2003	0.3589
	IFD	0.4400	0.1999	0.3565
	ZIP	0.4491	0.1932	0.3607
	IGDS	**0.4522**	**0.2089**	0.3627
Llama-3.1-8B	Base	0.0107	0.0039	0.0084
	Full	0.2600	**0.1230**	0.2145
	Random	0.2542	0.1174	0.2155
	Loss	0.2539	0.1158	0.2148
	IFD	0.2578	0.1166	0.2149
	ZIP	0.2451	0.1080	0.2109
	IGDS	**0.2606**	0.1187	**0.2198**
Qwen3-8B-Base	Base	0.3118	0.1243	0.2515
	Full	0.4892	0.2247	0.3954
	Random	0.4823	0.2185	0.3892
	Loss	0.4741	0.2126	0.3817
	IFD	0.4884	0.2232	0.3939
	ZIP	0.4849	0.2198	0.3916
	IGDS	**0.4901**	**0.2274**	**0.3968**

A.2 Supplementary Results for Full ROUGE-1/2/L Metrics

Table 7 details the ROUGE-1, ROUGE-2, and ROUGE-L metrics for the summarization task, supplementing the main results in Table 2. Across all three models (`Gemma-2-2B`, `Llama-3.1-8B`, and `Qwen3-8B-Base`), our IGDS method consistently outperforms other data selection baselines (Random, Loss, IFD, and ZIP). Notably, IGDS frequently achieves superior performance compared to training on the Full dataset—securing the highest scores on all metrics for Qwen3-8B-Base and leading in most metrics for the other models—demonstrating its effectiveness in selecting high-quality samples for text generation.

Table 8: Performance results of general capabilities on MMLU and TruthfulQA benchmarks.

	Math		Sum		Trans	
	MMLU	TruthfulQA	MMLU	TruthfulQA	MMLU	TruthfulQA
Base	54.12%	36.23%	54.12%	36.23%	54.12%	36.23%
IGDS	53.25%	36.46%	53.18%	33.57%	54.05%	35.88%
Full	53.75%	36.00%	52.67%	34.00%	53.12%	34.35%
Random	53.77%	35.63%	53.23%	31.74%	52.98%	36.12%
ZIP	53.39%	36.41%	51.84%	34.12%	53.67%	33.91%
IFD	53.21%	38.33%	52.45%	35.06%	51.33%	34.72%
Loss	53.84%	37.63%	54.01%	33.44%	52.19%	35.21%

A.3 Prompt Template for Task Feature Identification

Table 9 presents the detailed prompt templates employed across three distinct phases of our experiments: Task Feature Identification (Stage 1), Supervised Fine-tuning, and Evaluation. These templates cover the three downstream tasks: Mathematical Reasoning (Math), Text Summarization (Sum), and Machine Translation (Trans).

In the table, text enclosed in curly braces and formatted in italics (e.g., *{Question}*, *{Solution}*) represents data-specific placeholders. These are replaced by the actual content of the corresponding samples from the dataset during processing.

For the **Task Feature Identification** stage, we specifically mark the position used for feature extraction. The colon symbol highlighted with a yellow background and red font (:) indicates the exact token index where the model's internal hidden states (activations) are computed and extracted. This position serves as the final representation of the input context before the generation of the target sequence begins.

For the **Evaluation** stage, we employ a 4-shot setting for the Math task to ensure stable reasoning performance, while the Summarization and Translation tasks are evaluated in a zero-shot setting to test the model's direct instruction-following capabilities.

A.4 Detailed Topology of Task-Specific Features

While Figure 5 provides a macroscopic view of the structural distribution, illustrating *where* task-specific knowledge is generally concentrated, it is equally crucial to identify *what* these features specifically are to confirm they represent stable, intrinsic model properties. To this end, we visualize the fine-grained topological signatures of the identified **Task-Specific Features** for the Math, Summarization, and Translation tasks in Figure 7.

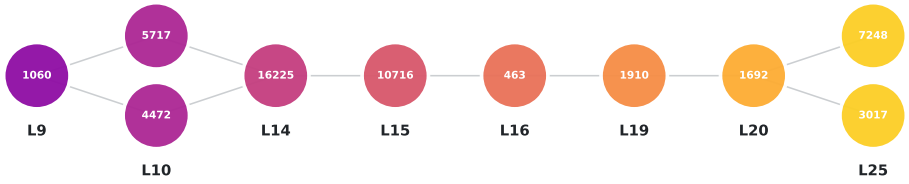
A.5 Evaluation of General Capabilities

To assess the impact of various methods on fundamental performance, we evaluated `Gemma-2-2B` and its different fine-tuning version across MMLU (Hendrycks et al., 2021) and TruthfulQA (Lin et al., 2022) benchmarks (Table 8). The results demonstrate that SFT does not adversely affect general capabilities: across all methods (IGDS, Full, Random, ZIP, IFD, Loss), performance fluctuations remain within a narrow margin, typically between -3% and $+1\%$ relative to the Base model. The models maintain high accuracy in reasoning and truthfulness across Math, Summarization, and Translation tasks, showing no signs of catastrophic forgetting. The result confirms that our fine-tuning strategies successfully preserve the core knowledge and logical proficiency of the foundation models while achieving task alignment.

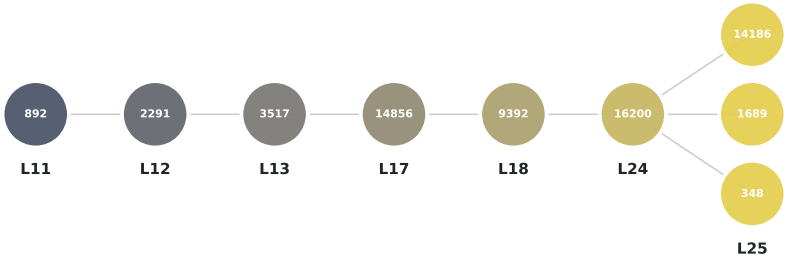
Table 9: Prompt used in Different stages for different tasks.

Stage	Task	Prompt
Task Feature Identification	Math	*{Question}* \n Please reason step by step, and put your final answer within \boxed{}. \n Solution: *{Solution}* \n The final answer is \boxed{*{Golden Answer}*}.
	Sum	Use a sentence to summarize this following text: \n *{Dialogue}* \n Summarization: *{Summary}*
	Trans	Please translate the following text into *{Target_language}*.\n Text:*{Source_text}* \n Translation: *{Target_text}*
Supervised Fine-tuning	Math	Instruction: *{Question}* \n Please reason step by step, and put your final answer within \boxed{}. \n Input: Output: Solution: *{Solution}* \n The final answer is \boxed{*{Golden Answer}*}.
	Sum	Instruction: Use a sentence to summarize this following text. Input: *{Dialogue}* Output: *{Summary}*
	Trans	Instruction: Please translate the following text into *{Target_language}*. Input: *{Source_text}* Output: *{Target_text}*
Evaluation	Math	Please reason step by step, and put your final answer within \boxed{} as the following format. \n Here are some examples: *{4-Shots}* \n ### Problem:\n *{Question}*\n### Solution:
	Sum	Use a sentence to summarize this following text: \n *{Dialogue}* \n Summarization:
	Trans	Please translate the following text into *{Target_language}*.\n Text:*{Source_text}* \n Translation:

(a) Topological signature of features for Math Task, which exhibits a globally distributed topology. Features are activated across the entire depth (L0-L25).



(b) Topological signature of features for Summarization Task, in which features are absent in early layers and emerge primarily from L9 onwards.



(c) Topological signature of features for Translation Task, which shows a sparse, localized activation pattern concentrated in middle (L11-L18) and late layers.

Figure 7: Fine-Grained Topology of Task-Specific Features in Gemma-2-2B. This figure serves as a microscopic supplement to the macroscopic distribution shown in Figure 5.