

ICONS: Influence Consensus for Vision-Language Data Selection

Xindi Wu¹ Mengzhou Xia¹ Rulin Shao² Zhiwei Deng³
 Pang Wei Koh^{2,4} Olga Russakovsky¹
¹Princeton University ²University of Washington ³Google DeepMind ⁴Allen Institute for AI
<https://princetonvisualai.github.io/icons/>

Abstract

Training vision-language models via instruction tuning often relies on large mixtures of data spanning diverse tasks and domains. However, these mixtures frequently include redundant information, increasing computational costs without proportional performance gains—necessitating more effective data selection strategies. Existing methods typically rely on task-agnostic heuristics to estimate data importance or focus on optimizing single tasks in isolation, limiting their effectiveness in multitask settings. In this work, we introduce ICONS, a gradient-based Influence CONsensus approach for vision-language data SElection. Our method leverages first-order training dynamics to estimate the influence of individual training examples on validation performance and aggregates these estimates across tasks via majority voting over task-specific influences. This cross-task consensus identifies data points that are consistently valuable across tasks, enabling us to prioritize examples that drive overall performance. The voting-based design further mitigates issues such as score calibration and outlier sensitivity, resulting in robust and scalable data selection for diverse multitask mixtures. With only 20% of the data from LLAVA-665K and CAMBRIAN-7M, our selected subsets retain 98.6% and 98.8% of the performance achieved with full datasets—and can even surpass full-data training at a 60% selection ratio on LLAVA-665K. Our approach also generalizes to unseen tasks and architectures, demonstrating strong transfer. We release two compact, high-utility subsets—LLAVA-ICONS-133K and CAMBRIAN-ICONS-1.4M—preserving impactful training examples for efficient and scalable vision-language model development.

1 Introduction

Visual instruction tuning is a crucial step in training multimodal language models [25, 26], enabling them to follow language instructions grounded in visual content. Recent approaches rely on large-scale datasets such as LLAVA-665K [25] and CAMBRIAN-7M [42], which contain 665K and 7M examples, respectively. While effective, these datasets introduce significant barriers to iteration and deployment: prolonged training times [3, 17], high storage demands [39, 8], and substantial compute costs [30, 43]. Moreover, not all examples contribute equally to all tasks—naïvely scaling up diverse data mixtures can introduce redundancy and inefficiency. This raises a fundamental question:

Can we identify a compact, multitask-effective subset of training data that preserves model capabilities across tasks while enabling faster experimentation?

Prior work has explored various data selection strategies, including gradient-based approaches [45, 6], influence functions [47, 19], and diversity-based sampling [48, 4]. However, many of these methods either optimize for single tasks in isolation or maximize source diversity without aligning to downstream needs. In multitask visual instruction tuning, this is particularly limiting: optimizing for one task may hurt generalization, and task-agnostic diversity may dilute impact. Rather than selecting

data based on per-task influence, we aim to identify samples that are broadly useful—training examples that consistently contribute across tasks. To do this, we aggregate gradient-based influence scores using a simple yet effective majority voting scheme.

We introduce **ICONS** (Influence **CON**sensus vision-language data **SE**lection), a method that builds upon the gradient-based selection approach LESS [45]. Given access to validation data for each target task, our method: (1) computes first-order gradient influence scores to measure how each training sample impacts task-specific performance, and (2) uses influence consensus through majority voting to identify training samples that show consistent positive value across multiple tasks. This consensus-based mechanism identifies universally valuable training examples: while some samples might be highly influential for individual tasks, we prioritize those that demonstrate broad utility across the task spectrum. While the computational cost of influence estimation is expensive, this front-loaded, one-time investment yields a standardized, compact dataset that can significantly accelerate development of multimodal models, and enables reusable gradient datastores that amortize costs across iterations and deliver long-term savings.

Using ICONS, we create LLAVA-ICONS-133K and CAMBRIAN-ICONS-1.4M, automatically curated 20% subsets of the LLAVA-665K dataset [25] and CAMBRIAN-7M [42] dataset, respectively. These compact datasets maintain 98.6% and 98.8% of their original performance across multiple vision-language tasks, providing significant improvements over randomly selecting same-sized subsets (95.8% and 95.4%) and eliminating approximately two-thirds of the performance drop from shrinking the training data. Moreover, our ICONS outperforms all baselines across different selection ratios, and remarkably achieves above-full-dataset performance, surpassing the original datasets at a 60% selection ratio for LLAVA-665K. Importantly, the selected subset shows strong transferability, e.g., LLAVA-ICONS-133K maintains 95.5-113.9% relative performance across unseen tasks, suggesting that ICONS identifies fundamentally valuable training data. We summarize our key contributions:

1. We propose ICONS, a simple yet effective method for multitask vision-language data selection that identifies broadly valuable training examples via majority voting over task-specific gradient influence scores.
2. Our consensus-based selection outperforms all baselines (§3.2) and we ablate influence aggregation strategies and show the advantage of voting-based consensus (§3.3). We further show that ICONS exceeds 102% of full-dataset performance at a 60% selection ratio on LLAVA-665K (§3.5).
3. We release LLAVA-ICONS-133K and CAMBRIAN-ICONS-1.4M, compact 20% subsets of LLAVA-665K and CAMBRIAN-7M respectively, achieving near-full performance (98.6% and 98.8%), transferring well to unseen tasks (§3.4), and serving as standardized training sets for resource-efficient development.

2 Influence consensus for vision-language data selection

We propose a consensus-driven, gradient-based data selection framework (Fig. 2) for visual instruction tuning datasets. We formalize the problem setup in §2.1 and establish gradient-based influence estimation preliminaries in §2.2. Our two-stage data selection framework consists of: the *specialist* stage (§2.3), which computes task-specific influence scores, and the *generalist* stage (§2.4), which builds cross-task consensus through voting-based aggregation.

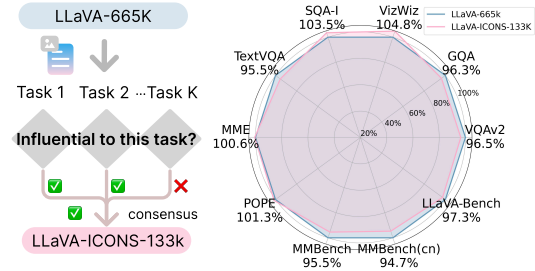


Figure 1: **Influence consensus for vision-language data selection.** (Left) Given a large scale visual instruction tuning dataset (LLAVA-665K), our method uses majority voting across task-specific influence scores to identify training samples that are consistently influential across multiple tasks, forming a compact 20% subset (LLAVA-ICONS-133K) with data points achieving influence consensus. (Right) The radar plot compares performance between LLAVA-665K and our selected subset, showing the selected subset achieves comparable results to the full dataset.

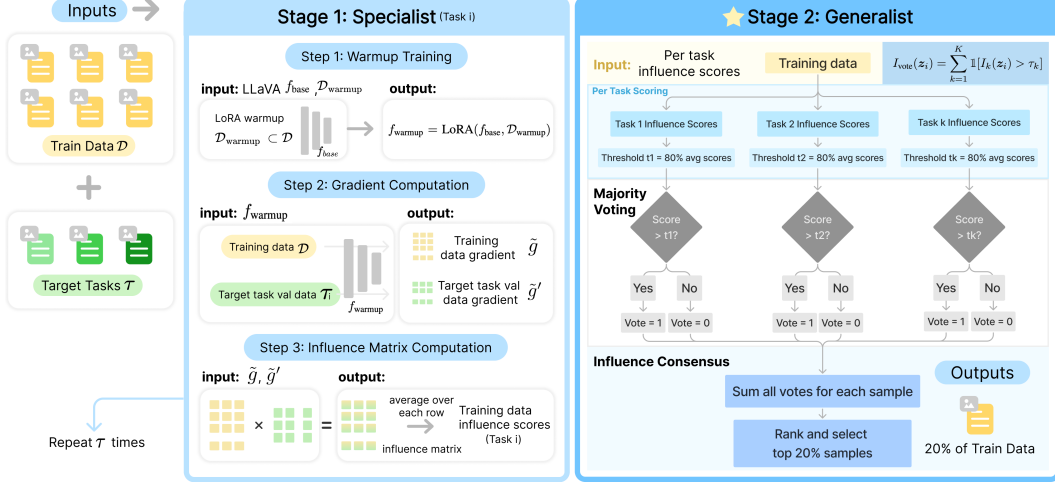


Figure 2: **ICONS**. The Specialist stage (*left*) processes each task individually through three steps: (1) warmup training on a small subset of data, (2) gradient computation for both training and target task validation data, and (3) influence matrix computation to generate per-task influence scores. This process is repeated for each target task. The Generalist stage (*right*) performs **Influence Consensus** to aggregate information across tasks, where samples scoring above the 80th percentile threshold for each task receive a vote. The final selection is made by summing votes across tasks and selecting the top 20% most influential samples, creating a compact yet highly effective training dataset that performs well across all tasks.

2.1 Problem formulation

Given a large-scale visual instruction tuning dataset $\mathcal{D} = \{(\mathbf{I}_i, \mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$ containing N samples, where each data point $\mathbf{z}_i = (\mathbf{I}_i, \mathbf{x}_i, \mathbf{y}_i)$ includes an image \mathbf{I}_i , natural language instruction \mathbf{x}_i , and corresponding target response \mathbf{y}_i ¹, and given access to validation data \mathcal{V}_k for each downstream task $T_k \in \mathcal{T} = \{T_1, \dots, T_K\}$, our goal is to select a compact subset $\mathcal{S} \subset \mathcal{D}$ of size $M \ll N$ that maximizes model performance across multiple downstream tasks:

$$\mathcal{S}^* = \arg \max_{\mathcal{S} \subset \mathcal{D}, |\mathcal{S}|=M} \sum_{k=1}^K \text{Rel}(f_{\mathcal{S}}, T_k), \quad \text{Rel}(f_{\mathcal{S}}, T_k) = \frac{\text{Score}(f_{\mathcal{S}}, T_k)}{\text{Score}(f_{\mathcal{D}}, T_k)}, \quad (1)$$

where $f_{\mathcal{S}}$ and $f_{\mathcal{D}}$ denote models trained on subset \mathcal{S} and full dataset \mathcal{D} , respectively. $\text{Score}(f, T_k)$ is the task-specific evaluation score achieved by model f on task T_k . We define the average relative performance across all tasks as $\text{Rel.} = \sum_{k=1}^K \text{Rel}(f_{\mathcal{S}}, T_k) / K$. Rel. quantifies the subset-trained model’s performance relative to that of the model trained on the entire dataset, with values close to 1 indicating that the subset maintains the performance of full training [21]. Our objective is to select a subset where $\text{Rel.} \approx 1$, i.e., the model trained on the subset achieves comparable performance to the one trained with full dataset.

2.2 Preliminaries

Building on the problem formulation in §2.1, we formalize how to estimate the influence of training samples on downstream task performance. Since our goal is to maximize $\text{Rel}(f_{\mathcal{S}}, T_k)$ across tasks as defined in Eqn. 1, we need an efficient way to estimate how each training sample contributes to the $\text{Score}(f_{\mathcal{S}}, T_k)$ term in the numerator. Denote a training data point as \mathbf{z} and a validation data point as \mathbf{z}' from validation set \mathcal{V}_k for task T_k . Following [37, 45], we estimate how $\mathbf{z} \in \mathcal{D}$ affects validation loss by measuring its gradient alignment with reducing validation loss on \mathcal{V}_k , which directly impacts task-specific evaluation. When training with SGD and batch size 1, using data point \mathbf{z} at timestep t leads to a model update $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \nabla \ell(\mathbf{z}; \boldsymbol{\theta}_t)$, where η_t is the learning rate. To reduce the computational cost, we use the first-order Taylor expansion to estimate the loss on a given validation

¹The framework supports multi-turn conversational data, yet we formalize the problem setup for single-turn instruction-tuning for clarity and simplicity.

data point \mathbf{z}' at time step $t + 1$ as:

$$\ell(\mathbf{z}'; \boldsymbol{\theta}_{t+1}) \approx \ell(\mathbf{z}'; \boldsymbol{\theta}_t) + \langle \nabla \ell(\mathbf{z}'; \boldsymbol{\theta}_t), \boldsymbol{\theta}_{t+1} - \boldsymbol{\theta}_t \rangle.$$

The influence of a training data point \mathbf{z} on a validation data point \mathbf{z}' is:

$$\mathcal{I}_t(\mathbf{z} \rightarrow \mathbf{z}') = \ell(\mathbf{z}'; \boldsymbol{\theta}_{t+1}) - \ell(\mathbf{z}'; \boldsymbol{\theta}_t) \approx -\eta_t \langle \nabla \ell(\mathbf{z}'; \boldsymbol{\theta}_t), \nabla \ell(\mathbf{z}; \boldsymbol{\theta}_t) \rangle,$$

which we refer to as an influence score.

The gradient-based selection approach selects training samples $\{\mathbf{z}\}$ that maximize the gradient inner product $\langle \nabla \ell(\mathbf{z}'; \boldsymbol{\theta}_t), \nabla \ell(\mathbf{z}; \boldsymbol{\theta}_t) \rangle^2$ through a greedy, first-order approximation, which leads to larger reductions in validation loss for point \mathbf{z}' . While it omits second-order terms compared to influence functions [19], it provides an efficient approximation to rank the impact of training samples [45].

2.3 Specialist: individual task influence ranking

To rank the influence of training data for each target task, we compute the influence score of each training data point on a validation set that represents the target task distribution. Following LESS [45], the process involves three steps: (1) training the model on 5% randomly selected data as a lightweight warm-up to initialize visual instruction-following capabilities, (2) computing gradients for training and validation data and compressing the gradients via random projection, and (3) computing the influence score to quantify the impact of each training data on validation set.

Step 1: Warm-up Training. Following LESS [45], we first apply LoRA [12] on a small random subset $\mathcal{D}_{\text{warmup}} \subset \mathcal{D}$ (5%) to obtain $f_{\text{warmup}} = \text{LoRA}(f_{\text{base}}, \mathcal{D}_{\text{warmup}})$. This allows the model to develop basic visual instruction-following capabilities.

Step 2: Gradient computation. For each training data $\mathbf{z}_i \in \mathcal{D}$ and validation data $\mathbf{z}'_j \in \mathcal{D}_{\text{val}}^k$ from \mathcal{T}_k , we compute their gradients with respect to f_{warmup} parameters θ_w :

$$\mathbf{g}_i = \nabla_{\theta_w} \mathcal{L}(f_{\text{warmup}}(\mathbf{z}_i), \mathbf{y}_i), \quad \mathbf{g}'_j = \nabla_{\theta_w} \mathcal{L}(f_{\text{warmup}}(\mathbf{z}'_j), \mathbf{y}'_j)$$

where \mathbf{y}_i and \mathbf{y}'_j are the targets for \mathbf{z}_i and \mathbf{z}'_j , respectively. In order to reduce computational and storage overhead, we apply random projection to the gradient feature: $\tilde{\mathbf{g}}_i = R\mathbf{g}_i$ and $\tilde{\mathbf{g}}'_j = R\mathbf{g}'_j$, where $R \in \mathbb{R}^{d' \times d}$ is a random projection matrix with $d' \ll d$ that preserves inner products with high probability [16]. We further normalize the projected gradients, $\tilde{\mathbf{g}}_i = \frac{\tilde{\mathbf{g}}_i}{\|\tilde{\mathbf{g}}_i\|_2}$, $\tilde{\mathbf{g}}'_j = \frac{\tilde{\mathbf{g}}'_j}{\|\tilde{\mathbf{g}}'_j\|_2}$ to prevent bias from sequence length differences [45].

Step 3: Influence matrix computation. We compute the influence matrix $I \in \mathbb{R}^{|\mathcal{D}| \times |\mathcal{D}_{\text{val}}^k|}$ where each entry $I_{ij} = \langle \tilde{\mathbf{g}}_i, \tilde{\mathbf{g}}'_j \rangle$ is the influence of the training data \mathbf{z}_i on the validation data \mathbf{z}'_j , and then the average influence of training data \mathbf{z}_i on the target task k is calculated as:

$$\bar{I}_k(\mathbf{z}_i) = \frac{1}{|\mathcal{D}_{\text{val}}^k|} \sum_{j=1}^{|\mathcal{D}_{\text{val}}^k|} I_{ij}. \quad (2)$$

This influence estimation process provides a task-specific ranking for the training set \mathcal{D} with respect to task \mathcal{T}_k , where a higher influence score \bar{I}_i suggests a higher influence \mathcal{T}_k .

We can select a small training subset \mathcal{S}_k for a given task k by selecting the training examples \mathbf{z}_i with the highest-influence values $\bar{I}_k(\mathbf{z}_i)$. This simple greedy approach has been shown by LESS to be successful, and thus we use it as our task-specific (“specialist”) baseline. However, recall that our goal is to select a single compact subset that maximizes the performance across *all* tasks. We address this disconnection between task-specific rankings and overall optimization by proposing a voting-based generalist approach to identify the most broadly impactful training data.

2.4 Generalist: cross-task influence consensus

Our goal is to identify a training set subset $\mathcal{S} \subset \mathcal{D}$ of size $M \ll N$ such that its performance across all tasks remains high as defined by Eqn. 1. There are multiple ways to tackle this, depending on the

²In practice, we use cosine similarity instead of direct inner products to avoid biasing selection toward shorter sequences, since gradient norms tend to be inversely correlated with sequence length as noted in [45].

assumptions one makes about the task-specific influence scores $\bar{I}_k(z_i)$. The simplest approach is to merge together all the different tasks’ validation sets $\mathcal{D}_{\text{val}}^k$ (normalizing for their different sizes) and compute the total influence score for a training example z_i as:

$$I_{\text{Merge}}(z_i) = \sum_{k=1}^K \bar{I}_k(z_i). \quad (3)$$

A similar aggregation approach is the one suggested in LESS [45]:

$$I_{\text{Max}}(z_i) = \max_{k=1, \dots, K} \bar{I}_k(z_i), \quad (4)$$

i.e., the influence of the data is measured as its *highest* influence on any tasks. The set of M training examples with the highest aggregated influence scores would be selected for inclusion in the training set $\mathcal{S}_{\text{merge}}$ (correspondingly, \mathcal{S}_{max}). Both approaches, however, require that the influence scores \bar{I}_k be well-calibrated across the different tasks; as we show in §3.3 this may not necessarily be the case.

An alternative approach which does not require directly comparing influence scores \bar{I}_k across tasks k is to leverage the relative rank of the training examples within each task. Concretely, we compute $\text{rank}_k(z_i)$ for each example z_i relative to other examples for task k according to their influence scores (higher influence scores correspond to lower rank). We can have a couple of options. First, we can select the training subset either using the Round Robin (RR) approach [15] where we iterate over tasks and select the lowest-rank example which has not yet been selected to add to our set \mathcal{S}_{RR} . Alternatively, we can select the training subset $\mathcal{S}_{\text{MinRank}}$ such that all the examples within it have a low rank for some task k . Mathematically, albeit somewhat confusingly, this corresponds to:

$$\mathcal{S}_{\text{MinRank}} = \arg \max_{\mathcal{S} \subset \mathcal{D}, |\mathcal{S}|=M} \min_{\substack{\text{task } k \\ \text{example } z_i \notin \mathcal{S}}} \text{rank}_k(z_i), \quad (5)$$

i.e., all examples that are *not* included in \mathcal{S} would have high relative ranks $\text{rank}_k(z_i)$ for all tasks k . However, this approach does not consider the potential interplay between tasks. Recall that in Eqn. 1 we aim to maximize *sum* of the relative performance across all tasks k ; thus, if a training example is beneficial for multiple tasks, we may want to include it even at the expense of a lower-ranked example for a different task k . Thus, we introduce a simple consensus-based voting strategy that identifies training samples that consistently show a high influence score across various tasks. Concretely, we leverage the specialist training sets \mathcal{S}_k as defined in §2.3 consisting of the M highest-influence training examples for each task. We then select a combined training set as follows:

$$\mathcal{S}_{\text{ICONS}} = \arg \max_{\mathcal{S} \subset \mathcal{D}, |\mathcal{S}|=M} \sum_{k=1}^K \sum_{z_i \in \mathcal{S}} \mathbb{1}[z_i \in \mathcal{S}_k] \quad (6)$$

This simple approach offers a key advantage: it does not rely on calibration of influence scores across tasks, and does not make any a-priori assumptions about the relationship between tasks (e.g., that every task needs to have its highest-scoring examples included in the combined training set). Our generalist stage converts each task’s ranked list into a binary vote (“above threshold” or not) and then combines these votes across tasks, eliminating the need for task-specific normalization. As a result, the selection remains insensitive to scale differences while still capturing relative importance within each task. Meanwhile, a training sample is selected only when several tasks independently rank it as influential, preventing over-representation of single-task outliers and ensuring the cross-task utility.

3 Experiments

In this section, we first discuss our experiment setup and evaluation benchmarks (§3.1). We then present our main results by comparing ICONS with the state-of-the-art methods (§3.2), followed by analysis of different selection strategies (§3.3). We further evaluate the transferability of ICONS (§3.4). Lastly, we provide analyze performance trends under different selection ratios (§3.5).

3.1 Evaluation test-bed

Datasets & model. We apply ICONS on major visual instruction tuning (VIT) training datasets: LLaVA-665K [25], CAMBRIAN-7M [42] and VISION-FLAN-186K [46]. The majority of our

Method	VQAv2	GQA	VizWiz	SQA-I	TextVQA	POPE	MME	MMBench en	MMBench cn	LLaVA-W Bench	Rel. (%)
Full	79.1	63.0	47.8	68.4	58.2	86.4	1476.9	66.1	58.9	67.9	100
Random	75.7	58.9	44.3	68.5	55.3	84.7	1483.0	62.2	54.8	65.0	95.8
CLIP-Score [38]	73.4	51.4	43.0	65.0	54.7	85.3	1331.6	55.2	52.0	66.2	91.2
EL2N [36]	76.2	58.7	43.7	65.5	53.0	84.3	1439.5	53.2	47.4	64.9	92.0
Perplexity [31]	75.8	57.0	47.8	65.1	52.8	82.6	1341.4	52.0	45.8	<u>68.3</u>	91.6
SemDeDup [1]	74.2	54.5	46.9	65.8	<u>55.5</u>	84.7	1376.9	52.2	48.5	70.0	92.6
D2-Pruning [29]	73.0	58.4	41.9	<u>69.3</u>	51.8	85.7	1391.2	65.7	57.6	63.9	94.8
Self-Sup [41]	74.9	59.5	46.0	67.8	49.3	83.5	1335.9	61.4	53.8	63.3	93.4
Self-Filter [5]	73.7	58.3	53.2	61.4	52.9	83.8	1306.2	48.8	45.3	64.9	90.9
COINCIDE [21]	76.5	<u>59.8</u>	46.8	69.2	55.6	86.1	1495.6	<u>63.1</u>	54.5	67.3	<u>97.4</u>
RDS [15, 45]	75.1	57.9	48.6	68.0	54.9	<u>86.3</u>	1393.8	61.2	52.7	63.7	95.2
ICONS (ours)	<u>76.3</u>	60.7	<u>50.1</u>	70.8	55.6	87.5	<u>1485.7</u>	<u>63.1</u>	<u>55.8</u>	66.1	98.6

Table 1: **Selection results on LLaVA-665K.** Performance comparison of different data selection approaches when trained on 20% of the LLaVA-665K dataset. The best and second best results for each benchmark are shown in **bold** and underlined, respectively. Our method ICONS achieves the highest overall Rel. (98.6%), consistently outperforming existing approaches including COINCIDE [21] (97.4%) and D2-Pruning [29] (94.8%).

analysis and ablation experiments are conducted on LLaVA-665K. For our experiments, we use the LLaVA-v1.5 model [25] checkpoint after Stage 1 (pre-training for feature alignment) as defined in the original LLaVA training pipeline, with a default size of 7B parameters and LLaVA-665K unless otherwise specified. This checkpoint³ corresponds to the model after training the projector but before any visual instruction tuning in Stage 2. Importantly, this model has not been exposed to the LLaVA-665K VIT dataset prior to the data selection process. In all experiments, we train the models for one epoch following the official finetuning hyperparameters using LoRA. More details on computation, including hardware specifications and runtime are in Appendix A.

Target tasks. We evaluate ICONS across diverse multimodal benchmarks (Appendix C, Tab. 5) that test different capabilities of vision-language models: 1) Multiple-choice understanding: MM-Bench [51] and MME [7]⁴; 2) Visual question answering: VQAv2 [9], GQA [13], and VizWiz [10]; 3) Text understanding in images: TextVQA [40]; 4) Scientific reasoning: ScienceQA [28]; 5) Open-ended generation: LLaVA-W Bench [26]; 6) Factual consistency: POPE [24].

Baselines. We compare our ICONS against several baselines, including random selection, CLIP-Score [38] for measuring image-text alignment, EL2N [36] based on embedding L2 norms, and Perplexity [31] using language model scores. We also compare against SemDeDup [1] for semantic deduplication and D2-Pruning [29] for distribution-aware pruning. Additional baselines include Self-Sup [41] leveraging self-supervised signals, while Self-Filter [5] and COINCIDE [21] are designed for vision-language data selection. We reference LLaVA-665K baseline results from COINCIDE [21]. Additionally, we compare with representation-based data selection baseline (RDS) [15, 45].

3.2 Main results

LLaVA-665K selection. As shown in Tab. 1, ICONS achieves the best overall performance with 98.6% Rel. on LLaVA-665K, outperforming all baselines with LLaVA-ICONS-133K, 20% of the training data. Remarkably, we achieve comparable or better performance than full dataset training on several tasks: SQA-I (70.8 vs. 68.4), MME (1485.7 vs. 1476.9) and POPE (87.5 vs. 86.4). While COINCIDE achieves strong performance (97.4% Rel.), it falls short of ICONS on key tasks. Approaches like EL2N, Perplexity, SemDeDup achieve only 91-92% Rel., showing limitations in preserving performance.

CAMBRIAN-7M & VISION-FLAN-186K selection. We further provide results on VISION-FLAN-186K and CAMBRIAN-7M in Tab. 2. On VISION-FLAN-186K, our method achieves near-full performance (99.8% Rel.) using just 37k samples, significantly outperforming random selection (91.6%) across tasks. Similarly, on CAMBRIAN-7M, ICONS maintains a strong performance (98.8% Rel.) with 1,414k samples, while random selection achieves 95.4%. These results demonstrate

³llava-v1.5-mlp2x-336px-pretrain-vicuna-7b-v1.5, which has no prior exposure to the visual instruction tuning data.

⁴For MME, we focus on its perception section following [21], which evaluates vision capabilities.

Dataset	#Data	Method	VQAv2	GQA	VizWiz	SQA-I	TextVQA	POPE	MME	MMBench	LLaVA-W Bench	Rel. (%)
VISION-FLAN-186K	186k	Full	68.0	49.2	41.7	60.8	50.4	83.4	1,263.2	52.6	45.9	100.0
	37k	Random	64.1	45.8	37.5	58.7	45.3	82.9	1,079.8	46.5	39.6	91.6
		ICONS (ours)	67.4	48.8	44.1	60.2	49.9	83.0	1,252.5	51.9	45.4	99.8
CAMBRIAN-7M	7,068k	Full	80.2	62.9	58.4	75.3	60.9	86.5	1,524.6	69.1	58.9	100.0
	1,414k	Random	74.2	57.5	61.9	71.0	57.1	86.4	1,465.7	63.3	49.6	95.4
		ICONS (ours)	79.6	62.1	60.7	73.9	59.8	86.2	1,503.1	67.8	55.8	98.8

Table 2: **Selection results on VISION-FLAN-186K and CAMBRIAN-7M.** Performance comparison of different data selection approaches when trained on 20% of the VISION-FLAN-186K [46] and CAMBRIAN-7M [42] datasets. ICONS achieves strong performance (99.8% and 98.8% **Rel.** respectively) while using only 20% of the training data, significantly outperforming random selection which is one of the strongest baselines, and approaching full performance.

Aggregation	VQAv2	GQA	VizWiz	SQA-I	TextVQA	POPE	MME	MMB (en)	MMB (cn)	LLaVA-W	Rel. (%)
Full	79.1	63.0	47.8	68.4	58.2	86.4	1476.9	66.1	58.9	67.9	100
Merge	<u>75.7</u>	59.6	47.9	65.5	<u>55.5</u>	86.0	1422.1	59.0	51.0	66.2	96.4
Max	75.2	59.8	48.1	66.2	<u>55.5</u>	85.5	1470.7	58.3	51.8	66.2	96.1
Merge-GausNorm	75.1	<u>60.1</u>	46.4	69.8	54.5	85.6	<u>1482.6</u>	58.9	<u>52.5</u>	66.3	96.8
Merge-SumNorm	75.5	59.1	51.7	68.7	43.5	<u>87.1</u>	1478.3	59.5	50.9	69.8	95.3
Round Robin	75.4	59.1	48.3	<u>70.6</u>	55.2	86.6	1474.5	<u>61.6</u>	51.5	66.9	96.7
MinRank	75.2	59.0	49.7	70.4	55.1	86.9	1456.3	61.1	52.4	<u>68.4</u>	<u>97.1</u>
Vote (ours)	76.3	60.7	<u>50.1</u>	70.8	55.6	87.5	1485.7	63.1	55.8	66.1	98.6

Table 3: **Comparison of aggregation approaches.** Performance of different influence aggregation methods when selecting 20% of the LLaVA-665K dataset. Our proposed aggregation approach (**Vote**) consistently achieves the best overall performance (98.6% Rel.), outperforming both score-based (**Merge**, **Max**), their normalized variants (**Merge-GausNorm**, **Merge-SumNorm**) and rank-based (**Round Robin**, **MinRank**) baselines.

that our approach scales effectively to both small and large datasets, consistently preserving model capabilities while drastically reducing the training data required.

Comparisons with representation-based data selection. We compare our method against **RDS** (Representation-based Data Selection)[15, 45], a strong baseline in language-only instruction tuning. RDS computes training-validation similarity using final-layer representation of the last token in each sequence instead of gradients. For a fair comparison, we use the same influence matrix formulation (Eqn.2) and apply majority voting to reach influence consensus. Our method consistently outperforms RDS across all tasks, particularly those requiring perceptual grounding – e.g., higher scores on GQA (60.7 vs. 57.9), SQA-I (70.8 vs. 68.0), and MME (1485.7 vs. 1393.8). While RDS is effective in selecting large-scale text-only data (e.g. TULU-2/3 [14, 20]), its evaluation has largely focused on language-only tasks, where semantic similarity alone is often sufficient. In contrast, vision-language tasks demand alignment between modalities, where representation similarity is limited as it only reflects current embedding proximity and gradient-based approaches directly estimate each sample’s contribution to the validation loss. Our gradient-based approach directly estimates each sample’s impact on validation loss, capturing cross-modal training dynamics and prioritizing impactful training points. We further provide qualitative comparisons in Appendix §I.

3.3 Analysis of aggregation strategies

Ablations. As introduced in §2.4, we explore multiple strategies for aggregating task-specific influence rankings into a single compact subset. We compare our majority voting approach (**Vote**) with different aggregation approaches for combining task-specific influence scores: 1) score-based methods (**Merge**, **Max**) and their normalized variants (**Merge-GausNorm**, **Merge-SumNorm**), 2) rank-based methods (**Round Robin**, **MinRank**). Our voting-based strategy outperforms all alternatives (Tab. 3), achieving the highest overall Rel. (98.6%). This shows that building a cross-task consensus via majority voting is a simple yet effective strategy for identifying consistently influential examples across tasks, without assuming calibration or comparability of scores.

Limitations of score-based aggregation. Score-based methods like **Merge** (Eqn. 3) and **Max** (Eqn. 4) assume calibrated influence scores across tasks, which is rarely the case. We observe substantial variation in the distribution of influence scores across tasks with standard deviations spanning from 8.15×10^{-3} (MME) to 1.26×10^{-3} (VQAv2), indicating that influence scores for MME

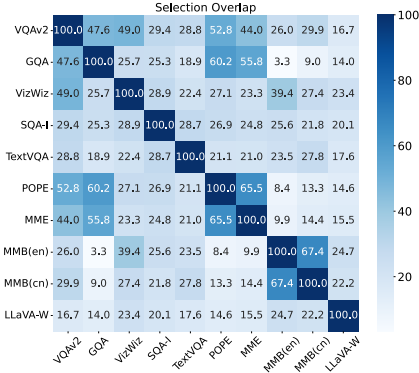


Figure 3: **Pairwise overlap heatmap between specialists.** The values show overlap percentages between benchmarks’ selected samples.

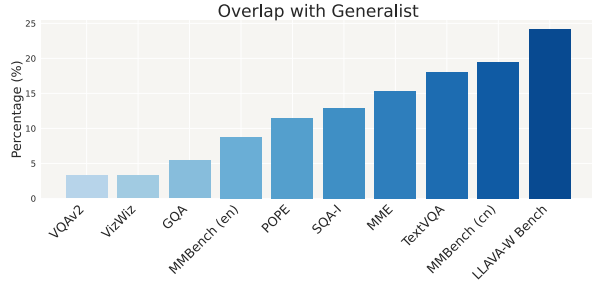


Figure 4: **Data overlap between specialists and generalist selection.** Overlap varies significantly, from 3.27% (VQAv2) to 24.21% (LLaVA-W Bench), reflecting varying alignment between task-specific and consensus selections.

are much more spread out, while those for VQAv2 are tightly concentrated. Similarly, mean influence scores vary in both magnitude and sign: MME has a relatively high positive mean (1.68×10^{-3}), while tasks like POPE (-2.83×10^{-4}) and GQA (-8.89×10^{-5}) have negative means. These divergent patterns show that certain tasks have much wider influence scores distributions, making a sample helpful for one task but neutral or harmful for another. Aggregating raw scores thus biases selection toward tasks with higher variance or skewed means. To address this calibration issue, we experimented with normalization strategies: **Merge-SumNorm**, which rescales each task’s influence scores by dividing them by a task-specific normalization factor (i.e., sum), and **Merge-GausNorm**, which normalizes the scores using task-wise mean and standard deviation before averaging:

$$I_{\text{Merge-SumNorm}}(z_i) = \sum_{k=1}^K \frac{I_k(z_i)}{\sum_j I_k(z_j)} \quad I_{\text{Merge-GausNorm}}(z_i) = \sum_{k=1}^K \frac{I_k(z_i) - \mu_k}{\sigma_k}$$

However, as shown in Tab. 3, both methods still underperform compared to our voting-based strategy, reinforcing the limitations of relying on score magnitudes directly.

Limitations of rank-based aggregation. Rank-based methods sidesteps the challenge of comparing raw influence scores by focusing on within-task ranking. **Round Robin** selects samples by cycling through each task and picking the highest-scoring remaining sample for that task, ensuring balanced coverage. **MinRank** (Eqn. 5) selects samples that have the best minimum rank across all tasks, prioritizing examples that perform exceptionally well in at least one task regardless of their performance in others. Although these methods ensure balanced coverage across tasks, they can overfit to outlier tasks. This is particularly evident with **LLaVA-W Bench** [26], which is an outlier in its influence ranking: both Round Robin and MinRank achieve relatively high scores on it (e.g. MinRank: 68.4), but this comes at the cost of lower performance on all other tasks (Tab. 3). This suggests that purely rank-based selection can trade off some overall efficacy on the mainstream tasks and hurt multi-task balance. In contrast, our **Vote** approach is more robust and avoids this by focusing on multi-task consensus rather than forcing equal representation, yielding better balance and higher overall performance (98.6% Rel.), highlighting the importance of identifying broadly influential examples rather than narrowly optimizing per-task rankings.

Divergent multi-task influence patterns. As shown in Fig. 3, the pairwise overlap heatmap shows notable variation in training data influence across tasks. High overlap – e.g., VQAv2 and VizWiz (49.0%) or POPE and GQA (60.2%), suggests that certain samples are beneficial across similar tasks. However, low overlap, like the 3.3% between MMBench (en) and GQA, highlights that highly influential samples for one task may have limited impact on others. Even closely related tasks, such as MMBench in different languages (English and Chinese), share 67.4% of influential samples. To understand task-specific influence matrices from the specialist stage, we select the top 20% samples per task (Specialists). Overlap with our generalist subset (Fig. 4) varies significantly, from minimal in tasks like VQAv2 (3.27%) and VizWiz (3.28%) to substantial agreement in tasks like LLaVA-W Bench [26] (24.21%). These findings empirically demonstrate significant overlap in influential samples across tasks and validate our approach: by analyzing task-specific gradient-based influence patterns and building consensus across tasks, we can identify a compact subset that captures broadly useful samples across tasks, yielding strong performance with significantly less data.

	AI2D	ChartQA	DocVQA	InfoVQA	MMVet	Naturalbench	RealworldQA	CMMMU	Rel. (%)
Full	55.4	17.5	28.9	26.5	31.1	12.4	52.4	22.1	100.0
Random	50.2	15.1	25.2	24.3	27.6	11.1	49.8	21.9	91.6
LLAVA-ICONS-133K	53.9	17.1	27.9	27.5	29.7	12.8	55.0	25.2	98.7
Per-task Rel. (%)	97.3	97.7	96.5	103.8	95.5	103.2	104.4	114.0	-

Table 4: **Unseen-task generalization.** Performance comparison on unseen benchmarks when trained on selected subsets. Notably, we observe improvements on InfoVQA (103.8%), RealWorldQA (104.4%), and CMMMU (114.0%), highlighting strong generalization to unseen tasks.

3.4 ICONS generalizes to unseen tasks

LLAVA-ICONS-133K demonstrates exceptional generalization on entirely unseen benchmarks that were not used during data selection. As shown in Tab. 4, we test across a diverse spectrum of tasks including MMVet [49], NaturalBench [22], AI2D [18], ChartQA [32], DocVQA [34], InfoVQA [33], RealWorldQA [44] and CMMMU [50]. LLAVA-ICONS-133K achieves 95.5-113.9% (Rel.) compared to full dataset training (InfoVQA: 103.8%, NaturalBench: 105.5%, RealWorldQA: 104.4%, and CMMMU: 113.9%), suggesting that, in some cases, training on LLAVA-ICONS-133K may even outperform training on the full dataset, despite these tasks not being included in the selection process. Importantly, LLAVA-ICONS-133K significantly outperforms random selection across all benchmarks. This suggests that our selection approach successfully captures fundamental visual-language understanding capabilities that transfer well across different task formats and domains. We further provide the cross-architecture generalization results in Appendix §D.

3.5 ICONS outperforms baselines across ratios and exceeds full-data training at 60%

To understand how ICONS scales, we evaluate it across different selection ratios, progressively scaling the subset size from 5% to 60% of LLAVA-665K. As shown in Fig. 5, our results reveal several key patterns: First, ICONS shows particularly strong performance in the low-selection regime (5-20%), where identifying the most influential samples is crucial. Second, as the selection ratio increases, the performance gap between different methods gradually narrows. This convergence pattern is expected, as larger sample sizes naturally capture more of the dataset’s diversity and information. Despite this convergence trend, ICONS consistently outperforms all baselines across all selection ratios. Remarkably, it even surpasses full dataset performance at the 60% ratio, achieving over 102% relative score. One hypothesis is that ICONS can also effectively filter out potentially harmful or noisy training samples that might negatively impact model training, thereby surpassing the full training performance.

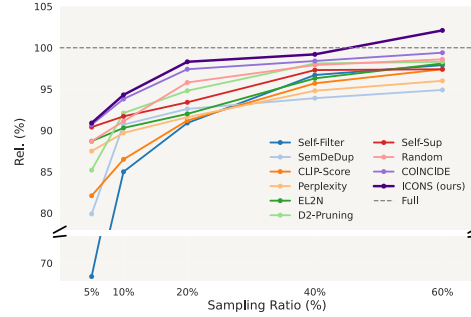


Figure 5: **Different selection ratios.** ICONS consistently outperforms all baselines across different selection ratios and remarkably exceeding 102% at 60% selection ratio.

4 Conclusion

In this work, we introduce ICONS, a simple yet effective influence consensus-based approach for visual instruction tuning data selection. By leveraging gradient-based influence estimation and aggregating task-specific selections through majority voting, our two-stage specialist-to-generalist approach selects training examples that are broadly beneficial across multiple downstream tasks. ICONS addresses limitations of prior selection methods by avoiding assumptions about score comparability across tasks and reducing the sensitivity to outlier task rankings, which can bias selection in both score-based and rank-based approaches. Beyond data selection, it provides a principled way to reason about data influence in multitask mixtures. Through extensive experiments, we show that ICONS builds compact, high-impact datasets without sacrificing performance or generalization, achieving 98.6% of full dataset performance using only 20% of LLAVA-665K and generalizing well to unseen tasks or architectures. We release LLAVA-ICONS-133K and CAMBRIAN-ICONS-1.4M, 20% subsets of the LLAVA-665K and CAMBRIAN-7M datasets, maintaining strong performance on diverse tasks and transferring well to unseen ones. We hope our work inspires further exploration into data-efficient methods for vision-language models across diverse applications.

References

- [1] Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023.
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.
- [3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- [4] Alexander Bukharin and Tuo Zhao. Data diversity matters for robust instruction tuning. *arXiv preprint arXiv:2311.14736*, 2023.
- [5] Ruibo Chen, Yihan Wu, Lichang Chen, Guodong Liu, Qi He, Tianyi Xiong, Chenxi Liu, Junfeng Guo, and Heng Huang. Your vision-language model itself is a strong filter: Towards high-quality instruction tuning with data selection. *arXiv preprint arXiv:2402.12501*, 2024.
- [6] Zhiwei Deng, Tao Li, and Yang Li. Influential language data selection via gradient trajectory pursuit. *arXiv preprint arXiv:2410.16710*, 2024.
- [7] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Zhenyu Qiu, Wei Lin, Jinrui Yang, Xiawu Zheng, et al. Mme: a comprehensive evaluation benchmark for multimodal large language models. *corr abs/2306.13394* (2023), 2023.
- [8] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruva Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36, 2024.
- [9] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.
- [10] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617, 2018.
- [11] Zayd Hammoudeh and Daniel Lowd. Training data influence analysis and estimation: A survey. *Machine Learning*, 113(5):2351–2403, 2024.
- [12] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.
- [13] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.
- [14] Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*, 2023.
- [15] Hamish Ivison, Muru Zhang, Faeze Brahman, Pang Wei Koh, and Pradeep Dasigi. Large-scale data selection for instruction tuning. *arXiv preprint arXiv:2503.01807*, 2025.
- [16] William B Johnson. Extensions of lipshitz mapping into hilbert space. In *Conference modern analysis and probability, 1984*, pages 189–206, 1984.
- [17] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.
- [18] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 235–251. Springer, 2016.
- [19] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR, 2017.

- [20] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.
- [21] Jaewoo Lee, Boyang Li, and Sung Ju Hwang. Concept-skill transferability-based data selection for large vision-language models. *arXiv preprint arXiv:2406.10995*, 2024.
- [22] Baiqi Li, Zhiqiu Lin, Wenxuan Peng, Jean de Dieu Nyandwi, Daniel Jiang, Zixian Ma, Simran Khanuja, Ranjay Krishna, Graham Neubig, and Deva Ramanan. Naturalbench: Evaluating vision-language models on natural adversarial samples. *arXiv preprint arXiv:2410.14669*, 2024.
- [23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.
- [24] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023.
- [25] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024.
- [26] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.
- [27] Zikang Liu, Kun Zhou, Wayne Xin Zhao, Dawei Gao, Yaliang Li, and Ji-Rong Wen. Less is more: Data value estimation for visual instruction tuning. *arXiv preprint arXiv:2403.09559*, 2024.
- [28] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.
- [29] Adyasha Maharana, Prateek Yadav, and Mohit Bansal. D2 pruning: Message passing for balancing diversity and difficulty in data pruning. *arXiv preprint arXiv:2310.07931*, 2023.
- [30] Pratyush Maini, Skyler Seto, He Bai, David Grangier, Yizhe Zhang, and Navdeep Jaitly. Rephrasing the web: A recipe for compute and data-efficient language modeling. *arXiv preprint arXiv:2401.16380*, 2024.
- [31] Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining llms at scale. *arXiv preprint arXiv:2309.04564*, 2023.
- [32] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*, 2022.
- [33] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706, 2022.
- [34] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021.
- [35] Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltingen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pages 15630–15649. PMLR, 2022.
- [36] Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34:20596–20607, 2021.
- [37] Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33:19920–19930, 2020.
- [38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

- [39] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE, 2020.
- [40] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019.
- [41] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.
- [42] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024.
- [43] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, et al. Sustainable ai: Environmental implications, challenges and opportunities. *Proceedings of Machine Learning and Systems*, 4:795–813, 2022.
- [44] x.ai. Introducing Grok 1.5v: The Latest Advancement in AI, November 2024. [Online; accessed 14-November-2024].
- [45] Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.
- [46] Zhiyang Xu, Chao Feng, Rulin Shao, Trevor Ashby, Ying Shen, Di Jin, Yu Cheng, Qifan Wang, and Lifu Huang. Vision-flan: Scaling human-labeled tasks in visual instruction tuning. *arXiv preprint arXiv:2402.11690*, 2024.
- [47] Shuo Yang, Zeke Xie, Hanyu Peng, Min Xu, Mingming Sun, and Ping Li. Dataset pruning: Reducing training data by examining generalization influence. *arXiv preprint arXiv:2205.09329*, 2022.
- [48] Simon Yu, Liangyu Chen, Sara Ahmadian, and Marzieh Fadaee. Diversify and conquer: Diversity-centric data selection with iterative refinement. *arXiv preprint arXiv:2409.11378*, 2024.
- [49] Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*, 2023.
- [50] Ge Zhang, Xinrun Du, Bei Chen, Yiming Liang, Tongxu Luo, Tianyu Zheng, Kang Zhu, Yuyang Cheng, Chunpu Xu, Shuyue Guo, et al. Cmmu: A chinese massive multi-discipline multimodal understanding benchmark. *arXiv preprint arXiv:2401.11944*, 2024.
- [51] Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, Dahua Lin, Yuan Liu, and Haodong Duan. Mmbench: Is your multi-modal model an all-around player. *arXiv preprint arXiv:2307.06281*, 2, 2023.

Appendices

A	Computational complexity	14
A.1	Complexity analysis	14
A.2	Resource requirements	14
A.3	Discussion on cost-benefit justification	14
B	Related work	14
B.1	Visual instruction tuning	14
B.2	Data selection	14
C	Additional experiment details & ablations	15
C.1	Additional task details	15
C.2	Projection dimension	15
C.3	Warm-up Ratio	15
D	Additional analysis	15
D.1	From specialist to generalist	16
D.2	Visual dependency influence ranking	16
D.3	Cross-architecture generalization	17
D.4	Consistency analysis	17
E	Algorithm details	18
F	Limitations	18
G	Broader impact	19
H	Future work	19
I	Visualizations	19
I.1	Representation-based vs. Gradient-based data selection	19
I.2	Specialists & Generalist	19

A Computational complexity

A.1 Complexity analysis

Computing gradient-based influence requires a non-trivial amount of computational resources. In the specialist stage, the complexity scales with both the dataset size $|\mathcal{D}|$ and the gradient dimension d . This stage consists of three steps. First, the warm-up training has a complexity of $\mathcal{O}(|\mathcal{D}_{\text{warmup}}|)$. Second, the gradient computation stage has a computational complexity of $\mathcal{O}(|\mathcal{D}| + |\mathcal{D}_{\text{val}}|)$ for forward and backward passes, with storage requirements of $\mathcal{O}(|\mathcal{D}| \cdot d + |\mathcal{D}_{\text{val}}| \cdot d)$ for the gradients. Third (and finally), the influence matrix computation requires $\mathcal{O}(|\mathcal{D}| \cdot |\mathcal{D}_{\text{val}}| \cdot d')$ compute cost, where d' is the reduced dimension after projection. The generalist stage, focusing on influence consensus across tasks, has lower computational requirements. It begins with threshold computation, requiring $\mathcal{O}(K \cdot |\mathcal{D}| \log |\mathcal{D}|)$ operations for sorting across K tasks. The voting process then takes $\mathcal{O}(K \cdot |\mathcal{D}|)$ compute, followed by a final selection step with complexity $\mathcal{O}(|\mathcal{D}| \log |\mathcal{D}|)$ for sorting the aggregated votes. Storage requirements for this stage are minimal, primarily for the final selected subset.

A.2 Resource requirements

In practice, for LLaVA-665K training data, the warmup training phase requires 0.75 hours using eight L40 GPUs. We parallelize the gradient computation across 100 A6000 GPUs, taking approximately one hour and requiring 103GB of total storage for the gradients. The influence consensus stage is notably efficient, completing in less than a minute on a single L40 GPU. While these computational demands are substantial, they represent front-loaded, one-time costs that can be used across multiple target tasks and model iterations. This makes our method extendable for new tasks, as the expensive training data gradient computations only need to be performed once.

A.3 Discussion on cost-benefit justification

Although gradient-based data selection is computationally intensive, we argue that the initial cost is justified by three key considerations. First, the computational expense is largely a one-time investment: once gradients are computed, they can be stored in our gradient datastore and reused across multiple model iterations, target tasks, and diverse downstream applications. This reusability becomes especially valuable as the number of target datasets grows, because each new target dataset can leverage existing gradient computations, making the selection increasingly efficient at scale.

Second, our empirical results demonstrate substantial performance benefits. Training on a strategically chosen 60% subset of data not only reduces training time but also surpasses the performance obtained by using the full dataset. This improvement underscores how directing more compute resources toward a carefully selected subset can yield higher returns on a fixed set of data.

Lastly, the initial compute-intensive investment in data selection is amortized across future training iterations and future developers. By leveraging the curated, higher-quality dataset, they can substantially reduce training costs.

B Related work

B.1 Visual instruction tuning

Multimodal large language models (MLLMs), e.g., Flamingo [2], LLaVA [26], BLIP2 [23], and Cambrian [42], enhance the capabilities of large language models (LLMs) on various multimodal tasks. A key component in advancing MLLMs is visual instruction tuning [26], a training process that enables these models to interpret and follow instructions within a vision-language context, transforming them into versatile multimodal assistants. This tuning process not only improves the models' instruction-following abilities but also aligns their outputs more closely with user expectations, thus enhancing their utility in practical applications [26].

B.2 Data selection

Data selection methods [11] can be categorized based on the types of information they utilize for selection. Representation-based approaches [1, 21] leverage neural embeddings to capture data

Task	MME	POPE	SQA-I	MMBench		VQAv2	GQA	VizWiz	TextVQA	LLaVA-W
				en	cn					
$ \mathcal{D}_{\text{val}} $	986	500	424	1,164	1,164	1,000	398	800	84	84
$ \mathcal{D}_{\text{test}} $	2,374	8,910	4,241	1,784	1,784	36,807	12,578	8,000	5,000	84
Task Type	Y/N	Y/N	MCQ	MCQ	MCQ	VQA	VQA	VQA	VQA	VQA

Table 5: **Statistics of Target Tasks.** Our target tasks include diverse benchmarks and answer formats, covering different vision-language capabilities. Task types include Multiple-Choice Questions (MCQ), Visual Question Answering (VQA), and Yes/No Questions (Y/N).

representations. Loss trajectory-based methods [35] prioritize data points that contribute most significantly to reducing generalization error over training. Gradient-based techniques [36, 45, 6] select data based on gradient information. Recent work has explored various approaches to select optimal visual instruction tuning datasets. Concurrent work TIVE [27] employs gradient-based selection to identify representative instances. TIVE assumes that the number of specialist data should be proportional to task difficulty and thus samples specialist data based on an estimation of task difficulty. Our method does not rely on this assumption – we directly select samples that benefit the greatest number of tasks. COINCIDE [21] clusters data based on representations associated with concept-skill compositions. Our work follows targeted instruction tuning selection approach LESS [45] to utilize gradient information to calculate the specialist influence (i.e., the influence on a specific task) and extends it to general scenarios by aggregating information from various tasks and selecting data for multiple downstream tasks via majority voting.

C Additional experiment details & ablations

C.1 Additional task details

Here, we provide further details on the target tasks, as summarized in Tab. 5. These tasks cover a wide range of multimodal benchmarks commonly used, including Yes/No questions (Y/N), multiple-choice understanding questions (MCQ) and visual question answering (VQA).

C.2 Projection dimension

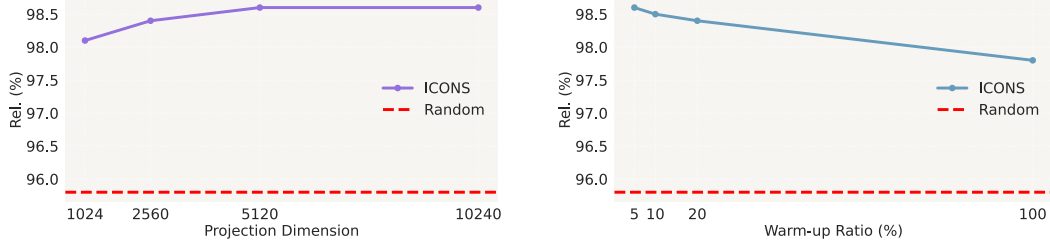
We primarily set the projection dimension to 5120, reducing features from 338.7M to 5120 dimensions. The choice of 5120 was empirically validated for its trade-off between effective capturing gradient representation and maintaining a manageable parameter space. Our LLaVA-v1.5-7b-lora architecture includes a total of 7.4B parameters, with 338.7M parameters being trainable after LoRA adaptation, accounting for approximately 4.58% of the total parameter count. We further ablate different projection dimensions (1024, 2560, 5120, and 10240), with results provided in Fig. 6a.

C.3 Warm-up Ratio

To initiate training, we use 5% of the total training data. We conducted ablation studies to evaluate the impact of varying warm-up ratios (5%, 10%, 20%, and 100%) on selection performance, as shown in Fig. 6b. Our experiments reveal that increasing the warm-up data size does not lead to performance improvements. Surprisingly, models trained with smaller warm-up ratios (5-20%) consistently outperform those trained with the full dataset (100%). Specifically, the 5% warm-up ratio achieves the best performance at 98.6%, while using the complete dataset results in a performance drop to 97.8%. This finding suggests that a small subset of training data is sufficient and even beneficial for model initialization, and potentially gives better signals in the early training stages.

D Additional analysis

In this section, we provide additional analyses to better understand different aspects of our ICONS. We begin by analyzing the effectiveness of task-specific selections and their aggregation into a generalist subset (§D.1). We further explore whether incorporating visual dependency information into the selection process affects performance across different types of vision-language tasks (§D.2).



(a) **Projection Dimension Ablation.** We show the performance of ICONS at different projected dimensions (1024, 2560, 5120, 10240), compared to the random baseline. The performance increases with the projected dimension and reaches a plateau around dimension 5120.

(b) **Warm-up Ratio Ablation.** The blue line represents ICONS performance across different warm-up ratios (5%, 10%, 20%, and 100%), while the red dashed line shows the random baseline performance. Results show that smaller warm-up ratios (5-20%) achieve better performance compared to using the full dataset (100%).

Figure 6: Ablation studies on (left) projection dimension and (right) warm-up ratio.

Method	VQAv2	GQA	VizWiz	SQA-I	TextVQA	POPE	MME	MMBench (en)	MMBench (cn)	LLaVA-W Bench
Full	79.1	63.0	47.8	68.4	58.2	86.4	1476.9	66.1	58.9	67.9
Specialist	77.1	61.1	53.1	69.8	55.7	86.6	1506.1	66.0	56.4	67.1
Generalist	76.3	60.7	50.1	70.8	55.6	87.5	1485.7	63.1	55.8	66.1
Delta (%)	1.04	0.65	5.65	-1.43	0.18	-1.04	1.35	4.34	1.06	1.49

Table 6: **Single-task Selection (Specialist) vs. Consensus-aware Multi-task Selection (Generalist).** The single-task data selection approach selects 20% of LLaVA-665K per task, while our consensus-aware multi-task data selection approach selects a total of 20% data across all tasks.

We also evaluate the transferability of our selected subset across different model scales (§D.3). Additionally, we evaluate the consistency of our method across multiple runs (§D.4), showing its robustness and reliability.

D.1 From specialist to generalist

To understand the intermediate task-specific influence matrices we obtained from the specialist stage, we select 20% of data for each individual task. The task-specific data (Specialists) achieves comparable or superior performance than full data training (Tab. 6). With influence consensus at the generalist stage, we select a 20% subset with only a 1.33% average drop across tasks compared to specialist baselines. This validates our approach: by understanding task-specific influence patterns and building consensus across tasks, we can identify a compact, universal training set that maintains strong performance with significantly less data.

D.2 Visual dependency influence ranking

Recent work [42] has shown that vision-language tasks vary in their reliance on visual information: tasks like MMBench [51] depend heavily on visual grounding, while others like SQA-I [28] can be handled primarily through language, showing only a 5% drop in performance when visual input is removed [42]. To take visual dependency of training data into account, we further explored gradient-based Visual Dependency Score (VDS). For each data point, we calculate the gradient of the model’s auto-regressive cross-entropy loss with both the original image and a Gaussian noise image $I_{\text{noise}} \sim \mathcal{N}(0, 1)$, keeping the text input constant. This quantifies how much the visual component contributes to model performance. We construct an adapted influence matrix: visual influence matrix $\mathcal{I}_{\text{VDS}} \in \mathbb{R}^{|\mathcal{D}| \times |\mathcal{D}_{\text{val}}|}$, which quantifies the visual influence of each training sample \mathbf{z}_i on each validation sample \mathbf{z}'_j w.r.t the model’s gradient alignment and visual dependency. \mathcal{I}_{VDS} is computed as:

$$\mathcal{I}_{\text{VDS},ij} = \langle \nabla_{\theta} \mathcal{L}(\mathbf{z}'_i), \nabla_{\theta} \mathcal{L}(\mathbf{x}_j, I_j) - \nabla_{\theta} \mathcal{L}(\mathbf{x}_j, I_{\text{noise}}) \rangle, \quad (7)$$

where $\nabla_{\theta} \mathcal{L}(\mathbf{x}_j, I_j)$ and $\nabla_{\theta} \mathcal{L}(\mathbf{x}_j, I_{\text{noise}})$ are the gradients computed with the original and Gaussian noise images, respectively. The visual influence matrix \mathcal{I}_{VDS} provides insights into which training

	VQAv2	GQA	VizWiz	SQA-I	TextVQA	POPE	MME	MMBench (en)	MMBench (cn)	LLaVA-W Bench
Full	79.1	63.0	47.8	68.4	58.2	86.4	1476.9	66.1	58.9	67.9
w/o VDS	76.3	60.7	50.1	70.8	55.6	87.5	1485.7	63.1	55.8	66.1
w/ VDS	75.8	60.9	50.3	69.5	54.8	86.8	1489.3	64.3	56.3	67.9
Delta (%)	-0.66	+0.33	+0.40	-1.84	-1.44	-0.80	+0.24	+1.90	+0.90	+2.72

Table 7: **Impact of Visual Dependency Score (VDS) on Selection Performance.** Rows show performance without VDS, with VDS, and the performance change (Delta). VDS improves performance on LLaVA-W Bench (+2.72%), MMBench (en) (+1.90%), and MMBench (cn) (+0.90%), but decreases performance on SQA-I (-1.84%), TextVQA (-1.44%), and POPE (-0.80%).

	VQAv2	GQA	Vizwiz	SQA-I	TextVQA	POPE	MME	MMBench (en)	MMBench (cn)	LLAVA-W	Rel. (%)
Full	80.0	63.3	58.9	71.2	60.2	86.7	1541.7	68.5	61.5	69.5	100.0
Random	77.3	60.7	57.6	69.1	56.8	82.9	1517.2	63.2	56.3	67.5	95.7
7B-selected	78.8	60.4	57.4	70.4	58.3	84.3	1527.5	64.9	59.7	68.2	97.3
13B-selected	78.9	61.2	57.5	71.3	58.4	85.9	1535.2	66.1	59.8	68.8	98.1

Table 8: **Cross-Architecture Generalization.** Our LLaVA-ICONS-133K selected via LLaVA-v1.5-7B model (7B-selected) shows strong cross-architecture transferability, achieving 97.3% Rel., while the data selected via LLaVA-v1.5-13B model (13B-selected) reaches 98.1%, showing that our selected subset generalizes well to different architectures.

samples have the most influence on the validation samples from a visual perspective. This matrix can be used to further rank and select training data that are most impactful for tasks requiring strong visual grounding, ensuring that the selected subset effectively supports vision-dependent performance.

Our empirical results demonstrate that VDS-based data selection has varying effectiveness across different vision-language tasks (Tab. 7). The approach shows substantial improvements on tasks requiring strong visual understanding, such as open-ended generation (LLaVA-W Bench: +2.72%) and multiple-choice understanding (MMBench-EN: +1.90%, MMBench-CN: +0.90%). However, tasks that primarily rely on textual reasoning show decreased performance, including SQA-I (-1.84%) and TextVQA (-1.44%). These results align with and extend the findings in Cambrian [42], demonstrating that the effectiveness of VDS corresponds to a task’s visual dependency - tasks that maintain performance without visual inputs show limited or negative impact from VDS-based selection, while visually-dependent tasks benefit significantly. This pattern suggests that VDS effectively identifies training samples where visual information plays an important role in training.

D.3 Cross-architecture generalization

We further conduct experiments on cross architecture generalization to evaluate the transferability of our selected data across different model scales. While our subset was initially selected using LLaVA-v1.5-7B as the base model, we investigate whether these same examples remain effective for training larger architectures like LLaVA-v1.5-13B. This tests whether our selection criteria identify universally valuable training examples rather than model-specific patterns. Our results in Tab. 8 show cross-architecture generalization, with 13B model trained on 7B-selected subset achieving 98.1% Rel.. Both 7B-selected and 13B-selected subsets outperform random selection (95.7%), with the 13B-selected option showing particular strength in reasoning tasks like MMBench and POPE. This suggests that our selected subset captures fundamental visual-language understanding patterns that generalize well across different model architectures.

D.4 Consistency analysis

To evaluate the consistency of ICONS, we conduct three independent runs of the experiments. As shown in Fig. 7, our method demonstrates high consistency across different runs, achieving $98.6 \pm 1.2\%$ Rel., which shows a notable improvement over random baseline, which achieves $95.8 \pm 2.7\%$. The lower standard deviation (1.2% vs 2.7%) further indicates that our approach produces more stable and reliable outcomes compared to the random baseline.

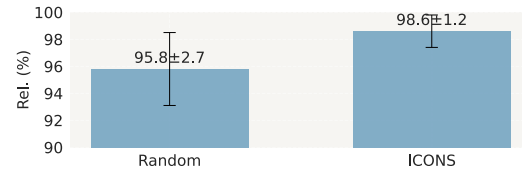


Figure 7: **Rel. (%) Across Runs.** We show the performance across three different runs for random selection and our ICONS.

E Algorithm details

We provide detailed pseudocode for our two-stage ICONS framework. Stage 1 (specialist) computes task-specific influence scores through gradient-based analysis with efficient random projections. Stage 2 (generalist) implements our voting-based consensus mechanism to select samples that are influential across multiple tasks.

Algorithm 1 ICONS Stage 1: Specialist (Task-specific Influence Computation)

Require: Training dataset $D = \{(x_i, I_i, y_i)\}_{i=1}^N$, target tasks $T = \{T_1, \dots, T_K\}$

Require: Warm-up ratio r (default 5%)

Ensure: Task-specific influence scores $\{\bar{I}_k\}_{k=1}^K$

```

1: for each task  $T_k \in T$  do
2:   // Step 1: Warm-up Training
3:   Sample warm-up set  $D_{\text{warmup}} \subset D$  of size  $r|D|$ 
4:    $f_{\text{warmup}} \leftarrow \text{LoRA}(f_{\text{base}}, D_{\text{warmup}})$ 
5:   // Step 2: Gradient Computation
6:   for each training data  $z_i \in D$  do
7:      $g_i \leftarrow \nabla_{\theta_w} L(f_{\text{warmup}}(z_i), y_i)$ 
8:      $\tilde{g}_i \leftarrow \text{Normalize}(Rg_i)$  {Random projection}
9:   for each validation data  $z'_j \in D_{\text{val}}^k$  do
10:     $g'_j \leftarrow \nabla_{\theta_w} L(f_{\text{warmup}}(z'_j), y'_j)$ 
11:     $\tilde{g}'_j \leftarrow \text{Normalize}(Rg'_j)$ 
12:   // Step 3: Influence Matrix Computation
13:   for each  $z_i \in D, z'_j \in D_{\text{val}}^k$  do
14:      $I_{ij}^k \leftarrow \langle \tilde{g}_i, \tilde{g}'_j \rangle$ 
15:   // Compute average influence per training sample
16:    $\bar{I}_k(z_i) \leftarrow \frac{1}{|D_{\text{val}}^k|} \sum_{j=1}^{|D_{\text{val}}^k|} I_{ij}^k$ 
17: return Task-specific influence scores  $\{\bar{I}_k\}_{k=1}^K$ 

```

Algorithm 2 ICONS Stage 2: Generalist (Influence Consensus-based Data Selection)

Require: Task-specific influence scores $\{\bar{I}_k\}_{k=1}^K$

Require: Selection ratio p , number of tasks K

Ensure: Selected subset $S \subset D$ of size $m \ll N$

```

1: // Compute voting thresholds
2: for each task  $T_k \in T$  do
3:    $\tau_k \leftarrow (1 - p)$ -th percentile of  $\{\bar{I}_k(z_i)\}_{i=1}^N$ 
4: // Voting process
5: for each training sample  $z_i \in D$  do
6:    $I_{\text{vote}}(z_i) \leftarrow 0$ 
7:   for each task  $T_k \in T$  do
8:      $\text{vote}_k(z_i) \leftarrow \mathbb{1}[\bar{I}_k(z_i) \geq \tau_k]$ 
9:      $I_{\text{vote}}(z_i) \leftarrow I_{\text{vote}}(z_i) + \text{vote}_k(z_i)$ 
10: // Select top samples based on total votes
11:  $S \leftarrow$  top- $p$  samples by  $I_{\text{vote}}$ 
12: return Selected subset  $S$ 

```

F Limitations

Our approach primarily faces one practical limitation: computing gradients for large training datasets is computationally expensive (Appendix §A). This computational overhead could potentially constrain the method’s applicability when working with extremely large-scale datasets. To support broader research community, we release LLAVA-ICONS-133K dataset to help research iteration and model development under resource-constrained settings.

G Broader impact

Our exploration focuses on scientific understanding and practical applications of vision-language data selection. While our work does not directly imply negative impacts, it may indirectly propagate existing biases present in the original datasets. Therefore, it is important to incorporate rigorous bias-mitigation measurements for data selection. On the positive side, our method enables more efficient and sustainable model development by reducing data redundancy, computational cost while maintaining or even improving performance. Discussion on these critical aspects should remain a priority as we further explore the potential of vision-language data selection.

H Future work

Our work opens several promising research directions for improving vision-language data selection. While our work focuses specifically on visual instruction tuning data, our influence consensus approach can be naturally extended to other stages of MLLM training, such as alignment stage. The majority voting mechanism may under-represent tasks with unique characteristics or those in the long tail, as it prioritizes samples that broadly benefit multiple tasks to build the *main knowledge pool*. This can lead to limited support for specialized tasks or the reinforcement of spurious correlations spanning multiple tasks. Future work could explore **weighted voting mechanisms**, in which tasks are assigned weights based on their relative importance or contribution to overall model performance for more balanced data selection. Additionally, investigating more efficient gradient computation and storage methods would help scale these methods to larger datasets while maintaining strong performance across diverse vision-language tasks.


I Visualizations

I.1 Representation-based vs. Gradient-based data selection

While ICONS leverages gradient-based influence signals, we explore how representation-based data selection (RDS) performs in the same setting (§3.2). We analyze the top-ranked training examples selected by RDS after the generalist stage in Fig. 8 vs. samples selected by ICONS in Fig. 19. Interestingly, we observe that the representation-based variants often favor training examples with repeated images or instructions, which may dominate the learned representations without contributing to better generalization. Some of the highest-scoring samples under representation-based similarity are duplicated image-question pairs with only the answer choices shuffled. We hypothesize that this is a side effect of the way multimodal representations are constructed—where visually dominant or textually redundant samples occupy high-density regions in embedding space. However, these samples do not necessarily translate into broader utility across tasks, as seen in the performance gap in Tab. 1. This discrepancy raises broader questions about what it means for multimodal data to be *diverse*. While we leave these questions open for future exploration, our results suggest that gradient-based influence, though computationally more expensive, is better aligned with generalization and multi-task data mixture settings.


I.2 Specialists & Generalist

We visualize the most influential top three examples across specialists (figs. 9 to 18) and the generalist selection (Fig. 19), along with samples from their corresponding tasks. Notably, the selected high-influence examples by specialists show strong task-specific characteristics both structurally and contextually - they mirror the key attributes of their target tasks in terms of question structure, reasoning patterns, and required visual-language understanding capabilities. Furthermore, the visualization of top influential examples reveals distinct patterns in what makes training samples valuable for different vision-language tasks. VQAv2, GQA, and SQA-I specialists favor multi-turn Q&A scenarios that test both visual comprehension and contextual understanding, while TextVQA, POPE, and MME specialists emphasize text recognition, object verification, and spatial relationships respectively. MMBench-EN and MMBench-CN show consistent patterns despite language differences, focusing on clear, unambiguous scenes that translate well. The LLaVA-W Bench specialist prioritizes examples requiring detailed explanations and multi-step reasoning, and the answers are generally longer. The generalist model values diverse scenarios that combine multiple skills simultaneously. Common




What are the floating blue objects for?
A. swimming
B. boundaries
C. decoration
D. first aid
Answer with the option's letter from the given choices directly

B



What are the floating blue objects for?
A. decoration
B. first aid
C. swimming
D. boundaries
Answer with the option's letter from the given choices directly

D



Please provide a short description for this region: [0.0, 0.33, 0.83, 0.55]

Right side of pizza

Please provide a short description for this region: [0.02, 0.42, 0.48, 0.55].

Piece of pizza closest to brown plate at bottom

Please provide the bounding box coordinate of the region this sentence describes: a margarita pizza

[0.0, 0.33, 0.83, 0.55]

Please provide the bounding box coordinate of the region this sentence describes: far edge of pizza

[0.0, 0.33, 0.83, 0.55]

Please provide the bounding box coordinate of the region this sentence describes: pizza

[0.0, 0.33, 0.83, 0.55]

Please provide a short description for this region: [0.0, 0.33, 0.83, 0.55].


Pizza area on the right side of the plate

Please provide a short description for this region: [0.02, 0.42, 0.48, 0.55].

Closest slice with only a little tomato.

Figure 8: **Top three samples selected by RDS.** These highly-ranked samples are selected due to representation-based similarity but do not necessarily contribute to better generalization, highlighting a key limitation of representation-based selection in multimodal settings.

characteristics that make these examples particularly valuable include multi-turn interactions, clear visual elements, factual and inferential reasoning, cross-modal interaction, and the ability to test multiple capabilities within a single example. This suggests that the most effective training samples are those that combine multiple types of reasoning while maintaining clear, unambiguous ground truth that can be consistently learned across tasks.



Is this a contemporary passenger train?


No

Is this in the country?

Yes

Is the train moving?

No



Is this toaster made out of stainless steel?


Yes

Why would someone leave this toaster on the sidewalk?

Broken

Is the street cracked?

Yes




Which national flag is in the segment screen of this broadcast?

A. netherlands
B. german
C. france
D. uk

Answer with the option's letter from the given choices directly.

D



What does the sofa appear to be blocking?

Window

What is the object in the upper right?

Lamp

What is the cat sitting on?

Couch

What animal is on the couch?

Cat

Is the cat sleeping?

Yes

Do the cushions match the color of the couch?

No

What is the color of the side lamps?

Pink

Does the couch have legs under it?

No

What is on the couch?

Cat

Is there a bed?

No

What's on the sofa?

Cat

Are the cushions golden?

No

How many pillows are on the couch?

2

What kind of store is the person in?

None

Who sleeps on the bed?

Cat

How many pillows in the shot?

2


What color is the pillow on the right?

Blue

What is covering the couch?

Cat

Figure 9: **VQAv2**. Top-left: A sample from VQAv2 [9]. Remaining panels show top three influential samples selected using the specialist influence ranking step.



Is it overcast?

No

Do you see both helmets and skis?

No

Are the snow pants green?

Yes

Is the snow both bright and white?

Yes

Is there either a fence or a snowboard in the photograph?

Yes

What is the person in front of?

Sky

What's the person in front of?

Sky

What is the person doing?

Snowboarding

What is the person in front of the sky doing?

Snowboarding

How hard is the snowboard?

Very

Do the number and the shirt have the same color?

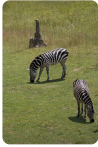
Yes

Does the snowboard look red and hard?

Yes

Are there red snowboards or kites?

Yes



Please provide the bounding box coordinate of the region this sentence describes: a large rock in back.

[0.31, 0.08, 0.49, 0.35]

Please provide the bounding box coordinate of the region this sentence describes: black stripes on zebra fur.

[0.43, 0.37, 0.58, 0.42]

Please provide a short description for this region: [0.33, 0.32, 0.63, 0.58].

A small zebra with big black stripes.

Please provide a short description for this region: [0.33, 0.11, 0.48, 0.34].

Rock structure in grass.

Please provide a short description for this region: [0.67, 0.76, 0.76, 0.84].

Black zebra hooves.

Please provide the bounding box coordinate of the region this sentence describes: the zebra is eating grass.

[0.31, 0.44, 0.42, 0.58]

Please provide the bounding box coordinate of the region this sentence describes: a zebra having a snack.

[0.65, 0.54, 0.81, 0.85]

Please provide a short description for this region: [0.61, 0.77, 0.79, 0.86].


A zebra shadow in grass.

Please provide a short description for this region: [0.31, 0.32, 0.66, 0.58].

An animal standing in place.

Please provide the bounding box coordinate of the region this sentence describes: a striped zebra eating.

[0.33, 0.32, 0.63, 0.57]



Who is the catcher in front of?

Men

Who is in front of the men that are wearing a helmet?

Catcher

Who is wearing a uniform?

Men

Are the men to the left of the catcher wearing a cap?

No

Who is wearing a helmet?

Men

Who wears the face mask?

Men

Who is wearing helmets?

Catcher

What kind of clothing is black?

Uniform

Are the trousers white?

Yes

Who wears a face mask?

Men

Do the number and the shirt have the same color?

Yes

Which place is it?

Stadium

What place was the image taken at?

Stadium

What is the person in front of the men wearing?

Helmets

Does that uniform look blue?

No

What is the catcher wearing?

Helmets

Which color is the shirt?

White

What is the black clothing item called?


Uniform

Is the color of the uniform different than the belt?

No

Who is wearing the uniform?

Men




Which place is it?

Airport

On which side is the man?


Left

Figure 10: **GQA**. Top-left: A sample from GQA [13]. Remaining panels show top three influential samples selected using the specialist influence ranking step.



What is this? and what color is it?
When the provided information is insufficient, respond with 'Unanswerable'.
Answer the question using a single word or phrase.

Black



Would this be a good entree?

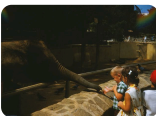
No

What is the white cloth that those snacks are sitting on

Doily

Would this be an appropriate food to serve to an infant?

No



What are the people holding in their hands?

Food

Is this a teddy bear?

No

Is the girl feeding the elephant?

No

Why are these people gathered together?

Zoo

Is this some kind of celebration?

No

Are there flowers on the girl's dress?

Yes

Is this someone's home?

No

Where are the people in the picture?

Zoo

Was this photo taken in a zoo?


Yes

Is the sky clear?

Yes

Is it summertime?

Yes



Is this a man?

No

Is there a parking lot?

Yes


Does this person look upset?

No

Is the lady going up or down the mountain?

Down


Figure 11: **VizWiz**. Top-left: A sample from VizWiz [10]. Remaining panels show top three influential samples selected using the specialist influence ranking step.



Context: Select the best answer. Which property do these three objects have in common?

A. shiny
B. slippery
C. opaque

C



Are there pipes in the bathroom?

Yes

What kind of floor is in the photo?


Black

What is color of the toilet seat?

White

Is the toilet functional?

Yes



How many stools are next to the kitchen counter or island?

There are two stools next to the kitchen counter or island.

What color is the refrigerator in the kitchen?


The refrigerator in the kitchen is white.

What type of material is used for the table and chairs?

The table and chairs are made of wood.

Can you describe the layout of the living space and kitchen?

The living space and kitchen have an open floor plan, which means they are situated next to each other without any wall separations or partitions. This design creates a spacious, airy, and connected environment that allows for easy socialization and movement between the two spaces. In the kitchen, there is a small island with two stools near it, providing a casual seating area for dining or conversation. The wooden table and chairs, along with the white refrigerator, further contribute to the overall aesthetic and layout of the combined living and kitchen area.



What is Santa using?

Laptop

What is the Santa doing?

Typing

Could it be Christmas?

Yes

Is this indoors?

Yes

Figure 12: SQA. Top-left: A sample from SQA [28]. Remaining panels show top three influential samples selected using the specialist influence ranking step.



Figure 13: **TextVQA**. Top-left: A sample from TextVQA [40]. Remaining panels show top three influential samples selected using the specialist influence ranking step.

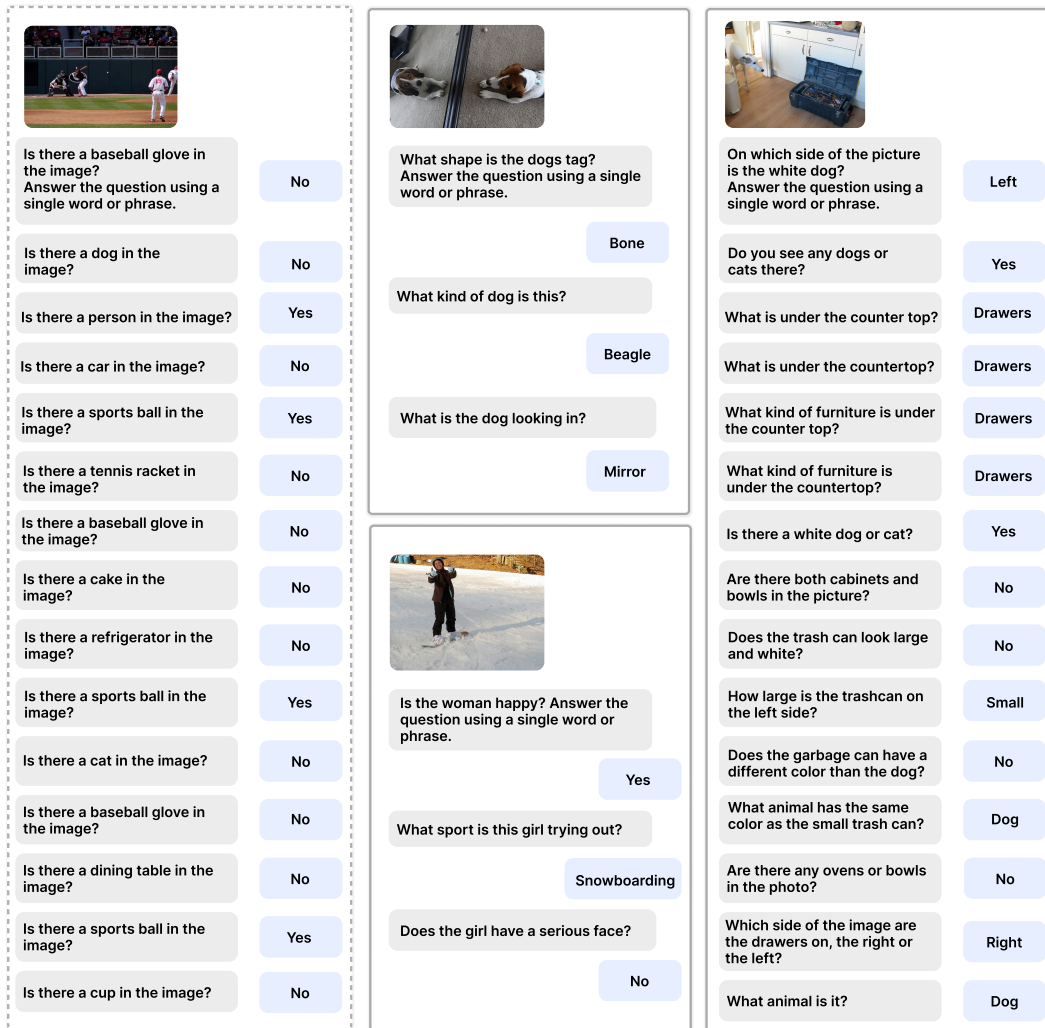


Figure 14: **Pope**. Top-left: A sample from Pope [24]. Remaining panels show top three influential samples selected using the specialist influence ranking step.

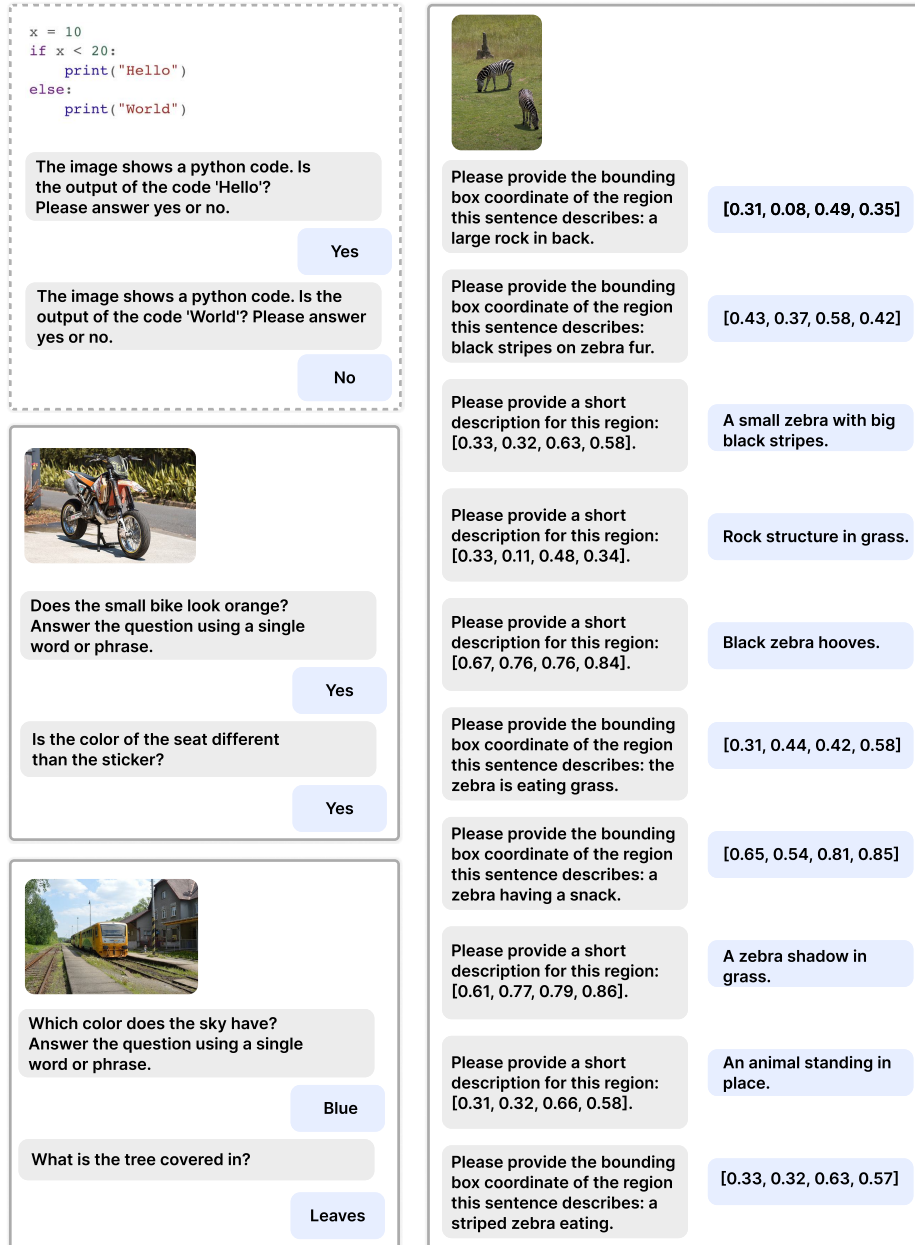

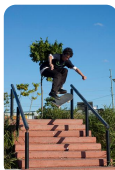


Figure 15: **MME**. Top-left: A sample from MME [7]. Remaining panels show top three influential samples selected using the specialist influence ranking step.




Question: Which of the following was a dependent variable in this experiment?
A: the temperature of the soda
B: the size of the ice pieces

A



Why is the man midair in the middle of the steps?
A. was thrown
B. he fell
C. stood up
D. performing trick
Answer with the option's letter from the given choices directly.

D



Is the machine turned on?
Answer the question using a single word or phrase.


Yes

Where is this machine used?

Kitchen

Can you see through the bottle on the right?

Yes



What type of bread is this?
Answer the question using a single word or phrase.

Banana



What fruit is in the back?

Banana

Is the fruit ripe?

Yes

Figure 16: **MMBench (en)**. Top-left: A sample from MMBench (en) [51]. Remaining panels show top three influential samples selected using the specialist influence ranking step.

ocean water


potato chips

Question: 这两个物体有什么共同的属性?

A: 黄色

B: 咸的

B



Is the window open?

Answer the question using a single word or phrase.

Yes

Could the bed's metal leg be broken?

No

What color are the blinds/shutters?

White

What is seen in the room?


Bed

What kind of tree is closest to the house?

Oak

What is the blue object?

Bed



Is the machine turned on?

Answer the question using a single word or phrase.


Yes

Where is this machine used?

Kitchen

Can you see through the bottle on the right?

Yes



Why is the man midair in the middle of the steps?

A. was thrown

B. he fell

C. stood up

D. performing trick

Answer with the option's letter from the given choices directly.

D

Figure 17: **MMBench (cn)**. Top-left: A sample from MMBench (cn) [51]. Remaining panels show top three influential samples selected using the specialist influence ranking step.



Figure 18: **LLaVA-W Bench**. Top-left: A sample from LLaVA-W Bench [26]. Remaining panels show top three influential samples selected using the specialist influence ranking step.

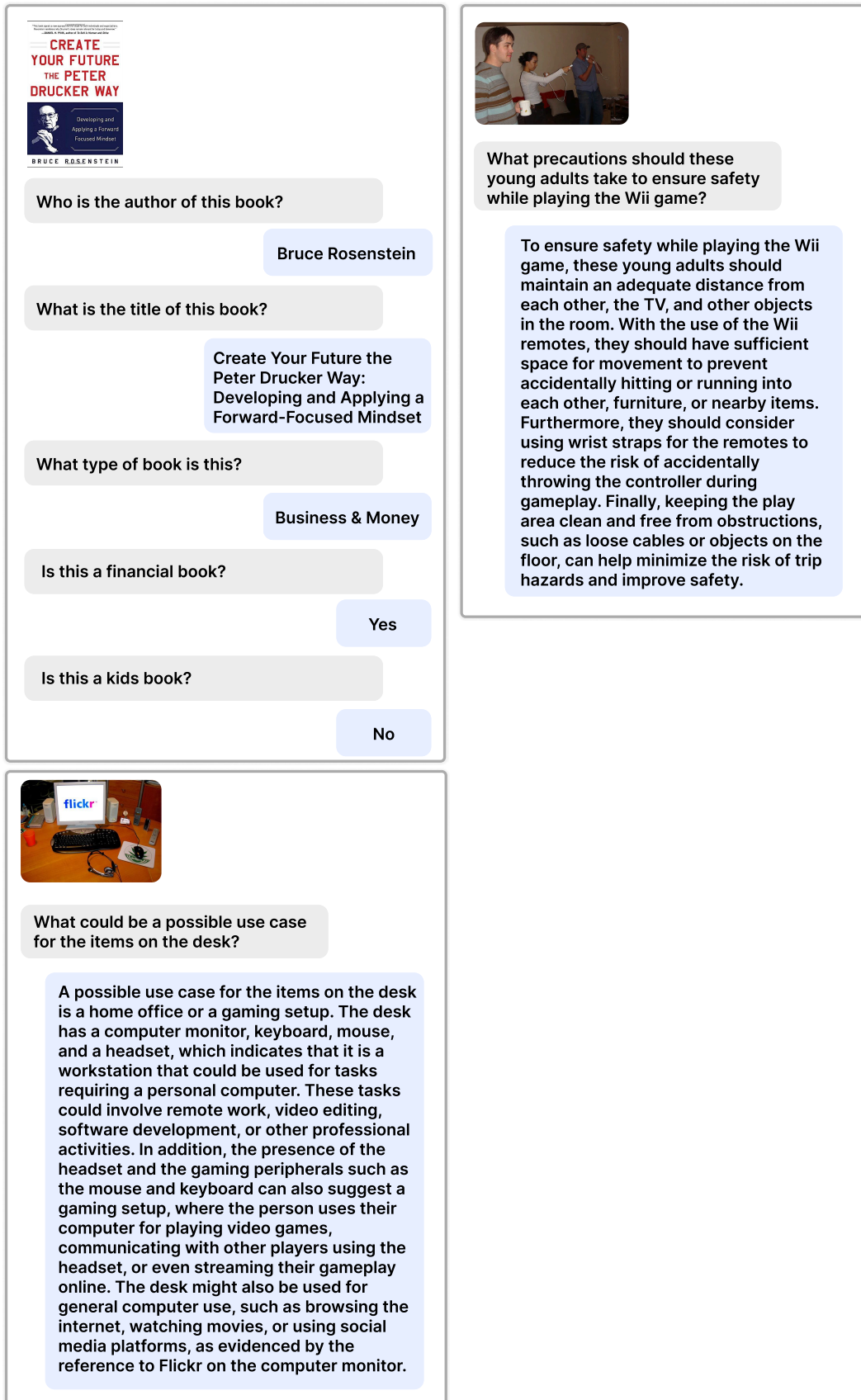


Figure 19: **Generalist**. We show top three influential samples selected after the generalist stage.