OASIS: Online Sample Selection for Continual Instruction Tuning

Anonymous ACL submission

Abstract

In continual instruction tuning (CIT) scenarios, where new instruction tuning data continuously arrive in an online streaming manner, training delays from large-scale data significantly hinder real-time adaptation. Data selection can mitigate this overhead, but existing strategies often rely on pre-trained reference models, which are impractical in CIT setups since future data are unknown. Recent reference model-free online sample selection methods address this, but typically select a fixed number of samples per batch (*e.g.*, top-k), making them vulnerable to distribution shifts where informativeness varies across batches. To address these limitations, we propose OASIS, an adaptive online sample selection approach for CIT that (1) selects informative samples by estimating each sample's informativeness relative to all previously seen data, beyond batch-level constraints, and (2) minimizes informative redundancy of selected samples through iterative selection score updates. Experiments on various large foundation models show that OASIS, using only 25% of the data, achieves comparable performance to full-data training and outperforms the state-of-the-art sampling methods.

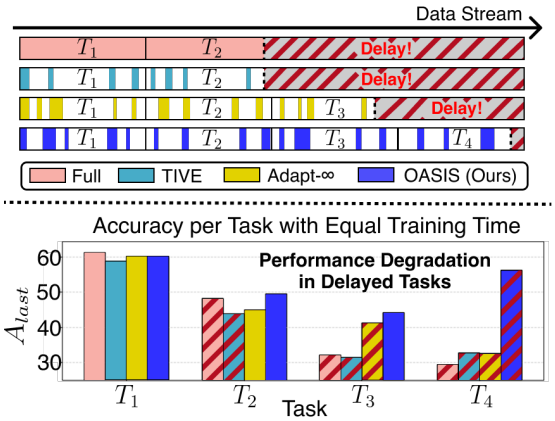

Figure 1: **Real-time adaptation under equal training time.** Bar width indicates training data volume in the CIT data stream. While 'Full' trains on all data, TIVE (Liu et al., 2024b), Adapt-∞ (Maharana et al., 2025), and OASIS use 25% selected data. Under equal training time, 'Full' degrades on newly arrived tasks (*e.g.*, T_3, T_4), since sequential training on all data provides sufficient time for earlier tasks (*e.g.*, T_1, T_2) but insufficient time for new ones. TIVE and Adapt-∞ achieve only marginal speedup despite using 25% data, as the backward-pass selection overhead limits real-time adaptation. OASIS uses inference-only selection with minimal overhead, enabling efficient sample selection and strong adaptation to new tasks.

1 Introduction

A key factor in the success of recent large foundation models (LFMs), including LLMs and multimodal LLMs (MLLMs), is training on large-scale instruction tuning data (Bai et al., 2023b; Chen et al., 2023). While emerging high-quality datasets enable LFMs to better adapt to user preferences and contexts (Maharana et al., 2025; Lau et al., 2024; Zhang, 2024), scaling such datasets also amplifies risks of overfitting and delays in training time (Ma et al., 2023; Liu et al., 2024b).

Unlike conventional training paradigms, where LFMs are optimized on static datasets prior to deployment, real-world applications often require adaptation to continuously arriving data streams (Seo et al., 2024b), *e.g.*, for learning new concepts. This motivates continual instruction tuning (CIT) (Chen et al., 2024; Maharana et al., 2025), where models adapt to shifting data distributions. However, training with the massive influx of continuously arriving data in CIT often causes forgetting of prior knowledge (Zhai et al., 2024; Zhu et al., 2024b), overfitting (He et al., 2025; Ghosh et al., 2024), and significant training delays, which hinder the model's ability to learn from new data promptly (*i.e.*, real-time adaptation) (Koh et al., 2022; Caccia et al., 2022). For example, as shown in Fig. 1, full-data training delays the adaptation to newer tasks (T_3 and T_4) as the model spends excessive time training on earlier tasks (T_1 and T_2).

Data selection methods (Sorscher et al., 2022;

Lee et al., 2024; Abbas et al., 2024) mitigate training delays by selecting informative subsets, but most rely on a reference model trained on the entire data in advance (Mindermann et al., 2022; Shin et al., 2023). Training a reference model is infeasible in CIT, where new data arrive continuously and future data distributions remain unknown. While reference model-free methods (Qin et al., 2024; Hong et al., 2024) eliminate such reliance, their strategy of selecting a fixed number of samples (*e.g.*, top-k) from each streaming batch fails to capture inter-batch and intra-batch informativeness.

Specifically, since informative samples are unevenly distributed over time (Seo et al., 2025), *e.g.*, some batches contain many novel or forgotten samples while others contain few, fixed-size selection per batch often includes uninformative samples while missing critical ones, thereby overlooking inter-batch informativeness variation. Moreover, in continual data streams, similar instances often recur periodically (Koh et al., 2023), *e.g.*, Christmas data in winter, swimsuit data in summer. As a result, many samples within a batch receive similar selection scores (Hong et al., 2024), thus selecting the top-k without considering intra-batch similarity leads to redundancy in the chosen subset.

To address these limitations, we propose **O**nline **A**daptive **S**ample selection via **I**nformative **S**tatistics (**OASIS**), which (i) selects informative samples by estimating each sample's informativeness relative to all previously encountered data, beyond batch-level constraints, and (ii) reduces the redundancy by considering sample-wise similarity within each batch. Specifically, OASIS maintains online estimates of the global mean and variance of informativeness as batches arrive, and uses these statistics to assess the relative informativeness of each sample. Moreover, to avoid selecting similar samples within a batch (Hong et al., 2024), OASIS iteratively updates the samples' selection scores: once a sample is chosen, the scores of remaining candidate samples in the batch are adjusted according to their shared information with the selected sample. Note that OASIS requires only a forward pass for sample selection, roughly half the cost of a backward pass (Huo et al., 2018), enabling efficient selection and reducing training delays, thereby facilitating real-time adaptation to newly encountered tasks, as shown in Fig. 1.

We empirically validate OASIS by comparing it with recent sample selection baselines across various LLMs (*e.g.*, LLama-3.1-8B (Grattafiori et al., 2024) and Qwen3-8B (Yang et al., 2025)) and MLLMs (*e.g.*, LLaVA-1.5-7B (Liu et al., 2024a) and Qwen-VL-2.5-7B (Bai et al., 2025)) on multiple CIT benchmarks. In particular, OASIS incurs only a 1.51% performance degradation compared to full-data training, while training with only 25% of samples on MICVIT benchmark.

We summarize our contributions as follows:

- We propose an adaptive sample selection strategy that selects batch samples based on their informativeness relative to the entire dataset, moving beyond batch-wise selection.
- We propose a redundancy reduction strategy that leverages sample-wise similarity to minimize redundancy among selected samples.
- By combining these two strategies, our proposed OASIS significantly outperforms baselines in CIT through extensive evaluations.

2 Related Work

Continual Instruction Tuning. Existing instruction tuning methods often focus on fixed tasks (Zhu et al., 2024a; Bai et al., 2025), overlooking continuously emerging instruction tuning data (Maharana et al., 2025; Guo et al., 2025). To adapt LFMs to dynamically changing data distributions, *continual instruction tuning* (CIT) has been proposed, which aims to learn new tasks while preserving knowledge from previously encountered instruction tuning data (He et al., 2023). To reflect real-world distribution shifts, various CIT benchmarks have been proposed in both text-only (*e.g.* Long Sequence (Razdaibiedina et al., 2023), TRACE (Wang et al., 2023)) and multi-modal (*e.g.* COAST (Cao et al., 2024), UCIT (Guo et al., 2025)) domains, along with corresponding strategies (*e.g.* Fwd-Prompt (Zheng et al., 2024), SEFE (Chen et al., 2025)). However, they train LFMs on all data, leading to overfitting (Rice et al., 2020; Zhai et al., 2024), high computational costs (Wang et al., 2024; Panos et al., 2025), and training delays (Caccia et al., 2022; Ghunaim et al., 2023). Although aL-SAR (Seo et al., 2025) improves the training efficiency of LFMs through dynamic layer freezing based on batch informativeness, it still relies on the entire dataset, limiting its overall efficiency.

Data Selection. Motivated by the observation that not all data contribute equally to learning, recent work explores selecting informative subsets to match full-data performance with lower training cost (Lee et al., 2024; Abbas et al., 2024;

Qin et al., 2024). Bayesian (Deng et al., 2023b) and RHO-LOSS (Mindermann et al., 2022) enhance training efficiency by training only with a selected subset of data, but both require a reference model pretrained on the full dataset for data selection. TIVE (Liu et al., 2024b) uses smaller reference data, but requires full-layer gradient computation, incurring high computational overhead that offsets any computation savings from data selection. This reliance on reference models limits their use in CIT, where sequentially arriving data makes full-data pretraining infeasible. To address this, reference model-free online sample selection methods like GradNorm (Katharopoulos and Fleuret, 2018), InfoBatch (Qin et al., 2024), and DivBS (Hong et al., 2024) have been proposed. However, they select a fixed number of samples per batch based on difficulty or dissimilarity, limiting adaptability to shifting data distributions, where some batches may contain more informative samples (*e.g.*, newly encountered or forgotten) than others. SelfSup (Sorscher et al., 2022), COINCIDE (Lee et al., 2024), and DBP (Abbas et al., 2024) select samples by K-means clustering with a sensitive hyperparameter K, which is difficult to tune under non-i.i.d. streams due to representation shifts (Książek et al., 2025). Recently, Adapt-∞ (Maharana et al., 2025) tackles data selection under shifting data distributions, but it assumes known task boundaries and relies on costly intermediate gradients, limiting its real-world applicability.

3 Approach

We first present the problem statement for online sample selection in CIT in Sec.3.1. We then introduce our method, **O**nline **A**daptive **S**ample selection via **I**nformative **S**tatistics (**OASIS**), in Sec. 3.2, which consists of two components: **O**nline **R**elative **I**nformativeness **S**election (**ORIS**) in Sec.3.3 and **S**imilarity-aware **I**nformation **R**edundancy **E**limi**N**ation (**SIREN**) in Sec. 3.4.

3.1 Problem Statement of Online Sample Selection in CIT

CIT trains an LFM on a data stream $\mathcal{D}$ comprising a sequence of T tasks, *i.e.*, $\mathcal{D} = \mathcal{D}_1, \dots, \mathcal{D}_T$, where each $\mathcal{D}_i = \{(x_1^i, y_1^i), (x_2^i, y_2^i), \dots\}$ denotes the dataset for task i. Note that explicit task boundaries may be absent, and the data distribution across tasks can even be identical. Reflecting real-world scenarios where data are collected continuously over time (Koh et al., 2022), online CIT assumes that data arrive as a online stream of samples, denoted as $(x_1^i, y_1^i), (x_2^i, y_2^i), \dots$, whereas *offline* CIT provides each task chunk $\mathcal{D}_i$ at once. At timestep t, a batch $\mathcal{B}_t$ with batch size $N_\mathcal{B}$ is drawn from $\mathcal{D}$. A subset $\mathcal{B}_t^* \subset \mathcal{B}_t$ containing N_s samples ($N_s < N_\mathcal{B}$) is then selected according to a predefined selection ratio, and only this selected subset $\mathcal{B}_t^*$ is used for training. Given model parameters θ of an LFM f and loss function $\mathcal{L}$, the objective is:

$$\min_\theta \mathbb{E}_{\mathcal{B}_t \sim \mathcal{D}} \left[\mathcal{L}\big(f_\theta(\mathcal{B}_{t,x}^*), \mathcal{B}_{t,y}^*\big) \right]. \tag{1}$$

3.2 Online Adaptive Sample Selection via Informative Statistics

Existing online sample selection methods typically select a fixed number of samples per batch, *e.g.*, top-N_S selection scores, which struggles under distribution shifts (*e.g.*, CIT), due to: (i) varying numbers of informative samples per batch over time, and (ii) redundancy among co-occurring similar samples within a batch (Hong et al., 2024).

To address these limitations, we propose two strategies: (i) Online Relative Informativeness Selection (ORIS), which selects samples based on their relative informativeness across the entire data distribution, rather than intra-batch ranking; and (ii) Similarity-aware Information Redundancy Elimination (SIREN), which mitigates redundancy by updating, at each selection step, the selection scores of remaining samples within a batch according to their similarity with already selected ones.

Specifically, for each training batch, ORIS computes sample-wise informativeness using Fisher Information (FI) and estimates relative informativeness within the overall FI distribution. During this calculation, SIREN adjusts these scores using sample-wise gradient similarity to account for the impact of training one sample on others. Integrating ORIS and SIREN, we call our method Online Adaptive Sample selection via Informative Statistics (OASIS). We provide an overview of OASIS in Fig. 2, and pseudocode in Sec. A.17.

3.3 Online Relative Informativeness Selection

Informativeness I. To select informative samples, we first define the sample-wise informativeness I based on Fisher Information (FI), which measures a model's potential information gain from each example (Deng et al., 2023a). Computing full FI is costly, since it requires forward and backward passes across all layers, which would

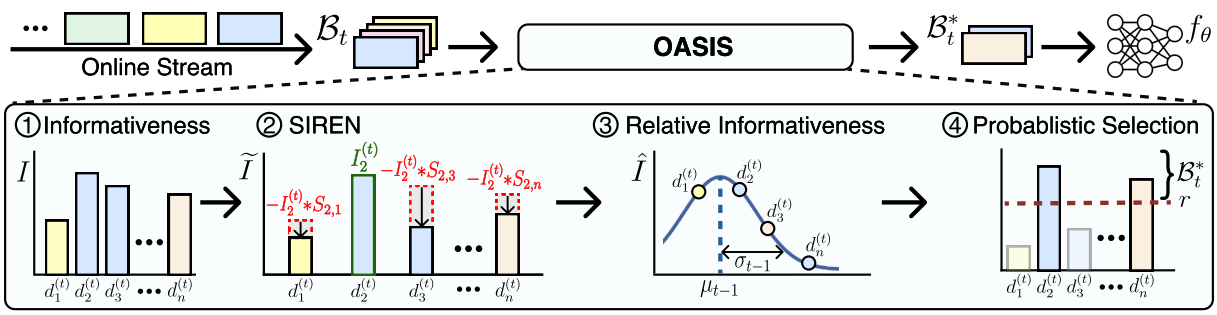

Figure 2: **Overview of our proposed OASIS.** For each online batch $\mathcal{B}_t$: (1) OASIS first scores the informativeness I for all sample in batch $\mathcal{B}_t = \{d_1^{(t)}, d_2^{(t)}, \ldots\}$ (Eq. 2); (2) It then iteratively reduces redundancy by adjusting I of other samples to $\widetilde{I}$ based on their gradient similarity $S_{i,j}$ to the most informative sample (here, $d_2^{(t)}$). (Sec. 3.4); (3) OASIS computes relative informativeness $\hat{I}$ by normalizing the updated informativeness $\widetilde{I}$ using EMA μ_t and EMV σ_t (Eq. 4); (4) Finally, OASIS computes selection probability P_S and selects samples exceeding a uniformly drawn threshold r, resulting in a selected subset $\mathcal{B}_t^* \subset \mathcal{B}_t$ (Eq. 6). Model f_θ is then trained using only $\mathcal{B}_t^*$.

negate the efficiency gains of training on only a selected subset. We therefore adopt an efficient approximation: (i) computing gradients only at the last layer θ_L, since gradients of preceding layers are proportional to it by the chain rule (Koh et al., 2023), and (ii) using a first-order diagonal FI approximation that avoids Hessian computation (Kirkpatrick et al., 2017; Soen and Sun, 2021). Formally, we define the informativeness of the i_{th} sample $d_i^{(t)} = (x_i^{(t)}, y_i^{(t)})$ in batch $\mathcal{B}_t$ as:

$$I_i^{(t)} = \operatorname{tr}\left[\left(\nabla_{\theta_L} L(d_i^{(t)})\right) \cdot \left(\nabla_{\theta_L} L(d_i^{(t)})\right)^\top\right], \quad (2)$$

where $L(d_i^{(t)}) = \ell(f_\theta(x_i^{(t)}), y_i^{(t)})$, ℓ is the loss function, and $\operatorname{tr}(\cdot)$ denotes the trace operator. While existing methods use gradients from middle (Maharana et al., 2025) or all layers (Liu et al., 2024b) for selection, incurring costs close to full-data training, ORIS requires only last-layer gradients, yielding near inference-only cost.

Relative Informativeness $\hat{I}$. However, we cannot compare I across batches, as θ in Eq. 2 changes over time due to model updates at each batch iteration. In other words, cross-batch comparison of I would require re-forwarding all previously encountered samples with the latest θ, which is computationally prohibitive and often infeasible in online CIT scenarios where previous samples may be discarded (Maharana et al., 2025). To avoid these issues, instead of tracking all samples' I, we maintain a running mean of I over seen samples and evaluate each new sample by its deviation from this mean, yielding a relative informativeness over all previously observed data.

To be specific, inspired by the exponential decay model of forgetting (Shin and Lee, 2020; Mahto et al., 2020; Chien et al., 2021), we maintain exponential moving average (EMA) and variance (EMV) of I, which exponentially decay the influence of previous statistics. Formally, at each timestep t, given batch $\mathcal{B}_t$ of size $N_\mathcal{B}$ and decay factor β, we update EMA μ_t and EMV σ_t of I as:

$$\begin{aligned} \mu_t &= \beta \bar{I}^{(t)} + (1-\beta)\mu_{t-1}, \\ \sigma_t &= \beta(\bar{I}^{(t)} - \mu_{t-1})^2 + (1-\beta)\sigma_{t-1}. \end{aligned} \quad (3)$$

where $\bar{I}^{(t)} = \frac{1}{N_\mathcal{B}} \sum_{i=1}^{N_\mathcal{B}} I_i^{(t)}$ denotes average $I^{(t)}$ of samples in $\mathcal{B}_t$.

Using these statistics, we estimate each sample's relative informativeness. By Theorem 3.1, normalization with the EMA μ_{t-1} and EMV σ_{t-1} (updated up to step $t-1$) yields a standard normal distribution, enabling deviation measurement of each sample via Z-score normalization (see Sec. A.1 for proof). Formally, we define relative informativeness of the i_{th} sample with informativeness $I_i^{(t)}$ in batch $\mathcal{B}_t$ at time step t as:

$$\hat{I}_i^{(t)} = \frac{I_i^{(t)} - \mu_{t-1}}{\sigma_{t-1}}. \quad (4)$$

Note that we normalize with μ_{t-1} and σ_{t-1} rather than μ_t and σ_t, since the latter already incorporate $I_i^{(t)}$ and thus induce dependence between numerator and denominator (*i.e.*, self-normalization bias (Jing et al., 2003; de la Peña et al., 2009)).

Theorem 3.1. *Let $I_i^{(t)}$ be defined by Eq. 2 for samples in $\mathcal{B}_t$, and let $(\mu_{t-1}, \sigma_{t-1}^2)$ be the EMA and EMV from Eq. 3 computed from batches prior to t. Assume local stationarity and local weak dependence of $\{I_i^{(t)}\}$ with uniformly bounded second*

moments. Then

$$\hat{I}_i^{(t)} = \frac{I_i^{(t)} - \mu_{t-1}}{\sigma_{t-1}} \approx \mathcal{N}(0,1). \quad (5)$$

Probabilistic Selection. We then determine whether to select each sample based on $\hat{I}^{(t)}$. Inspired by coreset methods (Bang et al., 2021; Seo et al., 2024a), which select both informative and easy-to-learn samples, we adopt probabilistic selection based on $\hat{I}$ rather than deterministically selecting those above a fixed threshold I_T (set by the target selection ratio). Specifically, we apply Sigmoid to $\hat{I} - I_T$, and perform Bernoulli sampling, allowing samples below the threshold to be included with nonzero probability. Formally, with $r \sim \mathcal{U}(0,1)$, where $\mathcal{U}(0,1)$ denotes the uniform distribution over the interval $[0,1]$, we define the selected samples set $\mathcal{B}_t^*$ from batch $\mathcal{B}_t$ as:

$$\mathcal{B}_t^* = \{d_i^{(t)} \in \mathcal{B}_t \mid \text{sigmoid}(\hat{I}_i^{(t)} - I_T) > r\}. \quad (6)$$

Note that existing online selection methods require knowledge of the total sample count to fix per-batch quotas. In contrast, ORIS uses only the selection ratio to determine I_T, enabling operation on endless real-world data streams. We provide details on determining I_T in Sec. A.7.

3.4 Similarity-Aware Information Redundancy Elimination

In real-world data streams, similar task samples often arrive consecutively. Selecting samples within $\mathcal{B}_t$ using ORIS without considering intra-batch similarity can thus lead to redundancy. To mitigate this, we adjust I based on sample-wise similarity. Ideally, one would iteratively select the most informative unchosen sample, train the model, and then recompute I for the remaining candidates using the updated model, thereby accounting for the reduced informativeness of samples similar to those already trained on (Hekimoglu et al., 2023). However, this procedure incurs prohibitive computational cost due to repeated I computation, rendering batch-wise training impractical.

Instead of re-forwarding to recompute I, we approximate its change induced by selecting other samples. Inspired by Du et al. (2018); Seo et al. (2025), which show that gradient alignment correlates with informational overlap, when a sample is selected, we multiply its I by its gradient similarity with each remaining sample and subtract the result from their I values, thereby reducing the informativeness of samples that overlap with the selected one. Formally, given a batch $\mathcal{B}_t = \{d_i^{(t)}\}_{i=1}^{N_\mathcal{B}}$, we first calculate I for each sample and sort them in descending order, *i.e.*, $I_1^{(t)} > \cdots > I_{N_\mathcal{B}}^{(t)}$. We then add the most informative sample (*i.e.*, d_1) to the set H, the set of samples that is assumed to have been trained on. For each remaining sample $d_i^{(t)} \in \mathcal{B}_t \setminus H$, we define updated informativeness $\widetilde{I}_i^{(t)}$ as:

$$\widetilde{I}_i^{(t)} = I_i^{(t)} - \sum_{h \in H} \cos(g_i, g_h) \cdot I_h, \quad (7)$$

where $g_i = \nabla \ell_{\theta_L}(f(x_i), y_i)$ refers to the last-layer gradient of sample $d_i = (x_i, y_i)$. We iteratively add $d_1, \cdots, d_{N_\mathcal{B}}$ to H and repeat the process iteratively, as shown in Algorithm 1.

However, when $|H| > 1$, overlapping similarities between samples in H cause duplicate subtractions. We resolve this using the inclusion-exclusion principle to capture higher-order redundancy as:

$$\widetilde{I}_i^{(t)} = I_i^{(t)} - \sum_{h \in H} \cos(g_i, g_h) \cdot I_h + \sum_{\substack{U \subseteq H \\ |U| \geq 2}} (-1)^{|U|} \cos(g_i, \bar{g}_U) \cdot \bar{I}_U, \quad (8)$$

where U denotes all possible non-empty subsets of H with $|U| \geq 2$, $\bar{g}_U = \frac{1}{|U|}\sum_{u \in U} g_u$ and $\bar{I}_U = \frac{1}{|U|}\sum_{u \in U} I_u$ denote the average gradient and average I over subset $U \subseteq H$, respectively.

We use $\widetilde{I}^{(t)}$ to calculate relative informativeness $\hat{I}^{(t)}$ instead of directly using $I^{(t)}$, as shown in Fig. 2. By applying SIREN before computing $\hat{I}^{(t)}$ in ORIS, we effectively reduce redundancy without requiring any re-forwarding process.

4 Experiments

4.1 Experimental Setup

In all experiments, we report the mean and standard deviation of the results from three different seeds. Moreover, we conduct Welch's t-test with a significance level of 0.05. The highest performance is highlighted in bold, and results not significantly different from the best are <u>underlined</u>.

Models. For MLLMs, we use LLaVA-1.5 (Liu et al., 2024a) and Qwen-VL-2.5 (Bai et al., 2025) as our MLLMs. In the main paper, we focus on LLaVA-1.5-7B and Qwen-VL-2.5-7B, while we provide experiments with other model sizes of LLaVAs (1B, 3B, and 13B) and Qwen-VLs (0.5B),

in Sec. A.8. During training, we update only the LoRA adapters (Hu et al., 2022), keeping the LLM frozen for training efficiency (Ye et al., 2023). For LLMs, we use Llama-3.1-8B (Grattafiori et al., 2024) and Qwen3-8B (Yang et al., 2025).

Baselines. We compare OASIS with recent state-of-the-art sample selection methods, including GradNorm (Katharopoulos and Fleuret, 2018), Self-Sup (Sorscher et al., 2022), COINCIDE (Lee et al., 2024), DBP (Abbas et al., 2024), InfoBatch (Qin et al., 2024), DivBS (Hong et al., 2024), TIVE (Liu et al., 2024b), Adapt-∞ (Maharana et al., 2025), and Random selection, which, despite its simplicity, often surpasses SOTA selection methods as noted in Gupta et al. (2023); Zheng et al. (2023); Maharana et al. (2025). While baselines select a fixed number of samples per batch, OASIS dynamically selects samples probabilistically. For fair comparisons, we ensure OASIS uses a comparable or fewer total samples (see Sec. A.16. See Sec. A.5 for the baselines' details.

Metrics. We report A_{last}, the accuracy measured at the end of training, and A_{avg}, the average accuracy measured at each task boundary.

Benchmarks. We evaluate on a range of CIT benchmarks, including text-only benchmarks (Long Sequence (Razdaibiedina et al., 2023), TRACE (Wang et al., 2023)) and multimodal benchmarks (COAST (Cao et al., 2024), Adapt (Maharana et al., 2025)). However, COAST assumes balanced task sizes (*i.e.*, 20k samples per task), which contradicts real-world data imbalance, while Adapt includes datasets containing COCO (Lin et al., 2014) images that overlap with LLaVA's instruction-tuning data. To address these limitations, we introduce **M**ulti-image **I**mbalanced **C**ontinual **V**isual **I**nstruction **T**uning (**MICVIT**), a new benchmark that removes such overlaps while preserving naturally imbalanced task distributions. To better reflect real-world complexity, MICVIT primarily adopts multi-image datasets.

More details on implementation and benchmarks can be found in Sec. A.2 and A.4, respectively.

4.2 Quantitative Analysis

Comparison across Various Sample Selection Ratios. We compare OASIS with the baselines by training LLaVA-1.5-7B on MICVIT under various selection ratios (*i.e.*, 6.25%, 12.5%, and 25.0%). As shown in Tab. 2, OASIS significantly outperforms baselines, with the largest gap at the lowest sample selection ratio (*i.e.*, 6.25%). Since each sample has a greater impact at smaller coreset sizes (Zheng et al., 2023; Jafari et al., 2024), this highlights OASIS 's strength in identifying informative data. Moreover, at the 25.0% ratio, OASIS nearly matches full-data training, showing only a 1–2% drop in A_{last}. Finally, gains in both A_{last} and A_{avg} indicate that OASIS not only boosts final accuracy but also accelerates convergence by prioritizing informative samples.

Methods using K-means clustering (*i.e.*, Self-Sup, COINCIDE, DBP, TIVE, and Adapt-∞) often perform worse than random selection due to imbalanced clustering under imbalanced data distributions, as detailed in Sec. A.15.

Comparison across Various Benchmarks. We also compare OASIS with baselines using a 6.25% selection ratio on existing CIT benchmarks, namely COAST, Adapt, Long Sequence, and TRACE (Tab. 3). As shown in the table, OASIS consistently outperforms the baselines across multiple CIT benchmarks, demonstrating its robustness and generalizability. We provide additional results with various selection ratios in Sec. A.3

Comparison with Qwen-VL. Beyond LLaVA-1.5-7B, we further test OASIS on Qwen-VL-2.5-7B. As shown in Tab. 1, OASIS consistently outperforms baselines, demonstrating its model-agnostic generalizability. See Sec. A.6 for results with different selection ratios and benchmarks on Qwen-VL.

Method	A_{avg} ↑	A_{last} ↑
Full-Data Training	72.31±0.42	78.18±0.75
Random	67.44±0.12	74.12±0.42
GradNorm (ICML 2018)	67.91±0.55	73.39±0.30
Self-Sup (NeurIPS 2022)	64.34±0.49	70.71±0.56
COINCIDE (EMNLP 2024)	65.12±0.38	71.04±0.72
DBP (ICLR 2024)	62.42±0.42	72.24±0.61
InfoBatch (ICLR 2024)	65.69±0.30	73.51±0.45
DivBS (ICML 2024)	66.18±0.74	75.27±0.42
TIVE (arXiv:2403)	64.64±0.59	72.16±0.81
Adapt-∞ (ICLR 2025)	65.95±0.61	72.18±0.16
OASIS (**Ours**)	**70.23±0.27**	**76.41±0.41**

Table 1: **Quantitative comparison with Qwen-VL-2.5-7B on MICVIT at a 25.0% selection ratio.** Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level.

Comparison of Computation Budget. We compare the computational cost of OASIS and baselines, as shown in Fig. 3. OASIS achieves higher performance while requiring fewer FLOPs. See Sec. A.14 for a detailed computational analysis.

Method	Selection Ratio (%)					
	6.25		12.5		25.0	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	71.80±0.44	79.66±0.43	71.80±0.44	79.66±0.43	71.80±0.44	79.66±0.43
Random	61.15±0.34	67.29±0.61	64.50±0.22	71.33±0.47	65.48±0.73	73.84±0.45
GradNorm (ICML 2018)	61.23±0.83	66.33±0.73	63.68±1.39	70.51±0.10	65.81±1.37	72.03±0.94
Self-Sup (NeurIPS 2022)	58.05±0.77	64.61±1.16	62.84±0.73	69.48±0.62	64.39±1.26	71.81±0.13
COINCIDE (EMNLP 2024)	59.28±1.45	64.83±0.45	62.33±0.92	69.50±0.39	64.92±0.81	72.25±1.17
DBP (ICLR 2024)	57.14±0.51	63.24±0.82	60.64±1.06	67.85±1.19	62.14±1.42	70.13±1.39
InfoBatch (ICLR 2024)	60.82±0.75	68.88±1.09	64.87±0.68	73.63±0.28	66.41±1.36	74.70±0.52
DivBS (ICML 2024)	61.07±0.58	69.06±0.54	65.31±0.41	73.76±0.85	64.22±0.71	75.38±1.25
TIVE (arXiv:2403)	58.75±1.20	64.19±1.40	61.27±0.59	68.38±0.41	64.29±1.02	72.76±0.83
Adapt-∞ (ICLR 2025)	59.13±0.32	64.37±0.85	61.53±0.84	69.82±0.99	65.50±0.52	73.57±1.22
OASIS (**Ours**)	**64.39±0.58**	**71.76±0.72**	**66.89±0.68**	**75.60±0.26**	**68.84±0.37**	**77.95±0.93**

Table 2: **Quantitative comparison between sample selection methods on MICVIT benchmark.** LLaVA-1.5-7B is used as the MLLM. Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level. 'Full-Data Training' uses all data without selection.

Method	MLLM Benchmark				LLM Benchmark			
	COAST		Adapt		Long Sequence		TRACE	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	31.56±1.42	39.06±0.55	55.89±0.20	54.26±0.23	85.83±0.48	82.83±0.47	50.84±0.70	57.73±0.54
Random	23.57±0.17	30.80±0.30	47.64±0.21	40.26±0.71	70.90±0.13	68.02±0.18	43.44±0.55	52.27±0.03
GradNorm	21.91±0.47	28.08±0.49	44.92±0.06	36.78±0.52	70.70±0.05	68.36±0.75	41.39±0.86	50.29±0.69
Self-Sup	22.16±0.27	30.09±0.51	43.85±0.75	34.49±1.37	67.88±0.28	64.24±0.43	37.82±0.39	47.23±0.44
COINCIDE	22.56±0.90	29.89±1.42	44.56±0.84	35.83±0.48	69.02±0.59	66.82±0.69	39.14±0.85	49.42±0.82
DBP	20.28±0.81	28.67±0.32	43.25±0.45	34.75±0.54	66.74±0.82	64.95±0.06	35.93±0.26	45.86±0.98
InfoBatch	22.93±0.73	29.14±0.56	47.65±0.30	40.60±0.96	74.38±0.37	69.84±0.24	41.26±0.27	50.19±0.13
DivBS	23.41±0.14	31.72±0.18	47.25±0.74	40.38±0.62	74.37±0.75	70.65±0.57	44.25±0.08	52.23±0.76
TIVE	21.15±0.09	28.12±0.10	44.72±0.58	35.28±1.37	68.15±0.41	65.79±0.90	36.57±0.84	47.04±0.02
Adapt-∞	22.42±0.16	29.39±0.72	45.06±0.34	35.51±1.29	71.48±0.97	68.45±0.31	42.38±0.14	50.48±0.44
OASIS (**Ours**)	**25.67±0.35**	**34.23±0.38**	**49.98±0.27**	**43.94±0.31**	**75.26±0.35**	**71.91±0.55**	**45.56±0.35**	**53.48±0.17**

Table 3: **Quantitative comparison across MLLM and LLM benchmarks at a 6.25% selection ratio.** We employ LLaVA-1.5-7B (MLLM) and LLaMA-3.1-8B (LLM). Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level. 'Full-Data Training' uses all data without selection.

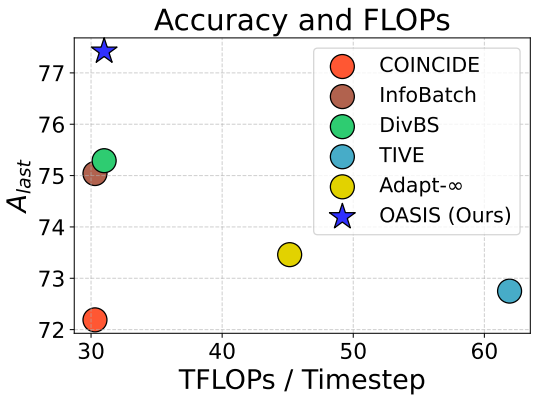

Figure 3: **Accuracy and FLOPs with 25% selection ratio on MICVIT.** The top-left corner illustrates effective and efficient sample selection.

Method	MICVIT	COAST
Random	0.263	0.624
GradNorm (ICML 2018)	0.255	0.598
Self-Sup (NeurIPS 2022)	0.358	0.581
COINCIDE (EMNLP 2024)	0.259	0.604
DBP (ICLR 2024)	0.288	0.587
InfoBatch (ICLR 2024)	0.308	0.632
DivBS (ICML 2024)	0.291	0.625
TIVE (arXiv:2403)	0.296	0.613
Adapt-∞ (ICLR 2025)	0.274	0.578
OASIS w/o SIREN	0.261	0.599
OASIS (Ours)	**0.242**	**0.563**

Table 4: **Density comparison of selected samples.** Lower density indicates higher diversity.

Comparison of Selected Samples' Diversity. We quantify diversity using kernel density, computed as the mean pairwise similarity under a Gaussian kernel, following Lee et al. (2024). As shown in Tab. 4, OASIS selects the most diverse set among all methods. We attribute this to the probabilistic sampling in ORIS, which promotes diversity, and the redundancy reduction in SIREN.

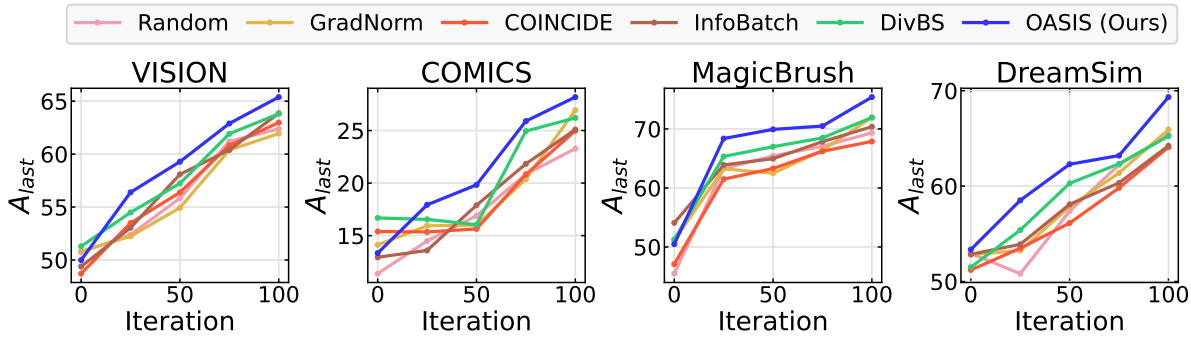

Figure 4: **Comparison of fast adaptation performance.** After CIT of LLaVA-1.5-7B on subsets (25% of the full data), selected using each sample selection baseline from MICVIT, we fine-tune the model for 100 epochs on each downstream task (*i.e.*, VISION, COMICS, MagicBrush, and DreamSim).

Method	MICVIT		COAST		Adapt	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Vanilla	62.54±0.55	67.16±0.39	21.35±0.38	29.39±0.26	46.36±0.53	38.57±0.38
(+) ORIS	65.55±0.41	72.45±0.30	25.80±0.32	35.28±0.52	50.52±0.42	44.65±0.10
(+) ORIS & SIREN (**Ours**)	**67.58±0.46**	**74.51±0.48**	**27.83±0.50**	**37.29±0.13**	**51.33±1.16**	**46.22±0.57**

Table 5: **Ablations for proposed components of OASIS.** We train LLaVA-1.5-7B using 6.25% of the dataset samples with 0.125, 0.25, and 0.25 online iterations in MICVIT, COAST, and Adapt, respectively. Vanilla: selecting a fixed number of samples with the highest FI per batch.

Comparison of Fast Adaptation Performance. We evaluate fast adaptation on unseen downstream tasks following CIT, where CIT serves as upstream continual pre-training. This setup assesses the generalizability of models trained on subsets selected by different online sample selection methods. Downstream tasks include COMICS (Iyyer et al., 2017), VISION (Bai et al., 2023a), DreamSim (Fu et al., 2023), and MagicBrush (Zhang et al., 2023), while MICVIT serve as upstream datasets. Results in Fig. 4 show that subsets selected by OASIS yield consistently superior performance across all downstream tasks, demonstrating superior generalizability. Note that zero-shot performance (*i.e.*, iteration 0) can vary depending on the similarity between the downstream tasks and the subset of data selected by each baseline, resulting in inconsistent ordering between baselines. However, as fine-tuning on downstream tasks progresses, OASIS exhibits consistently faster adaptation, demonstrating the superior generalizability of models trained on subsets selected by OASIS. See Sec. A.9 for additional fast adaptation results across benchmarks.

4.3 Ablation Study

Ablation Study on Different Components. We ablate OASIS to investigate the benefit of each proposed component, and summarize the results in Tab. 5. As shown, ORIS significantly improves performance by identifying informative samples, while SIREN further enhances performance by reducing redundancy between selected samples. Consistent improvements across benchmarks validate the effectiveness of our proposed components.

Additional Analyses. We further evaluate OASIS across different model scales (*e.g.*, 0.5B, 1B, 3B, and 13B) (Sec. A.8) and analyze informativeness metric evaluations (Sec. A.10), effect of task orders (Sec. A.11), effect of EMA ratio β (Sec. A.12), accuracy over time (Sec. A.13), and comparison of the number of selected samples (Sec. A.16).

5 Conclusion

We address the challenge of achieving high performance while training on a small subset of data in the online CIT setup. Prior works select a fixed number of samples per batch, which struggles in the CIT setup due to varying informativeness and frequent redundancy from co-occurring similar samples. To address this, we propose OASIS, comprising two components: ORIS, which selects informative samples by estimating each sample's informativeness relative to all previously encountered data, and SIREN, which reduces redundancy via sample-wise similarity. Extensive experiments on diverse CIT benchmarks demonstrate that our proposed OASIS consistently selects informative samples, outperforms existing baselines, and achieves performance comparable to full-data training.

Limitations. Our method efficiently selects informative samples in CIT, requiring only a forward pass without any backward computation. A promising future direction is to remove this forward-pass requirement entirely, enabling near-instantaneous adaptation and substantially faster processing for real-time CIT applications.

References

Amro Abbas, Evgenia Rusak, Kushal Tirumala, Wieland Brendel, Kamalika Chaudhuri, and Ari S Morcos. 2024. Effective pruning of web-scale datasets based on complexity of concept clusters. *arXiv preprint arXiv:2401.04578*.

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

Haoping Bai, Shancong Mou, Tatiana Likhomanenko, Ramazan Gokberk Cinbis, Oncel Tuzel, Ping Huang, Jiulong Shan, Jianjun Shi, and Meng Cao. 2023a. Vision datasets: A benchmark for vision-based industrial inspection. *arXiv preprint arXiv:2306.07890*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023b. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. 2021. Rainbow memory: Continual learning with a memory of diverse samples. In *CVPR*.

Lucas Caccia, Jing Xu, Myle Ott, Marcaurelio Ranzato, and Ludovic Denoyer. 2022. On anytime learning at macroscale. In *CoLLAs*. PMLR.

Meng Cao, Yuyang Liu, Yingfei Liu, Tiancai Wang, Jiahua Dong, Henghui Ding, Xiangyu Zhang, Ian Reid, and Xiaodan Liang. 2024. Continual llava: Continual instruction tuning in large vision-language models. *arXiv preprint arXiv:2411.02564*.

Cheng Chen, Junchen Zhu, Xu Luo, Hengtao Shen, Jingkuan Song, and Lianli Gao. 2024. Coin: A benchmark of continual instruction tuning for multimodel large language models. In *NeurIPS*.

Jinpeng Chen, Runmin Cong, Yuzhi Zhao, Hongzheng Yang, Guangneng Hu, Horace Ho Shing Ip, and Sam Kwong. 2025. Sefe: Superficial and essential forgetting eliminator for multimodal continual instruction tuning. *arXiv preprint arXiv:2505.02486*.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

Louis HY Chen, Larry Goldstein, and Qi-Man Shao. 2010. *Normal approximation by Stein's method*. Springer Science & Business Media.

Hsiang-Yun Sherry Chien, Javier S Turek, Nicole Beckage, Vy A Vo, Christopher J Honey, and Ted L Willke. 2021. Slower is better: revisiting the forgetting mechanism in lstm for slower information decay. *arXiv preprint arXiv:2105.05944*.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. 2020. Selection via proxy: Efficient data selection for deep learning. In *ICLR*.

Anirban DasGupta. 2008. *Asymptotic theory of statistics and probability*, volume 180. Springer.

Sam Dauncey, Christopher C. Holmes, Christopher Williams, and Fabian Falck. 2024. Approximations to the fisher information metric of deep generative models for out-of-distribution detection. *Transactions on Machine Learning Research*.

Victor H. de la Peña, Tze Leung Lai, and Qi-Man Shao. 2009. *Self-Normalized Processes: Limit Theory and Statistical Applications*. Springer.

Danruo Deng, Guangyong Chen, Yang Yu, Furui Liu, and Pheng-Ann Heng. 2023a. Uncertainty estimation by fisher information-based evidential deep learning. In *ICML*. PMLR.

Zhijie Deng, Peng Cui, and Jun Zhu. 2023b. Towards accelerated model training via bayesian data selection. In *NeurIPS*.

Yunshu Du, Wojciech M Czarnecki, Siddhant M Jayakumar, Mehrdad Farajtabar, Razvan Pascanu, and Balaji Lakshminarayanan. 2018. Adapting auxiliary losses using gradient similarity. *arXiv preprint arXiv:1812.02224*.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. 2023. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. In *NeurIPS*, volume 36, pages 50742–50768.

Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, Dinesh Manocha, and 1 others. 2024. A closer look at the limitations of instruction tuning. *arXiv preprint arXiv:2402.05119*.

Yasir Ghunaim, Adel Bibi, Kumail Alhamoud, Motasem Alfarra, Hasan Abed Al Kader Hammoud, Ameya Prabhu, Philip HS Torr, and Bernard Ghanem. 2023. Real-time evaluation in online continual learning: A new hope. In *CVPR*.

Annette Rios Gonzales, Nicolas Spring, Tannon Kew, Marek Kostrzewa, Andreas Säuberli, Mathias Müller, and Sarah Ebling. 2021. A new dataset and efficient baselines for document-level text simplification in german. In *ACL Workshop*, pages 152–161.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Haiyang Guo, Fanhu Zeng, Ziwei Xiang, Fei Zhu, Da-Han Wang, Xu-Yao Zhang, and Cheng-Lin Liu. 2025. Hide-llava: Hierarchical decoupling for continual instruction tuning of multimodal large language model. *arXiv preprint arXiv:2503.12941*.

Animesh Gupta, Irtiza Hasan, Dilip K Prasad, and Deepak K Gupta. 2023. Data-efficient training of cnns and transformers with coresets: A stability perspective. *arXiv preprint arXiv:2303.02095*.

Jinghan He, Haiyun Guo, Ming Tang, and Jinqiao Wang. 2023. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. 2020. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*.

Yongquan He, Wenyuan Zhang, Xuancheng Huang, Peng Zhang, Lingxun Meng, Xiang Zhou, Ke Zeng, and Xunliang Cai. 2025. Don't half-listen: Capturing key-part information in continual instruction tuning. In *ACL*.

Aral Hekimoglu, Adrian Brucker, Alper Kagan Kayali, Michael Schmidt, and Alvaro Marcos-Ramiro. 2023. Active learning for object detection with non-redundant informative sampling. *arXiv preprint arXiv:2307.08414*.

Feng Hong, Yueming Lyu, Jiangchao Yao, Ya Zhang, Ivor W Tsang, and Yanfeng Wang. 2024. Diversified batch selection for training acceleration. *arXiv preprint arXiv:2406.04872*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*.

Yebowen Hu, Tim Ganter, Hanieh Deilamsalehy, Franck Dernoncourt, Hassan Foroosh, and Fei Liu. 2023. Meetingbank: A benchmark dataset for meeting summarization. In *ACL*.

Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Cihang Xie, and Yuyin Zhou. 2025. HQ-edit: A high-quality dataset for instruction-based image editing. In *ICLR*.

Zhouyuan Huo, Bin Gu, and Heng Huang. 2018. Training neural networks using features replay. In *NeurIPS*.

Mohit Iyyer, Varun Manjunatha, Anupam Guha, Yogarshi Vyas, Jordan Boyd-Graber, Hal Daume, and Larry S Davis. 2017. The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives. In *CVPR*, pages 7186–7195.

Mohammad Jafari, Yimeng Zhang, Yihua Zhang, and Sijia Liu. 2024. The power of few: Accelerating and enhancing data reweighting with coreset selection. In *ICASSP*, pages 7100–7104. IEEE.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. 2024. Mantis: Interleaved multi-image instruction tuning. *Transactions on Machine Learning Research*.

Huaizu Jiang, Xiaojian Ma, Weili Nie, Zhiding Yu, Yuke Zhu, and Anima Anandkumar. 2022. Bongard-hoi: Benchmarking few-shot visual reasoning for human-object interactions. In *CVPR*, pages 19056–19065.

Bing-Yi Jing, Qi-Man Shao, and Qi Wang. 2003. Self-normalized cramér-type large deviations for independent random variables. *Annals of Probability*, 31(4):2167–2215.

Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. 2018. Dvqa: Understanding data visualizations via question answering. In *CVPR*, pages 5648–5656.

Angelos Katharopoulos and François Fleuret. 2018. Not all samples are created equal: Deep learning with importance sampling. In *ICML*, pages 2525–2534. PMLR.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in neural information processing systems*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

Hyunseo Koh, Dahyun Kim, Jung-Woo Ha, and Jonghyun Choi. 2022. Online continual learning on class incremental blurry task configuration with anytime inference. In *ICLR*.

Hyunseo Koh, Minhyuk Seo, Jihwan Bang, Hwanjun Song, Deokki Hong, Seulki Park, Jung-Woo Ha, and Jonghyun Choi. 2023. Online boundary-free continual learning by scheduled data prior. In *ICLR*.

Kamil Książek, Hubert Jastrzębski, Bartosz Trojan, Krzysztof Pniaczek, Michał Karp, and Jacek Tabor. 2025. Fenec: Enhancing continual learning via feature clustering with neighbor-or logit-based classification. *arXiv preprint arXiv:2503.14301*.

Allison Lau, Younwoo Choi, Vahid Balazadeh, Keertana Chidambaram, Vasilis Syrgkanis, and Rahul G Krishnan. 2024. Personalized adaptation via in-context preference learning. *arXiv preprint arXiv:2410.14001*.

Jaewoo Lee, Boyang Li, and Sung Ju Hwang. 2024. Concept-skill transferability-based data selection for large vision-language models. In *EMNLP*.

Lei Li, Yuwei Yin, Shicheng Li, Liang Chen, Peiyi Wang, Shuhuai Ren, Mukai Li, Yazheng Yang, Jingjing Xu, Xu Sun, and 1 others. 2023a. M^3 it: A large-scale dataset towards multi-modal multilingual instruction tuning. *arXiv preprint arXiv:2306.04387*.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2023b. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *CVPR*.

Zikang Liu, Kun Zhou, Wayne Xin Zhao, Dawei Gao, Yaliang Li, and Ji-Rong Wen. 2024b. Less is more: High-value data selection for visual instruction tuning. *arXiv preprint arXiv:2403.09559*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. 2021a. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. In *NeurIPS*.

Shuai Lu, Daya Guo, Shuo Ren, Junjie Huang, Alexey Svyatkovskiy, Ambrosio Blanco, Colin Clement, Dawn Drain, Daxin Jiang, Duyu Tang, and 1 others. 2021b. Codexglue: A machine learning benchmark dataset for code understanding and generation. In *NeurIPS*.

Chengcheng Ma, Yang Liu, Jiankang Deng, Lingxi Xie, Weiming Dong, and Changsheng Xu. 2023. Understanding and mitigating overfitting in prompt tuning for vision-language models. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4616–4629.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Adyasha Maharana, Jaehong Yoon, Tianlong Chen, and Mohit Bansal. 2025. Adapt-∞: Scalable continual multimodal instruction tuning via dynamic data selection. In *ICLR*.

Shivangi Mahto, Vy Ai Vo, Javier S Turek, and Alexander Huth. 2020. Multi-timescale representation learning in lstm language models. In *ICLR*.

Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *ACL*, pages 2263–2279.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *WACV*.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, and 1 others. 2022. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *ICML*. PMLR.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022. Numglue: A suite of fundamental yet challenging mathematical reasoning tasks. In *ACL*.

Aristeidis Panos, Rahaf Aljundi, Daniel Olmeda Reino, and Richard E Turner. 2025. Efficient few-shot continual learning in vision-language models. *arXiv preprint arXiv:2502.04098*.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. 2021. Deep learning on a data diet: Finding important examples early in training. In *NeurIPS*.

Bill Psomas, Ioannis Kakogeorgiou, Nikos Efthymiadis, Giorgos Tolias, Ondřej Chum, Yannis Avrithis, and Konstantinos Karantzalos. 2024. Composed image retrieval for remote sensing. In *IGARSS*, pages 8526–8534. IEEE.

Ziheng Qin, Kai Wang, Zangwei Zheng, Jianyang Gu, Xiangyu Peng, Zhaopan Xu, Daquan Zhou, Lei Shang, Baigui Sun, Xuansong Xie, and 1 others. 2024. Infobatch: Lossless training speed up by unbiased dynamic data pruning. In *ICLR*.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. 2023. Progressive prompts: Continual learning for language models. In *ICLR*.

Leslie Rice, Eric Wong, and Zico Kolter. 2020. Overfitting in adversarially robust deep learning. In *ICML*.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. In *NeurIPS*.

Minhyuk Seo, Seongwon Cho, Minjae Lee, Diganta Misra, Hyeonbeom Choi, Seon Joo Kim, and Jonghyun Choi. 2024a. Just say the name: Online continual learning with category names only via data generation. *arXiv preprint arXiv:2403.10853*.

Minhyuk Seo, Hyunseo Koh, and Jonghyun Choi. 2025. Budgeted online continual learning by adaptive layer freezing and frequency-based sampling. In *ICLR*.

Minhyuk Seo, Hyunseo Koh, Wonje Jeung, Minjae Lee, San Kim, Hankook Lee, Sungjun Cho, Sungik Choi, Hyunwoo Kim, and Jonghyun Choi. 2024b. Learning equi-angular representations for online continual learning. In *CVPR*, pages 23933–23942.

Agam Shah, Suvan Paturi, and Sudheer Chava. 2023. Trillion dollar words: A new financial dataset, task & market analysis. In *ACL*.

Hyo-Sang Shin and Hae-In Lee. 2020. A new exponential forgetting algorithm for recursive least-squares parameter estimation. *arXiv preprint arXiv:2004.03910*.

Seungjae Shin, Heesun Bae, Donghyeok Shin, Weonyoung Joo, and Il-Chul Moon. 2023. Loss-curvature matching for dataset selection and condensation. In *AISTATS*. PMLR.

Alexander Soen and Ke Sun. 2021. On the variance of the fisher information for deep learning. In *NeurIPS*.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. 2022. Beyond neural scaling laws: beating power law scaling via data pruning. In *NeurIPS*.

Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. 2017. A corpus of natural language for visual reasoning. In *ACL*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Xiao Wang, Yuansen Zhang, Tianze Chen, Songyang Gao, Senjie Jin, Xianjun Yang, Zhiheng Xi, Rui Zheng, Yicheng Zou, Tao Gui, and 1 others. 2023. Trace: A comprehensive benchmark for continual learning in large language models. *arXiv preprint arXiv:2310.06762*.

Ziqi Wang, Chang Che, Qi Wang, Yangyang Li, Zenglin Shi, and Meng Wang. 2024. Separable mixture of low-rank adaptation for continual visual instruction tuning. *arXiv preprint arXiv:2411.13949*.

Haoning Wu, Hanwei Zhu, Zicheng Zhang, Erli Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Annan Wang, Wenxiu Sun, Qiong Yan, and 1 others. 2024a. Towards open-ended visual quality comparison. In *ECCV*, pages 360–377. Springer.

Rujie Wu, Xiaojian Ma, Zhenliang Zhang, Wei Wang, Qing Li, Song-Chun Zhu, and Yizhou Wang. 2024b. Bongard-openworld: Few-shot reasoning for free-form visual concepts in the real world. In *ICLR*.

Zhiyang Xu, Ying Shen, and Lifu Huang. 2023. Multi-instruct: Improving multi-modal zero-shot learning via instruction tuning. In *ACL*, pages 11445–11465. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, and 1 others. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Zhenfei Yin, Jiong Wang, Jianjian Cao, Zhelun Shi, Dingning Liu, Mukai Li, Xiaoshui Huang, Zhiyong Wang, Lu Sheng, Lei Bai, and 1 others. 2023. Lamm: Language-assisted multi-modal instruction-tuning dataset, framework, and benchmark. In *NeurIPS*.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. 2024. Investigating the catastrophic forgetting in multimodal large language models. In *CPAL*.

Jiarui Zhang. 2024. Guided profile generation improves personalization with llms. In *ACL*.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. 2023. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *NeurIPS*. Curran Associates, Inc.

Chenye Zhao, Yingjie Li, and Cornelia Caragea. 2023. C-stance: A large dataset for chinese zero-shot stance detection. In *ACL*, pages 13369–13385.

Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. 2023. Coverage-centric coreset selection for high pruning rates. In *ICLR*.

Junhao Zheng, Qianli Ma, Zhen Liu, Binquan Wu, and Huawen Feng. 2024. Beyond anti-forgetting: Multimodal continual instruction tuning with positive forward transfer. *arXiv preprint arXiv:2401.09181*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2024a. MiniGPT-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*.

Didi Zhu, Zhongyi Sun, Zexi Li, Tao Shen, Ke Yan, Shouhong Ding, Kun Kuang, and Chao Wu. 2024b. Model tailor: Mitigating catastrophic forgetting in multi-modal large language models. In *ICML*.

A Technical Appendices and Supplementary Material

A.1 Proof of Theorem 3.1

In continual instruction tuning (CIT), the data distribution evolves gradually over time (*e.g.*, more Christmas-related samples in winter and more air-conditioner-related samples in summer) (Koh et al., 2023). Although this constitutes a distribution shift, within short horizons (*e.g.*, a week), the distribution of training batches can be regarded as approximately stable under memory-only training (Koh et al., 2022; Seo et al., 2024a), even in the presence of sharp task boundaries. Specifically, memory-only training constructs batches by only retrieving from an episodic memory containing previously encountered samples, thereby inducing smoother changes of training distribution, unlike ER (Rolnick et al., 2019), which mixes continuously encountered streaming data with memory samples and may cause abrupt shifts. As a result, our choice of memory-only training for CIT motivates the local stationarity assumption (Assumption 1) and local weak dependence assumption (Assumption 2) introduced below.

Setting. At time t, we observe a batch $\{(x_i^{(t)}, y_i^{(t)})\}_{i=1}^{N_\mathcal{B}}$ and calculate sample-wise FI $\{I_i^{(t)}\}_{i=1}^{N_\mathcal{B}}$ using Fisher Information Matrix (Eq. 2), while updating EMA μ_t and EMV σ_t of FI (Eq. 3) using FI averaged by batch ($\bar{I}^{(t)} = \frac{1}{N_\mathcal{B}} \sum_{i=1}^{N_\mathcal{B}} I_i^{(t)}$). We decompose $\bar{I}^{(t)}$ as

$$\bar{I}^{(t)} = m_t + \varepsilon_t,$$

where $m_t := \mathbb{E}[\bar{I}^{(t)}]$ denotes the underlying (population) FI at time t, and $\varepsilon_t := \bar{I}^{(t)} - m_t$ represents the centered sampling noise.

Assumption 1 (Local Stationarity)**.** *Although the data distribution P_t and model parameters θ_t evolve over time in continual learning, we assume that such changes are* smooth. *In particular, within a short horizon comparable to the EMA effective window $L_{\text{eff}} \approx \frac{1+\beta}{1-\beta}$, the mean $m_t = \mathbb{E}[\bar{I}^{(t)}]$ and variance $s_t^2 = \text{Var}(\bar{I}^{(t)})$ remain nearly constant. Formally, for temporal lag $h \in \mathbb{Z}$ with $|h| \le L_{\text{eff}}$,*

$$\sup_{|h| \le L_{\text{eff}}} |m_{t+h} - m_t| = o(\sqrt{1-\beta}), \tag{9}$$

$$\sup_{|h| \le L_{\text{eff}}} |s_{t+h}^2 - s_t^2| = o(1). \tag{10}$$

Assumption 2 (Local Weak Dependence)**.** *The noise $\{\varepsilon_t\}$ is assumed to be* locally weakly dependent. *That is, while sequential batches are correlated due to temporal proximity, correlations decay sufficiently fast as the temporal lag h grows. Formally, for lag $h \to \infty$,*

$$\text{Cov}(\varepsilon_t, \varepsilon_{t+h}) \to 0.$$

Lemma A.1 ($\hat{I}^{(t)}$ has finite $(2+\delta)$-moments)**.** *We estimate the Fisher information (FI) as the trace of the Fisher information matrix (FIM) in Eq. 2. Since the diagonal entries of the FIM are typically large in magnitude, the FI of a sample $(x_i^{(t)}, y_i^{(t)}) \in \mathcal{B}_t$, denoted $I_i^{(t)}$, can be approximated by a chi-square distribution (Dauncey et al., 2024),*

$$I_i^{(t)} \overset{d}{\approx} \sigma_t^2 \chi^2_{k_t},$$

where σ_t^2 is a scale parameter chosen such that the mean and variance match those of the Fisher information trace, i.e.,

$$\mathbb{E}[I_i^{(t)}] \approx \sigma_t^2 k_t, \quad \text{Var}(I_i^{(t)}) \approx 2\sigma_t^4 k_t.$$

Because both the sum of chi-square variables remains chi-square distributed, the batch-averaged FI $\bar{I}^{(t)}$ also follow chi-square approximations:

$$\bar{I}^{(t)} \overset{d}{\approx} \frac{\sigma_t^2}{B} \chi^2_{k_t \mathcal{B}_t}.$$

Since chi-square distributions have finite moments of all orders, it follows that for every $\delta > 0$,

$$\mathbb{E}\Big[\, |\bar{I}^{(t)}|^{2+\delta} \Big] < \infty, \quad \mathbb{E}\Big[\, |I_i^{(t)}|^{2+\delta} \Big] < \infty.$$

Lemma A.2 (Normal approximation rate for FI trace via chi-square)**.** *Let $X \sim \chi^2_k$ and consider the standardized variable $\widetilde{X} = (X-k)/\sqrt{2k}$. Then classical Berry-Esseen-type results (Dasgupta (2008); Chen et al. (2010)) guarantee that*

$$\sup_{x \in \mathbb{R}} \big| \Pr(\widetilde{X} \le x) - \Phi(x) \big| = O\left(\frac{1}{\sqrt{k}}\right).$$

Applied to $\bar{I}^{(t)}$ with $\bar{I}^{(t)} \approx \frac{\sigma_t^2}{\mathcal{B}_t} \chi^2_{k_t \mathcal{B}_t}$, we obtain

$$\sup_x \left| \Pr\left(\frac{\bar{I}^{(t)} - \mathbb{E}[\bar{I}^{(t)}]}{\sqrt{\text{Var}(\bar{I}^{(t)})}} \le x \right) - \Phi(x) \right| = O\left(\frac{1}{\sqrt{k_t \mathcal{B}_t}}\right). \tag{11}$$

Lemma A.3 (EMA/EMV are consistent approximations of the true mean and variance)**.** *Let $m_t = \mathbb{E}[\bar{I}^{(t)}]$ be the true mean of the batch-averaged FI and $s_t^2 = \mathrm{Var}(I_i^{(t)})$ the true sample-wise variance at time t. Let μ_{t-1} and v_{t-1} be the EMA and EMV computed from past batches, and define $\widehat{\sigma}_{t-1}^2 := B \cdot v_{t-1}$. Then, under Assumptions 1 and 2,*

$$|\mu_{t-1} - m_t| = o(s_t), \qquad \frac{\widehat{\sigma}_{t-1}^2}{s_t^2} \xrightarrow{p} 1.$$

In words: *the EMA mean μ_{t-1} is essentially the same as the true mean m_t, and the EMV-based variance estimator $\widehat{\sigma}_{t-1}^2$ consistently estimates the true variance s_t^2.*

Proof Lemma A.3. **Mean part.** The EMA is a weighted moving average:

$$\mu_{t-1} = (1-\beta) \sum_{k \geq 1} \beta^{k-1} \bar{I}_{t-k}.$$

Local stationarity (Assumption 1) says that the true mean m_t does not change much over the EMA window $L_{\text{eff}} \approx \frac{1+\beta}{1-\beta}$. Hence each difference $|m_{t-k} - m_t|$ is small, and the exponential weights $(1-\beta)\beta^{k-1}$ quickly decay. As a result, the EMA bias

$$|\mu_{t-1} - m_t| = (1-\beta) \sum_{k \geq 1} \beta^{k-1} |m_{t-k} - m_t|$$

is negligible compared to the natural variability s_t; formally, it is $o(s_t)$.

Variance part. The EMV is defined as

$$v_{t-1} = (1-\lambda) \sum_{k \geq 1} \lambda^{k-1} (\bar{I}_{t-k} - \mu_{t-k})^2.$$

This is an exponentially weighted average of past squared deviations. Local weak dependence (Assumption 2) ensures that correlations between far-apart batches vanish, so v_{t-1} consistently estimates $\mathrm{Var}(\bar{I}_t)$ (this is the usual consistency of exponential-kernel HAC estimators). Finally, since $\mathrm{Var}(\bar{I}_t) = s_t^2/B$ under i.i.d. samples in the batch, multiplying v_{t-1} by B yields a consistent estimator of the sample-wise variance s_t^2.

Thus, both the EMA mean and the EMV variance converge to their true population counterparts. □

Proof of Theorem 3.1. We prove that the one-step-lag standardized statistic $Z_{t,i} = \frac{I_i^{(t)} - \mu_{t-1}}{\widehat{\sigma}_{t-1}}$ is well-approximated by $\mathcal{N}(0,1)$ under Assumptions 1–2 and the chi-square structure of FI traces.

Step 1: Within-batch normal approximation. By Lemma A.1, the sample-wise FI admits a (scaled) chi-square representation: $I_i^{(t)} \overset{d}{\approx} \sigma_t^2 \chi_{k_t}^2$ for some effective degrees-of-freedom k_t. Therefore, by Lemma A.2,

$$\sup_{x \in \mathbb{R}} \left| \Pr\left(\frac{I_i^{(t)} - m_t}{s_t} \leq x \right) - \Phi(x) \right| = O\big(k_t^{-1/2}\big), \tag{12}$$

where $m_t = \mathbb{E}[\bar{I}_t] = \mathbb{E}[I_i^{(t)}]$ and $s_t^2 = \mathrm{Var}(I_i^{(t)})$.

Step 2: EMA/EMV ratio-consistency. The ideal normalization requires (m_t, s_t), but in practice we use the EMA mean μ_{t-1} and EMV variance $\widehat{\sigma}_{t-1}^2$ from past batches. By Lemma A.3, local stationarity ensures that $|\mu_{t-1} - m_t| = o(s_t)$, and local weak dependence guarantees $\widehat{\sigma}_{t-1}^2 / s_t^2 \xrightarrow{p} 1$. Hence $(\mu_{t-1}, \widehat{\sigma}_{t-1})$ consistently approximate (m_t, s_t), justifying their use as practical normalizers.

Step 3: Slutsky/self-normalization. We can decompose $Z_{t,i}$ as

$$Z_{t,i} = \frac{I_i^{(t)} - m_t}{s_t} \cdot \frac{s_t}{\widehat{\sigma}_{t-1}} - \frac{\mu_{t-1} - m_t}{\widehat{\sigma}_{t-1}}. \tag{13}$$

By (12), the standardized term $(I_i^{(t)} - m_t)/s_t$ is $O(k_t^{-1/2})$, close to $\mathcal{N}(0,1)$ in Kolmogorov distance. Lemma A.3 further ensures that $s_t/\widehat{\sigma}_{t-1} \xrightarrow{p} 1$ and $(\mu_{t-1} - m_t)/\widehat{\sigma}_{t-1} = o_p(1)$. Hence, both adjustment factors vanish asymptotically, and by Slutsky's theorem the distribution of $Z_{t,i}$ retains the normal approximation:

$$\sup_{x \in \mathbb{R}} \Big| \Pr(Z_{t,i} \leq x) - \Phi(x) \Big| \;\leq\; C_1\, k_t^{-1/2} \,+\, o(1).$$

Step 4: One-step lag and leakage. We used $(\mu_{t-1}, \widehat{\sigma}_{t-1})$ built from *past* batches only, so $I_i^{(t)}$ is not used inside its own normalizer; hence no self-normalization bias occurs. (If one includes t in the EMA/EMV, the induced bias is $O(1-\beta)$.)

Combining the steps establishes the stated claim $Z_{t,i} \approx \mathcal{N}(0,1)$ and the quantitative bound. □

A.2 Implementation Details

We fine-tune models using learning rates from their original papers for one epoch: 2e-5 for both LLaVA-1.5-7B and Qwen-VL-2.5-7B and 3e-4 for both LLaMA-3.1-8B and Qwen3-8B. We use the Adam optimizer with no decay and a Cosine LR scheduler. For batch size $N_{\mathcal{B}}$, we use 16 for

all experiments. For LoRA finetuning, we adopt LoRA with rank 128 at all linear layers in the LLM backbone. We set the number of batch iterations per sample encounter to COAST, Adapt, and MICVIT as 0.125, 0.125, and 0.0625, respectively. We assume an infinite memory setup, which assumes that all encountered samples can be stored in an episodic memory. For the batch retrieval method, we adopt memory-only training (Koh et al., 2022; Seo et al., 2024b), where training batches are only retrieved from the episodic memory at each iteration, enhancing robustness to distribution shifts (Koh et al., 2023; Seo et al., 2025). We set the EMA ratio β to 0.9 for all datasets. We conduct experiments on NVIDIA RTX A6000 GPUs. Every experiment takes no more than 2 days.

A.3 Experimental Results on Various Benchmarks

In addition to MICVIT, we compare OASIS with sample selection methods in COAST (Cao et al., 2024) and Adapt (Maharana et al., 2025) under selection ratios of 12.5% and 25.0% in Tab. 6. OASIS not only outperforms selection baselines but also achieves comparable results with a model trained on all data (*i.e.*, Full-Data Training) in diverse benchmarks.

A.4 Benchmark Configuration Details

We use Long Sequence (Razdaibiedina et al., 2023) and TRACE (Wang et al., 2023) as our text-only benchmarks. Long Sequence consists of datasets from two existing CL benchmarks, GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019), along with additional IMDB movie reviews dataset (Maas et al., 2011). TRACE benchmark consists of 8 tasks, NumGLUE-ds (Mishra et al., 2022), ScienceQA (Lu et al., 2022), Py150 (Lu et al., 2021b), C-STANCE (Zhao et al., 2023), FOMC (Shah et al., 2023), 20Minuten (Gonzales et al., 2021), NumGLUE-cm (Mishra et al., 2022), and MeetingBank (Hu et al., 2023), encompassing domain-specific tasks, multilingual capabilities, code generation, and mathematical reasoning.

We use COAST (Cao et al., 2024), Adapt (Maharana et al., 2025), and MICVIT for our multimodal benchmarks. For the COAST benchmark, we use the COAST-domain, which emulates a scenario where MLLMs needs to continuously adapt to diverse domains. Specifically, it consists of DocVQA (Mathew et al., 2021), ChartQA (Masry et al., 2022), IconQA (Lu et al., 2021a), and MedicalQA (He et al., 2020), which correspond to the document, chart, icon, and medical domains, respectively. Adapt benchmark consists of visual instruction tuning datasets, such as M3IT (Li et al., 2023a), MiniGPT4 (Zhu et al., 2023), MANTIS (Jiang et al., 2024), LaMM (Yin et al., 2023), and VisionFLAN (Xu et al., 2023). MICVIT comprises NLVR2 (Suhr et al., 2017), Bongard-OpenWorld (Wu et al., 2024b), Bongard-HOI (Jiang et al., 2022), Co-Instruct-DB (Wu et al., 2024a), DVQA (Kafle et al., 2018), HQ-Edit (Hui et al., 2025), and PatternCom (Psomas et al., 2024). We summarize the task configuration of each benchmark, including MICVIT in Tab. 7.

A.5 Sample Selection Baselines

Self-Sup (Sorscher et al., 2022). Self-Sup performs K-means clustering on the output embeddings from the final layer of the pretrained model and selects the samples closest to each cluster centroid. By doing so, the most representative sample in the underlying data distribution can be selected.

GradNorm (Katharopoulos and Fleuret, 2018). GradNorm is an importance sampling method that prioritizes the selection of highly informative samples based on gradient magnitude. It computes each sample's gradient norm with respect to the model's last layer parameters and assigns higher selection probabilities to samples with larger gradient norms.

COINCIDE (Lee et al., 2024). COINCIDE leverages multi-layer output representations for K-means clustering. Specifically, features are extracted from five distinct layers of the MLLM to obtain the representation of the candidate samples. Then it applies K-means clustering to these features, and computes transferability and density metrics of each cluster to determine how many samples should be selected from the respective cluster. Finally, within each cluster, the maximum mean discrepancy (MMD) (Kim et al., 2016) is computed greedily to obtain the samples that best reflect the overall cluster distribution.

DBP (Abbas et al., 2024). DBP aims to improve dataset quality by removing redundant and irrelevant samples. First, following Semdedup (Abbas et al., 2023), DBP applies K-means clustering to group similar samples with last layer outputs and eliminate semantic duplicates by retaining only the sample farthest from the cluster centroid among highly similar pairs. Next, DBP performs CLIP

Method	Selection Ratio (%)							
	12.5				25.0			
	COAST		Adapt		COAST		Adapt	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	31.56±1.42	39.06±0.55	55.89±0.20	54.26±0.23	31.56±1.42	39.06±0.55	55.89±0.20	54.26±0.23
Random	24.58±0.17	32.45±0.92	49.04±0.49	44.19±0.62	25.82±0.10	34.28±0.28	50.68±0.30	47.58±0.34
GradNorm (ICML 2018)	23.35±0.38	29.44±0.28	48.47±0.63	45.36±0.62	25.67±0.21	32.85±0.36	51.29±0.54	48.96±0.81
Self-Sup (NeurIPS 2022)	23.72±0.27	31.37±0.75	47.04±0.26	43.94±1.38	25.17±0.37	32.84±0.42	49.34±0.85	46.46±0.47
COINCIDE (EMNLP 2024)	24.45±0.84	31.08±0.59	48.52±0.55	44.76±0.43	26.22±0.34	34.90±0.57	51.74±0.64	47.40±0.31
DBP (ICLR 2024)	22.87±0.33	25.81±0.37	46.24±0.36	43.30±1.08	24.74±0.70	32.13±0.64	48.58±0.43	45.65±0.84
InfoBatch (ICLR 2024)	24.78±0.58	31.74±0.62	49.77±0.82	47.54±0.36	25.30±0.35	34.49±0.45	51.55±0.47	49.04±0.77
DivBS (ICML 2024)	25.13±0.12	33.30±0.87	49.96±0.48	47.17±0.51	26.91±0.02	35.01±0.16	52.36±0.05	50.02±1.21
TIVE (arXiv:2403)	23.41±0.23	30.17±0.43	47.75±0.73	44.38±0.99	25.18±0.78	34.08±0.60	49.69±1.24	45.97±0.88
Adapt-∞ (ICLR 2025)	24.73±0.36	31.69±0.35	48.38±0.20	45.42±0.46	26.07±0.25	33.50±0.04	50.56±0.25	47.54±0.53
OASIS (**Ours**)	**27.13±0.70**	**35.42±0.49**	**51.73±0.33**	**49.02±0.85**	**28.72±0.63**	**37.55±0.28**	**54.58±0.14**	**51.87±0.49**

Table 6: **Quantitative comparison on COAST and Adapt under different selection ratios (12.5% and 25.0%).** LLaVA-1.5-7B is used as the MLLM. Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level. 'Full-Data Training' uses all data without selection.

Dataset	# of samples / task	# of tasks	Task order
COAST (Cao et al., 2024)	20,000	4	ChartQA → DocVQA → IconQA → MedicalQA
Adapt (Maharana et al., 2025)	80,000	4	M3IT → MANTIS → LaMM → VisionFLAN
MICVIT (Ours)	50K / 30K / 100K / 46K / 17K / 25K / 79K	7	Bongard-OpenWorld → NLVR2 → Co-Instruct-DB → Bongard-HOI → PatternCom → DVQA → HQ Edit
Long Sequence (Razdaibiedina et al., 2023)	10,000	15	DBPedia → RTE → WiC → Amazon → Yahoo → AG News → MNLI → IMDB → MultiRC → CB → Yelp → COPA → QQP → SST2 → BoolQ
TRACE (Wang et al., 2023)	5,000	8	NumGLUE-ds → ScienceQA → Py150 → C-STANCE → FOMC → 20Minuten → NumGLUE-cm → MeetingBank

Table 7: **Task configurations of CIT benchmarks.**

score filtering, removing samples with low text-image similarities. Then, to prioritize more informative data, each cluster is evaluated for its complexity using two metrics: d_{inter}, the distance from the cluster centroid to other centroids, and d_{intra}, the average cosine distance of samples to their own centroid. A complexity score $C_j = d_{\text{inter},j} \cdot d_{\text{intra},j}$ is calculated for each cluster j, and samples are selected proportionally based on this score. Within each selected cluster, samples with the highest entropy are retained, ensuring that the final dataset is both diverse and rich in information.

InfoBatch (Qin et al., 2024). InfoBatch introduces a soft pruning strategy to improve training efficiency while preserving learning dynamics. Unlike traditional hard pruning methods that permanently discard data and risk introducing bias, InfoBatch probabilistically excludes a subset of well-learned samples—identified by their low loss values—during each training epoch. Furthermore, to maintain the integrity of the training trajectory, the gradients of the remaining samples are rescaled such that the expected aggregated gradient approximates that of the full batch.

DivBS (Abbas et al., 2024). From each online batch, DivBS focuses on selecting a fixed subset of training data that is both informative and diverse by maximizing the orthogonalized representativeness. To do so, DivBS utilizes the last layer gradient. DivBS aims to explicitly reduce inter-sample redundancy, ensuring the retained samples capture complementary and non-overlapping aspects of the batch distribution.

TIVE (Liu et al., 2024b). TIVE selects samples based on two criteria: instance influence and task difficulty, both derived from gradients computed across all layers of a reference model. Instance influence quantifies how much a given sample contributes to other samples during training, while task difficulty estimates the inherent complexity of learning a given task. Guided by these two metrics, TIVE prioritizes samples that are both highly influential and associated with difficult tasks.

Adapt-∞ (Maharana et al., 2025). Rather than utilizing layer output representations, Adapt-∞ uses gradients from the middle layer to cluster data into pseudo-tasks, enabling the model to identify and group related skills without requiring explicit task labels. It then performs multi-way sample

selection within each cluster using a pool of scoring functions—such as entropy and the proposed Image Grounding score—to retain the most informative samples. Unlike our online continual instruction tuning setting, Adapt-∞ was introduced in an offline continual instruction tuning setting, where task boundary assumptions exist, limiting its real-world applicability.

A.6 Experiments with Qwen-VL-2.5

In addition to experiments with LLaVA-1.5-7B (Liu et al., 2024a) in Sec. 4.2, we also compare OASIS with sample selection baselines using Qwen-VL-2.5-7B (Bai et al., 2025) as the MLLM. Specifically, we compare OASIS with baselines on COAST, Adapt, and MICVIT, and summarize the results in Tab. 8, Tab. 9, Tab. 10 respectively. As shown in the tables, OASIS outperforms the baselines, consistent with the results observed using LLaVA-1.5-7B.

A.7 Details on Determining I_T

As described in Sec. 3.3, we probabilistically select a sample based on its relative informativeness $\hat{I}$, with the selection probability defined as:

$$p(\hat{I}) = \sigma(\hat{I} - I_T), \tag{14}$$

where $\sigma(\cdot)$ denotes the sigmoid function and I_T is a predefined threshold. We assume $\hat{I} \sim \mathcal{N}(0, 1)$ that the original (*i.e.*, pre-normalized) informativeness I approximately follows a normal distribution. Based on this normal distribution assumption, the expected selection rate across the distribution is:

$$f(I_T) = \mathbb{E}_{\hat{I}\sim\mathcal{N}(0,1)}[\sigma(\hat{I} - I_T)] \tag{15}$$

$$= \int_{-\infty}^{\infty} \sigma(\hat{I} - I_T) \cdot \phi(\hat{I})\, d\hat{I}, \tag{16}$$

where $\phi(z)$ denotes the standard normal probability density function.

Given target selection rate $r \in [0, 1]$ (*e.g.*, 0.125), we find I_T such that $f(I_T) = r$. Since the integral has no closed form, we approximate it numerically using a Riemann sum and solve:

$$I_T = \arg\min_{I_T} (f(I_T) - r)^2. \tag{17}$$

Applying Eq. 17, selection ratios r of 0.0625, 0.125, and 0.25 yield corresponding I_T values of 2.06, 1.53, and 0.89, respectively. However, data distribution variations and the probabilistic selection process may cause the actual proportion of selected samples to deviate from the target ratio. To ensure fair comparison with baseline methods, we fine-tune I_T to achieve sample counts that closely approximate but remain marginally below the target selection ratio.

A.8 Experiments on Various Model Scales

In addition to LLaVA-1.5-7B and Qwen-VL-2.5-7B in the main paper, we compare baselines with various sizes of LLaVAs (*i.e.*, LLaVA-1.5-1B, 3B, and 13B) in Tab. 11 and Tab. 12 and Qwen-VL-2.5-0.5B in Tab. 13. As shown in the tables, OASIS consistently outperforms the baselines across architectures and model sizes by a significant margin, demonstrating its general applicability.

A.9 Detailed Experiments on Fast Adaptation

Beyond the fast adaptation results on MICVIT (Sec. 4.2), we also evaluate fast adaptation performance on COAST. As shown in Fig. 5, subsets selected by OASIS consistently achieve superior results across all downstream tasks, highlighting their strong generalizability. While zero-shot performance (iteration 0) may vary depending on task–subset similarity, fine-tuning results show that OASIS enables faster adaptation, confirming the superior generalization of models trained on its selected subsets.

A.10 Different Information Metrics

We compare OASIS using Fisher Information to other metrics for measuring the informativeness I of samples, including Entropy (Coleman et al., 2020), Perplexity (Li et al., 2023b), and EL2N (Paul et al., 2021). As shown in Tab. 14, using sample-wise FI as our Informativeness outperforms other metrics.

A.11 Effect of Task Order

In Tab. 15, we examine the effect of task order using three different task sequences for the top-3 performing baselines and OASIS on COAST and MICVIT. For each setting, 6.25% of the training data is selected using OASIS or the top three baseline methods. The results demonstrate that regardless of the change in task order, OASIS outperforms other selection methods.

A.12 Effect of EMA Ratio β

We further evaluate the effect of EMA ratio β on OASIS, and summarize the results in Tab. 16. As

Method	Selection Ratio (%)					
	6.25		12.5		25.0	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	34.59±0.44	41.90±.0.48	34.59±0.44	41.90±.0.48	34.59±0.44	41.90±.0.48
Random	24.53±0.47	31.59±0.26	26.39±0.54	34.14±0.55	27.89±0.15	37.32±0.28
GradNorm (ICML 2018)	23.85±0.39	29.69±0.84	25.96±0.57	33.28±0.82	26.47±0.19	35.13±1.14
Self-Sup (NeurIPS 2022)	22.44±0.18	30.28±0.80	24.18±1.26	32.53±0.42	25.81±0.05	33.92±0.59
COINCIDE (EMNLP 2024)	21.93±0.45	30.73±0.16	24.74±0.74	33.84±0.43	27.48±0.57	35.27±0.65
DBP (ICLR 2024)	22.58±0.29	27.92±0.27	24.01±0.48	32.43±0.58	25.35±0.52	34.79±0.61
InfoBatch (ICLR 2024)	24.60±0.36	32.88±0.54	27.48±0.39	34.26±1.39	28.93±0.41	38.12±0.42
DivBS (ICML 2024)	25.45±0.64	33.18±0.51	28.29±0.48	36.08±0.85	30.82±0.28	**39.87±0.74**
TIVE (arXiv:2403)	22.52±0.82	28.90±0.33	25.83±0.62	32.35±0.93	27.14±0.45	34.42±0.14
Adapt-∞ (ICLR 2025)	23.84±0.53	29.48±0.29	24.85±0.84	33.43±0.79	26.78±0.63	35.38±0.38
OASIS (**Ours**)	**27.29±0.18**	**35.68±0.62**	**29.16±0.13**	**36.27±0.75**	**31.04±0.38**	39.38±0.84

Table 8: **Quantitative comparison between online sample selection methods on COAST benchmark.** We use Qwen-VL-2.5-7B as the MLLM. Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level. 'Full-Data Training' uses all data without selection.

Method	Selection Ratio (%)					
	6.25		12.5		25.0	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	45.73±1.68	43.92±0.85	45.73±1.68	43.92±0.85	45.73±1.68	43.92±0.85
Random	34.87±0.50	27.49±0.25	36.02±0.35	32.16±0.62	38.73±0.44	34.14±0.57
GradNorm (ICML 2018)	32.53±0.36	25.41±0.72	34.70±0.28	30.25±0.43	35.38±0.40	32.26±0.42
Self-Sup (NeurIPS 2022)	30.31±0.31	22.04±0.65	31.11±0.64	27.73±0.83	33.86±0.50	31.63±1.35
COINCIDE (EMNLP 2024)	32.76±0.17	24.39±0.31	33.93±0.72	29.34±0.47	35.82±0.08	32.38±0.55
DBP (ICLR 2024)	29.82±0.52	22.26±0.36	31.16±0.27	28.04±0.14	34.24±0.61	30.05±0.72
InfoBatch (ICLR 2024)	34.95±0.20	27.19±0.18	35.46±0.83	30.77±0.54	38.25±0.72	34.10±0.50
DivBS (ICML 2024)	34.24±0.75	28.63±0.91	37.24±0.41	31.68±1.39	40.32±0.53	36.59±1.08
TIVE (arXiv:2403)	31.46±0.69	23.74±0.87	32.05±0.06	28.13±0.28	35.49±0.69	31.90±0.35
Adapt-∞ (ICLR 2025)	33.61±0.83	24.93±0.42	34.00±0.57	28.51±1.54	35.36±0.82	32.37±0.63
OASIS (**Ours**)	**36.04±0.26**	**30.78±0.84**	**38.72±0.62**	**33.10±0.43**	**41.13±0.91**	**37.36± 0.55**

Table 9: **Quantitative comparison between online sample selection methods on Adapt benchmark.** We use Qwen-VL-2.5-7B as the MLLM. Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level. 'Full-Data Training' uses all data without selection.

shown, extremely large or small values of β degrade performance: high β fails to capture the informativeness of past batches, while low β overemphasizes outdated informativeness. To balance this trade-off, we use a moderate value of $\beta = 0.9$ across all benchmarks and selection ratios.

A.13 Accuracy Over Time

We compare the average accuracy of seen tasks at several training time points between our method and baselines. Specifically, we evaluate performance across different selection ratios and benchmarks, with results shown in Fig.6 and Fig.7, respectively. As shown in the figures, our method outperforms baselines at all time points, demonstrating that its superior performance is not limited to specific intervals but is consistently maintained throughout the entire CIT process.

A.14 Comparison of Computational Costs

We compare the computational cost of OASIS with selection baselines in Tab. 17. As shown, all methods require forward passes over candidate samples. Sample selection methods (*e.g.*, Self-Sup and COINCIDE) that rely solely on features do not incur additional backward pass costs. In contrast, gradient-based methods (*e.g.*, OASIS and GradNorm) involve backward passes to compute sample-wise gradients. However, unlike TIVE and Adapt-∞, which introduce substantial overhead due to

Method	Selection Ratio (%)			
	6.25		12.5	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Full-Data Training	72.31±0.42	78.18±0.75	72.31±0.42	78.18±0.75
Random	59.23±0.85	65.94±0.38	62.18±0.19	69.03±0.10
GradNorm	57.49±0.92	63.38±0.64	63.90±0.22	69.34±0.47
Self-Sup	54.29±0.56	61.42±0.31	61.37±0.99	68.32±0.38
COINCIDE	57.34±0.77	62.90±0.48	62.38±0.73	69.23±0.44
DBP	54.53±0.43	60.38±0.56	60.63±0.30	67.81±0.52
InfoBatch	58.14±0.28	65.38±0.87	64.04±0.42	68.26±1.31
DivBS	60.28±0.98	67.48±0.29	65.26±1.34	70.48±0.38
TIVE	55.84±0.49	63.37±0.83	62.11±0.45	67.42±0.73
Adapt-∞	56.32±0.97	62.48±0.38	62.35±0.49	67.82±1.24
OASIS (**Ours**)	**61.64±0.43**	**69.43±0.72**	**66.54±0.88**	**72.35±0.28**

Table 10: **Quantitative comparison with Qwen-VL-2.5-7B on MICVIT benchmark under 6.25% and 12.5% selection ratios.** Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level.

Method	Model Scale					
	LLaVA-1.5-1B		LLaVA-1.5-3B		LLaVA-1.5-13B	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Random	55.04±0.12	59.75±0.30	57.37±0.12	60.39±0.68	67.49±0.47	73.60±0.49
GradNorm (ICML 2018)	53.44±0.05	57.39±0.22	56.19±0.70	60.73±0.08	65.32±0.53	71.50±0.98
Self-Sup (NeurIPS 2022)	52.01±0.33	55.96±0.27	54.59±0.53	56.76±0.50	64.71±0.71	71.62±0.57
COINCIDE (EMNLP 2024)	53.03±0.51	56.94±0.16	56.04±0.40	59.55±0.39	65.47±0.85	72.25±0.86
DBP (ICLR 2024)	51.02±0.23	55.62±0.34	53.32±0.68	55.89±0.39	62.27±0.02	68.37±0.02
InfoBatch (ICLR 2024)	55.48±0.08	60.29±1.09	58.92±1.31	60.39±0.64	66.78±0.45	72.04±0.93
DivBS (ICML 2024)	57.24±0.12	60.22±0.42	58.02±0.72	62.15±0.52	67.28±0.27	73.63±0.18
TIVE (arXiv:2403)	51.34±0.11	56.26±0.08	54.82±0.12	58.90±0.37	64.74±0.02	71.50±0.02
Adapt-∞ (ICLR 2025)	54.09±0.15	57.09±0.21	55.29±0.45	56.81±0.69	63.69±0.68	70.46±0.81
OASIS (**Ours**)	**57.86±0.10**	**62.17±0.25**	**60.46±0.26**	**64.01±0.40**	**68.04±0.21**	**74.60±0.22**

Table 11: **Quantitative comparison between online sample selection methods across model scales on MICVIT under selection ratio 6.25%.** Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level.

Method	Model Scale					
	LLaVA-1.5-1B		LLaVA-1.5-3B		LLaVA-1.5-13B	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Random	20.05±0.21	28.04±0.15	21.20±0.56	29.30±0.22	25.39±0.90	33.84±0.72
GradNorm (ICML 2018)	19.13±0.33	27.49±0.11	19.95±0.80	28.73±0.39	23.38±0.56	32.03±0.51
Self-Sup (NeurIPS 2022)	18.17±0.41	26.36±0.20	17.49±0.86	24.76±0.80	21.47±0.37	29.35±0.22
COINCIDE (EMNLP 2024)	19.46±1.17	28.27±0.09	20.11±0.16	28.74±0.90	23.77±0.79	31.74±0.27
DBP (ICLR 2024)	17.98±0.14	25.57±0.06	17.05±0.69	25.71±1.83	22.24±0.22	31.95±0.62
InfoBatch (ICLR 2024)	18.36±0.18	25.47±0.04	19.92±0.89	29.27±0.56	24.05±0.73	30.48±0.64
DivBS (ICML 2024)	21.21±0.24	28.52±0.13	21.39±0.35	31.41±0.80	26.22±0.10	35.41±0.10
TIVE (arXiv:2403)	18.04±0.25	26.35±0.30	17.47±0.74	26.11±0.62	24.58±0.34	32.86±0.60
Adapt-∞ (ICLR 2025)	19.91±0.16	26.81±0.28	18.69±0.61	28.00±0.58	22.52±0.61	31.20±0.21
OASIS (**Ours**)	**23.03±0.18**	**30.37±0.12**	**23.08±0.17**	**32.84±0.13**	**26.52±0.52**	**36.81±0.65**

Table 12: **Quantitative comparison between online sample selection methods across model scales on COAST benchmark under selection ratio 6.25%.** Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level.

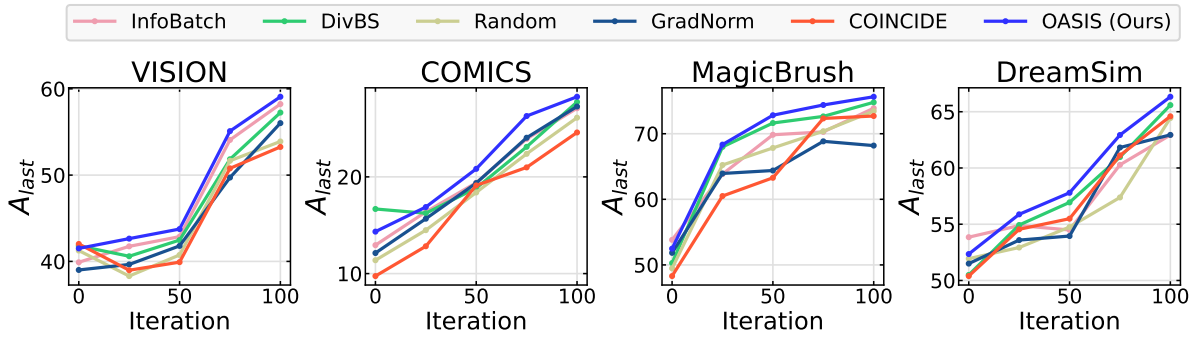

Figure 5: **Comparison of fast adaptation performance.** After CIT of LLaVA-1.5-7B on subsets (25% of the full data), selected using each sample selection baseline from COAST, we fine-tune the model for 100 epochs on each downstream task (*i.e.*, VISION, COMICS, MagicBrush, and DreamSim).

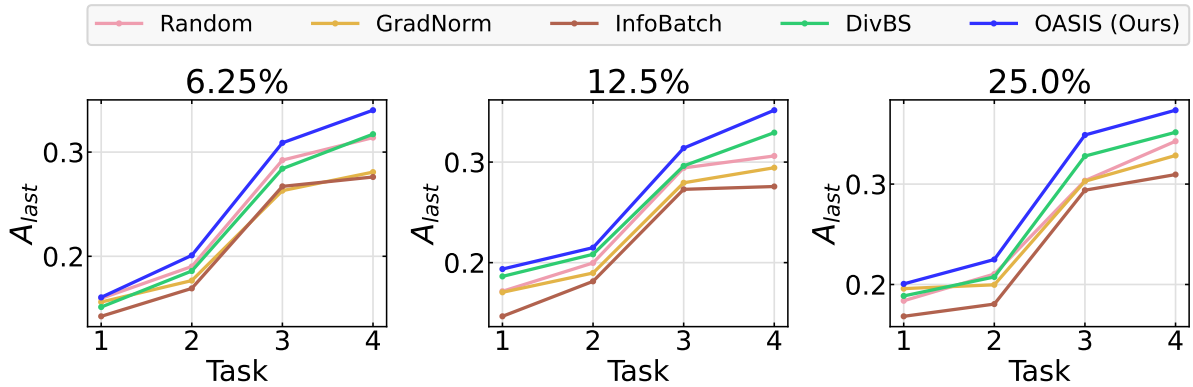

Figure 6: **Average accuracies over time on COAST benchmark under different selection ratios.** Performance at task t denotes the average accuracy over all seen tasks up to that point (*i.e.*, task 1 through task t). We use LLaVA-1.5-7B as the model across all selection ratios.

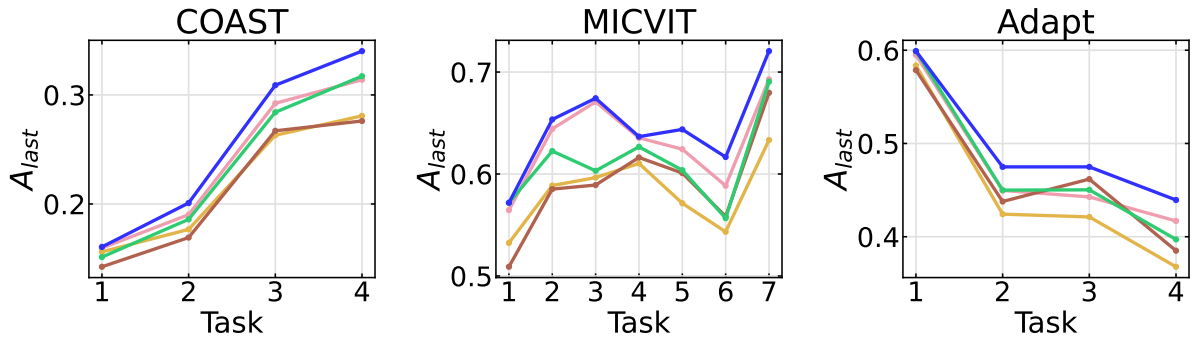

Figure 7: **Average accuracy over time across benchmarks.** Performance at task t denotes the average accuracy over all seen tasks up to that point (*i.e.*, task 1 through task t). We use a selection ratio of 6.25% across all benchmarks. We use LLaVA-1.5-7B as the model.

full-layer gradient computations or intermediate layer calculations, OASIS leverages only the last-layer gradients, resulting in negligible overhead compared to the forward pass (*i.e.*, less than 3.5%).

A.15 Distribution of data selected by selection baselines

The selection baselines that consistently underperform random selection across all selection ratios, namely Self-Sup, COINCIDE, DBP, TIVE, and Adapt-∞, rely on K-means clustering, which often yields imbalanced clusters (*i.e.*, certain clusters contain disproportionately more data) under imbalanced distributions. Note that in continual learning, data arrives in a streaming manner, making the temporal distribution at each time point likely imbalanced, even if the overall dataset is balanced across tasks. As a result, these methods often lead to skewed data selection, over-sampling data from certain tasks while under-sampling others.

Method	MICVIT		COAST	
	A_{avg} ↑	A_{last} ↑	A_{avg} ↑	A_{last} ↑
Random	55.66±0.36	57.97±0.61	17.64±0.50	23.24±1.44
GradNorm	54.39±1.59	56.90±0.41	14.94±0.79	22.54±0.31
Self-Sup	53.20±0.71	55.95±0.58	15.22±0.64	19.77±0.30
COINCIDE	55.18±0.16	58.32±0.61	13.87±0.50	21.67±0.15
DBP	53.43±0.63	57.25±1.50	14.20±0.28	18.82±0.20
InfoBatch	54.90±0.46	54.30±0.54	16.74±0.52	23.54±0.35
DivBS	55.52±0.12	57.22±0.17	17.35±0.68	23.57±0.15
TIVE	52.15±1.06	54.12±0.59	15.33±0.22	22.75±1.30
Adapt-∞	54.89±0.53	55.65±0.91	17.58±0.51	22.29±0.19
OASIS (**Ours**)	**57.66±0.17**	**59.84±0.64**	**19.02±0.71**	**25.08±0.59**

Table 13: **Quantitative comparison between online sample selection methods on MICVIT and COAST benchmark under selection ratio 6.25% at QwenVL-0.5B.** Bold indicates the highest performance; underlined results are within the 0.05 t-test significance level.

Method	MICVIT		COAST	
	A_{last} ↑	A_{avg} ↑	A_{last} ↑	A_{avg} ↑
Entropy	62.75±0.84	70.08±0.99	22.53±0.46	30.42±0.57
Perplexity	59.86±1.37	64.49±0.72	20.01±0.73	28.64±0.82
EL2N	60.36±1.44	67.15±0.81	23.28±0.56	31.81±0.60
FI (Ours)	**64.39±0.58**	**71.26±0.72**	**25.67±0.35**	**34.23±0.38**

Table 14: **Comparison of different information metrics.** On MICVIT and COAST benchmarks, we apply different existing information metrics in place of Fisher Information (FI) to OASIS, when selection ratio is 6.25%. Bold indicates the highest performance, and underlined results are within the 0.05 significance level of the t-test.

To quantify this skewness, we measure the difference in the number of selected samples between the most-sampled and least-sampled tasks for each baseline, and summarize the results in Tab. 18. As shown, these baselines exhibit significantly higher skewness than our proposed OASIS, even higher than random selection, ultimately contributing to their lower performance. In contrast, both OASIS and DivBS, the best-performing baseline, achieve much lower selection skewness, indicating a more balanced sample selection across tasks.

A.16 Comparison of the Number of Selected Samples

While baselines select a fixed number of samples per batch, thus selecting the same number of total samples, OASIS dynamically selects samples probabilistically. For fair comparison, we ensure our approach uses comparable or fewer total samples than those selected by other baselines. We summarize the selected samples for each baseline in Tab. 19. Despite using fewer samples, our method outperforms the baselines, as demonstrated in Sec. 4.2.

Benchmark	Order	Method	A_{avg} ↑	A_{last} ↑
MICVIT	ONCHoPDH	Random	61.15±0.34	67.29±0.61
		InfoBatch	60.82±0.75	68.88±1.09
		DivBS	61.07±0.58	69.06±0.54
		OASIS (Ours)	**64.39±0.58**	**71.76±0.72**
	HDPHoCNO	Random	57.85±0.63	66.94±0.74
		InfoBatch	58.04±0.47	67.69±0.80
		DIVBS	58.39±0.28	69.32±0.53
		OASIS (Ours)	**61.87±0.82**	**70.98±0.64**
	PNHOHoCD	Random	60.31±0.88	68.18±0.62
		InfoBatch	60.25±0.35	67.52±0.16
		DIVBS	61.70±0.74	69.16±0.43
		OASIS (Ours)	**63.12±0.58**	**71.31±0.91**
COAST	cdim	Random	23.57±0.17	30.80±0.30
		InfoBatch	22.93±0.73	29.14±0.56
		DivBS	23.41±0.14	31.72±0.18
		OASIS (Ours)	**25.67±0.35**	**34.23±0.38**
	imcd	Random	21.45±0.39	29.25±0.45
		InfoBatch	22.39±0.53	30.04±0.72
		DIVBS	23.84±0.26	31.10±0.56
		OASIS (Ours)	**25.01±0.41**	**33.96±0.53**
	dmci	Random	22.17±0.42	30.64±0.33
		InfoBatch	20.31±0.25	29.53±1.28
		DIVBS	22.92±0.33	32.31±0.74
		OASIS (Ours)	**24.36±0.59**	**33.28±0.26**

Table 15: **Ablation on task order on MICVIT and COAST benchmarks.** In MICVIT, ONCHoPDH, HDPHoCNO, and PNHOHoCD denote task orders of Bongard-OpenWorld → NLVR2 → Co-Instruct-DB, Bongard-HOI → PatternCom → DVQA → HQ Edit, HQ Edit → DVQA → PatternCom → Bongard-HOI → NLVR2 → Bongard-OpenWorld, and PatternCom → NLVR2 → HQ Edit → Bongard-Openworld → Bongard-HOI → Co-Instruct-DB → DVQA, respectively. In COAST, cdim, imcd, and dmci denote task orders of ChartQA → DocVQA → IconQA → MedicalQA, IconQA → MedicalQA → ChartQA → DocVQA, and DocVQA → MedicalQA → ChartQA → IconQA, respectively.

EMA Ratio β	A_{avg} ↑	A_{last} ↑
0.7	22.24±0.25	31.93±0.75
0.9	**24.36±0.59**	**33.28±0.26**
0.99	22.80±0.42	32.73±0.64
0.999	23.05±0.57	32.94±0.81

Table 16: **Ablation on EMA ratio.**

A.17 Detailed Algorithm of OASIS

We provide a comprehensive pseudocode of OASIS in Algorithm 1.

A.18 Impact Statements

In this work, we focus on advancing an online sample selection algorithm for continual instruction tuning. Our approach prioritizes training efficiency while maintaining strong performance, allowing large pre-trained MLLMs to adapt more effectively and make fairer, less biased decisions when pro-

Methods	Overhead Type	Relative $\mathcal{C}$ to OASIS
GradNorm (ICML 2018)	Forward Pass + Last Layer Gradient Compute	1.000
Self-Sup (NeurIPS 2022)	Forward Pass	0.976
COINCIDE (EMNLP 2024)	Forward Pass	0.976
DBP (ICLR 2024)	Forward Pass	0.976
InfoBatch (ICLR 2024)	Forward Pass	0.976
DivBS (ICML 2024)	Forward Pass + Last Layer Gradient Compute	1.000
TIVE (arXiv:2403)	Forward Pass + Full Layer Gradient Compute	2.038
Adapt-∞ (ICLR 2025)	Forward Pass + Middle Layer Gradient Compute	1.507
OASIS (**Ours**)	Forward Pass + Last Layer Gradient Compute	1.000

Table 17: **Comparison of computational costs.** OASIS incurst additional computational cost compared to forward-only baselines (*e.g.*, Self-Sup), but only by approximately 3.4%.

Method	**K-means based selection**	**Max–Min # of selected samples across tasks**
Self-Sup	O	658
COINCIDE	O	659
DBP	O	687
TIVE	O	929
Adapt-∞	O	746
Random	X	600
DivBS	X	520
OASIS (Ours)	X	482

Table 18: **Comparison of data selection distribution.** Comparison of methods with respect to sample variability across tasks.

Method	MICVIT			COAST			Adapt		
	6.25%	12.5%	25.0%	6.25%	12.5%	25.0%	6.25%	12.5%	25.0%
Baselines	43394	86788	173576	10000	20000	40000	40000	80000	160000
OASIS (Ours)	**43158**	**86251**	**173242**	**9874**	**19772**	**39427**	**39589**	**79604**	**159154**

Table 19: **Comparison of the number of selected samples.** We compare the average number of selected samples by OASIS across three different seeds with those selected by other sample selection baselines.

cessing continuous streams of real-world data.

A.19 Data Privacy and Content Sensitivity

For CIT benchmarks, we use existing datasets such as COAST and Adapt, which have already filtered out privacy-sensitive content, as well as MICVIT, a combination of existing multimodal benchmarks that are also free from sensitive content.

A.20 Potential Risks

In CIT setup, real-time data arrives continuously in a streaming manner, leading to imbalanced data distributions at each time step. This can unintentionally introduce bias throughout training.

A.21 Parameters for Packages

For evaluation, we measure accuracy, which selects an answer among multiple candidate choices. In ORIS, we normalize the informativeness score I using Z-score normalization based on our empirical evidence from the QQ-plot.

A.22 License For Artifacts

We utilize publicly available data, models, and codebases, as provided by the original papers for each baseline method.

A.23 Use of AI Assistance

We use AI assistance, such as GPT-4, solely for grammatical error corrections.

Algorithm 1 OASIS

Input: model f_θ, batch $\mathcal{B}_t$, batch size $N_\mathcal{B}$, EMA μ_t, EMV σ_t, threshold I_T, number of layers L
Initialize: Informativeness set $I^{(t)} \leftarrow \emptyset$, high informative sample set $H \leftarrow \emptyset$
// Stage 1: Calculate Informativeness $I^{(t)}$ I of each sample in $\mathcal{B}_t$
for each $(x_i^{(t)}, y_i^{(t)}) \in \mathcal{B}_t$ **do**
    Calcuate gradient $g_i \leftarrow \frac{\partial \ell(d_i^{(t)})}{\partial \theta_L}$
    Calculate Informativeness $I_i^{(t)} \leftarrow \text{tr}(g_i \cdot g_i^\intercal)$
    Add to set $I^{(t)} \leftarrow I^{(t)} \cup \{I_i^{(t)}\}$
end for
// Stage 2: SIREN (Similarity-aware Information Redundancy ElimiNation)
Initialize adjusted Informativeness $\widetilde{I}^{(t)} \leftarrow I^{(t)}$
while $|H| < N_\mathcal{B}$ **do**
    Add most informative sample $H \leftarrow H \cup \left\{\arg\max_{d_i^{(t)} \in \mathcal{B}_t \setminus H} \widetilde{I}_i^{(t)}\right\}$
    for each $d_i^{(t)} \in \mathcal{B}_t \setminus H$ **do**
        for each $d_h^{(t)} \in H$ **do**
            Calculate updated Informativeness $\widetilde{I}_i^{(t)} \leftarrow I_i^{(t)} - \cos(g_i, g_h) \cdot I_h^{(t)}$
        end for
        if $|H| > 1$ **then**
            Account higher-order redundancy due to overlapping similarities between $d_h^{(t)} \in H$
            $\widetilde{I}_i^{(t)} = \widetilde{I}_i^{(t)} + \sum_{\substack{U \subseteq H \\ |U| \geq 2}} (-1)^{|U|} \cos(g_i, \bar{g}_U) \cdot \bar{I}_U^{(t)}$
        end if
    end for
end while
// Stage 3: Calculate Relative Informativeness $\hat{I}^{(t)}$
for each $\widetilde{I}_i^{(t)}$ in $\widetilde{I}^{(t)}$ **do**
    Calculate Relative Informativeness $\hat{I}_i^{(t)} \leftarrow \frac{\widetilde{I}_i^{(t)} - \mu_t}{\sigma_t}$
end for
// Stage 4: Probablistic Sampling
Sample random threshold from uniform distribution $r \sim \mathcal{U}(0, 1)$
Select samples $\mathcal{B}_t^* \leftarrow \{(x_i^{(t)}, y_i^{(t)}) \in \mathcal{B}_t \mid \sigma(\hat{I}_i^{(t)} - I_T^{(t)}) > r\}$
Calculate average Informativeness $\bar{I}^{(t)} = \frac{1}{N_\mathcal{B}} \sum_{i=1}^{N_\mathcal{B}} I_i^{(t)}$
Update EMA $\mu_t = \beta \bar{I}^{(t)} + (1 - \beta)\mu_{t-1}$
Update EMV $\sigma_t = \beta(\bar{I}^{(t)} - \mu_{t-1})^2 + (1 - \beta)\sigma_{t-1}$
Output: selected samples $\mathcal{B}_t^*$