UNSEEN: Enhancing Dataset Pruning from a Generalization Perspective

Furui Xu[1,2*], Shaobo Wang[1,3*], Jiajun Zhang[4], Chenghao Sun[5], Haixiang Tang[6], Linfeng Zhang[1†]

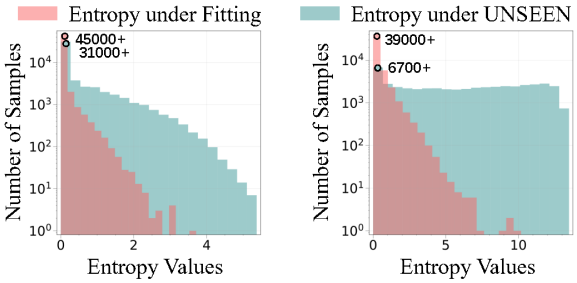
[1]EPIC Lab, Shanghai Jiao Tong University,
[2]East China University of Science and Technology,
[3]Alibaba Group,
[4]Beijing Jiaotong University,
[5]Central South University,
[6]University of Illinois at Urbana-Champaign

Abstract

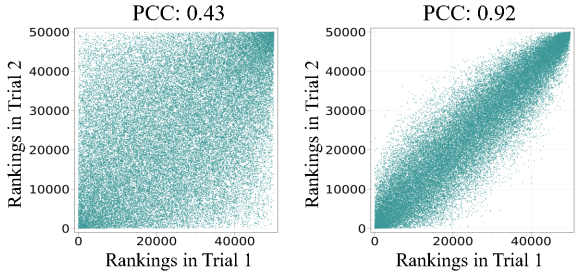
The growing scale of datasets in deep learning has introduced significant computational challenges. Dataset pruning addresses this challenge by constructing a compact but informative coreset from the full dataset with comparable performance. Previous approaches typically establish scoring metrics based on specific criteria to identify representative samples. However, these methods predominantly rely on sample scores obtained from the model's performance during the training (i.e., fitting) phase. As scoring models achieve near-optimal performance on training data, such fitting-centric approaches induce a dense distribution of sample scores within a narrow numerical range. This concentration reduces the distinction between samples and hinders effective selection. To address this challenge, we conduct dataset pruning from the perspective of generalization, i.e., scoring samples based on models not exposed to them during training. We propose a plug-and-play framework, UNSEEN, which can be integrated into existing dataset pruning methods. Additionally, conventional score-based methods are single-step and rely on models trained solely on the complete dataset, providing limited perspective on the importance of samples. To address this limitation, we scale UNSEEN to multi-step scenarios and propose an incremental selection technique through scoring models trained on varying coresets, and optimize the quality of the coreset dynamically. Extensive experiments demonstrate that our method significantly outperforms existing state-of-the-art (SOTA) methods on CIFAR-10, CIFAR-100, and ImageNet-1K. Notably, on ImageNet-1K, UNSEEN achieves lossless performance while reducing training data by 30%.

Introduction

In recent decades, the rapid development of deep learning has been driven by large-scale datasets (Deng et al. 2009; Kuznetsova et al. 2020; Zhou et al. 2017), producing many amazing achievements (Floridi and Chiriatti 2020; Chowdhery et al. 2023; Radford et al. 2021). However, this approach is inherently costly and requires substantial computational resources and considerable time (Cazenavette et al. 2022; Yu, Liu, and Wang 2023; Zhao, Mopuri, and Bilen 2020).



(a) Entropy distribution under the Fitting and UNSEEN frameworks on CIFAR-10 (left) and CIFAR-100 (right) datasets.



(b) Rank distribution of two identical trials with different random seeds under the Fitting (left) and UNSEEN (right) frameworks.

Figure 1: (a) Distribution of Entropy score on CIFAR-10 and CIFAR-100 under the fitting and UNSEEN frameworks. Under the fitting framework, Entropy scores exhibit dense clustering. Conversely, UNSEEN achieves uniform score dispersion, substantially improving discriminative separability. (b) Distribution of the rank assigned to each sample in the overall score ranking in two identical CIFAR-100 trials with different random seeds. Sample ranks fluctuate significantly under fitting but remain stable under UNSEEN. The Pearson correlation coefficient (PCC) between trials is 0.92 for UNSEEN, much higher than 0.43 under fitting.

Moreover, a significant portion of the data is redundant or erroneous (Zhang et al. 2024; Zheng et al. 2022; Xia et al. 2022; Paul, Ganguli, and Dziugaite 2021), contributing minimally to the improvement of model performance. Dataset pruning (Guo, Zhao, and Bai 2022), also known as coreset selection, mitigates data redundancy by constructing a com-

*These authors contributed equally.
†Corresponding author.

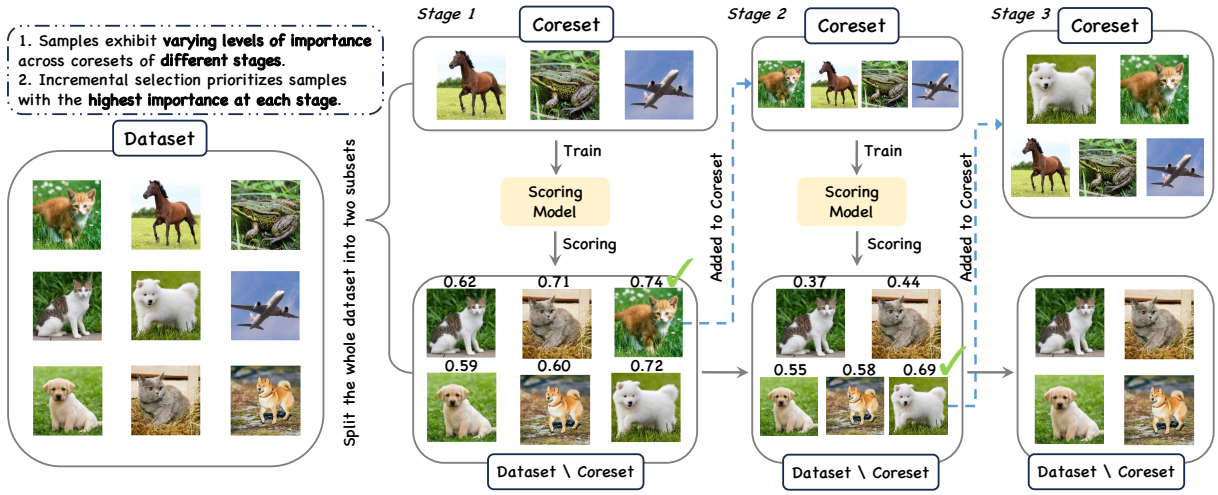


Figure 2: Samples exhibit varying levels of importance across coresets of different stages. Incremental selection prioritizes samples with the highest importance at each stage, offering a more principled and adaptive approach to coreset construction.

pact subset that enables the model to achieve performance comparable to that obtained with the full dataset. This strategy emphasizes identifying and selecting the most informative and representative samples.

Dataset pruning methods commonly identify important samples through various strategies, including geometry features (Xia et al. 2022; Zheng et al. 2022; Wan et al. 2024), uncertainty (Coleman et al. 2019; Pleiss et al. 2020; He et al. 2024; Zhang et al. 2024), error (Paul, Ganguli, and Dziugaite 2021; Toneva et al. 2019), and decision boundary (Margatina et al. 2021; Yang et al. 2024). These methods are based on their scores derived from the model's performance during the training (*i.e.,* fitting) phase. However, as models exhibit a strong capacity for sample fitting, the score distribution becomes densely concentrated. As demonstrated in Figure 1a, Under the conventional fitting framework, more than 90% of CIFAR-10 samples and 78% of CIFAR-100 samples exhibit Entropy values close to zero. Due to the overly clustered score distribution of samples, the model frequently fails to accurately identify challenging samples, and the selection results exhibit severe instability. As shown in Figure 1b, the rank assigned to each sample in the overall score ranking varies significantly between two otherwise identical experiments initialized with different random seeds. This observation indicates that conventional methods under the fitting framework exhibit limitations in effectively and robustly differentiating samples of varying difficulty.

To overcome this challenge, we propose a novel framework called **UNSEEN**, designed to assess the importance of data samples from a generalization perspective. Specifically, we employ cross-validation to facilitate mutual scoring across multiple models, where each sample's importance is determined exclusively by models that never encountered it during training. The implementation involves randomly partitioning the full dataset into mutually exclusive folds of equal size, training scoring models on each fold, and subsequently utilizing these models to score the samples excluded from their respective folds. As shown in Figure 1a, our generalization-based framework UNSEEN produces Entropy scores with more uniform dispersion across a broader scoring range compared to the conventional fitting-based approach. Figure 1b demonstrates that our method significantly enhances robust and discriminative selection.

Additionally, existing score-based methods are single-step *i.e.,* score samples once with the scoring model trained solely on the full dataset. As demonstrated in Figure 2, samples exhibit varying levels of importance when scored by models trained on different coresets. We frame coreset construction as an incremental process, considering the difficulty of samples for models trained on coresets of varying sizes. To comprehensively assess the importance of samples, we scale UNSEEN to multi-step scenarios and propose an evaluate-and-refill paradigm, **incremental selection**. It initiates by constructing an initial coreset through a selection criterion under the fitting or UNSEEN framework. A scoring model is subsequently trained on the coreset and is then employed to score pruned samples. Those that obtain the highest scores are incorporated into the coreset. This procedure persists until the final coreset attains the target cardinality.

Our contributions in this paper are as follows:

- We expose the limitations of previous fitting-based methods and introduce UNSEEN, a plug-and-play framework designed from a generalization perspective.
- We enhance conventional single-step pruning methods by scaling UNSEEN to a multi-step selection process, and propose incremental selection (IS), which provides a more comprehensive assessment of sample importance.
- Our method outperforms existing SOTA methods on CIFAR-10, CIFAR-100, and ImageNet-1K, and achieves 30% lossless pruning on ImageNet-1K.
- We extend the notion of difficulty from individual samples to the class level, prioritizing the minimization of inter-class disparity over uniform treatment across categories.

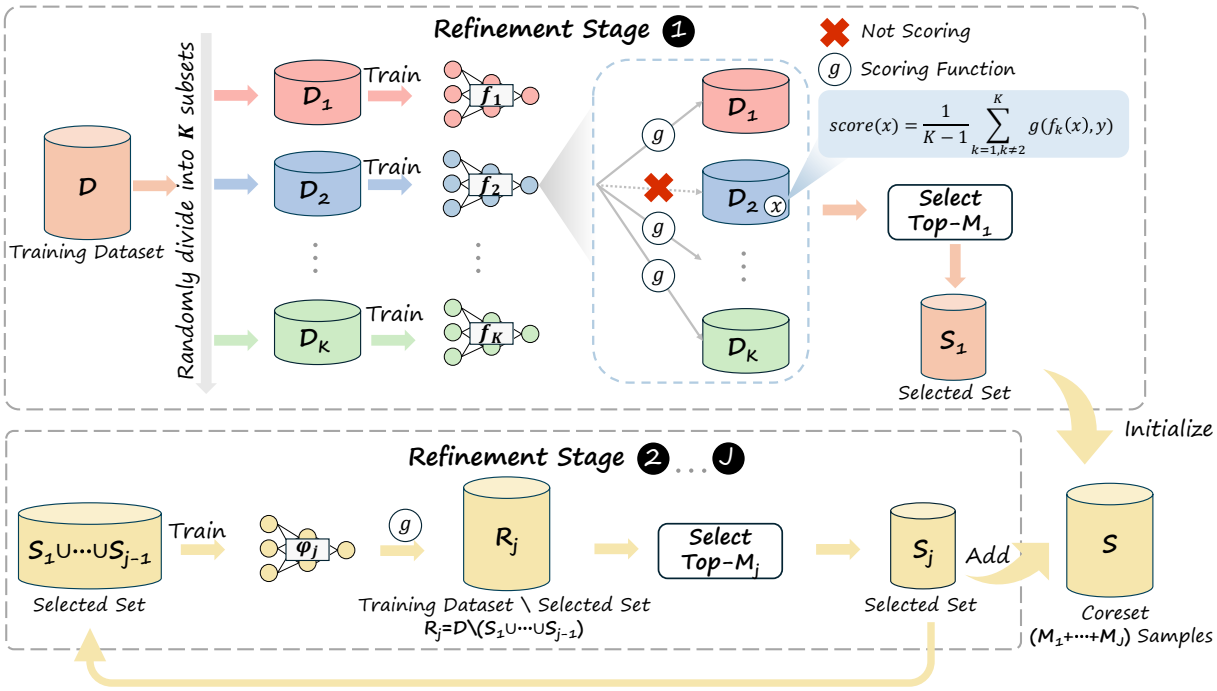


Figure 3: The pipeline of UNSEEN. First, the training dataset is randomly partitioned into K equal-sized subsets. Then, for each subset, a scoring model is trained and used to assign scores to the samples in the complementary subsets. The top M_1 samples with the highest scores are selected to form the initial coreset S_1. Next, a scoring model is trained on the selected samples and used to score the remaining unselected samples. Samples with the highest scores are incrementally added to the coreset. This procedure is repeated until the desired number of samples has been selected.

Related Work

Dataset Pruning. Current approaches to dataset pruning mainly include score-based and optimization-based methods. Emerging as the predominant paradigm, score-based methods select the representative subset by scoring samples based on specific metrics. Various criteria have been developed, such as geometry-based (Welling 2009; Xia et al. 2022; Zheng et al. 2022), uncertainty-based (Coleman et al. 2019; Pleiss et al. 2020; He et al. 2024; Zhang et al. 2024), error-based (Paul, Ganguli, and Dziugaite 2021; Toneva et al. 2019), decision-boundary-based (Margatina et al. 2021; Yang et al. 2024). (I) *Geometry*-based methods leverage spatial distribution characteristics to identify representative samples. Moderate (Xia et al. 2022) employs median distance criteria, retaining samples near the distributional median. CCS (Zheng et al. 2022) selects samples that ensure data coverage. (II) *Uncertainty*-based methods focus on identifying samples with low confidence or high uncertainty of prediction. Entropy (Coleman et al. 2019) selects samples with elevated cross-entropy. DynUnc (He et al. 2024) prioritizes samples with high uncertain prediction. TDDS (Zhang et al. 2024) targets samples with larger projected gradient variances. (III) *Error*-based methods identify and remove data points that demonstrate minimal contributions to model performance. Forgetting (Toneva et al. 2019) retains frequently misclassified samples that exhibit persistent training errors throughout the learning process. Optimization-based methods treat dataset pruning as an optimization problem. Glister (Killamsetty et al. 2021) introduces validation data on the outer optimization and the log-likelihood in the bilevel optimization. While theoretically promising, these methods face practical implementation barriers due to intricate bilevel optimization (Yang et al. 2024; Wang et al. 2025c). Additionally, proxy-based methods (Coleman et al. 2019; Sachdeva, Wu, and McAuley 2021; Wang et al. 2025b) use lightweight or shallow models to fit the training dataset, thus reducing computational cost.

Dataset Synthesis and Dataset Distillation. A parallel and increasingly prominent line of research moves beyond merely filtering existing datasets and instead focuses on actively transforming or generating new, higher-quality data. This paradigm, often termed dataset synthesis or distillation, aims to engineer a more informative and robust training signal than what is available in the raw data distribution (Wang et al. 2025a,d; Liu et al. 2025; Huang et al. 2025).

Methodology

Preliminaries

Consider a classification task with training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ where $x_i \in \mathbb{R}^d$ denotes input features and $y_i \in \{1, \ldots, c\}$ represents class labels for a c-category problem. The data follows an unknown distribution $\mathcal{P}$, and we aim to train a neural network $f_\theta : \mathbb{R}^d \to \mathbb{R}^c$ parameterized by $\theta \in \mathbb{R}^m$. The model minimizes the empirical

risk $\mathcal{L}(\mathcal{D};\theta) = \frac{1}{N}\sum_{i=1}^{N}\ell(f_\theta(x_i), y_i)$, where $\ell : \mathbb{R}^c \times \{1,\ldots,c\} \to \mathbb{R}^+$ is a loss function such as cross-entropy. The dataset pruning objective seeks a coreset $\mathcal{S} \subset \mathcal{D}$ with cardinality $|\mathcal{S}| = M < N$ that preserves model generalization performance. Formally, we require:

$$\mathop{\mathbb{E}}_{\substack{(x,y)\sim\mathcal{P}\\ \theta_0\sim\mathcal{P}_{\theta_0}}}[\ell(f_{\theta_\mathcal{D}}(x), y)] \approx \mathop{\mathbb{E}}_{\substack{(x,y)\sim\mathcal{P}\\ \theta_0\sim\mathcal{P}_{\theta_0}}}[\ell(f_{\theta_\mathcal{S}}(x), y)], \quad (1)$$

where $\theta_\mathcal{D}$ and $\theta_\mathcal{S}$ denote parameters trained on $\mathcal{D}$ and $\mathcal{S}$ respectively, with initialization $\theta_0 \sim \mathcal{P}_{\theta_0}$.

UNSEEN

We implement UNSEEN by cross-validated sample scoring and scale UNSEEN to incremental selection. The detailed pseudo-code of our approach is presented in Algorithm 1.

Cross-validated UNSEEN Scoring. The process initiates by partitioning the original dataset $\mathcal{D}$ into K mutually exclusive subsets $\{\mathcal{D}_k\}_{k=1}^K$ and corresponding complements $\mathcal{D}_k^c = \mathcal{D} \setminus \mathcal{D}_k$ through uniform random sampling, ensuring $\mathcal{D}_i \cap \mathcal{D}_j = \emptyset$ for all $i \neq j$ while maintaining $\bigcup_{k=1}^K \mathcal{D}_k = \mathcal{D}$. For each partition $\mathcal{D}_k$, a neural network f_{θ_k} is trained by optimizing $\theta_k = \arg\min_\theta \mathcal{L}(\mathcal{D}_k;\theta)$. This network subsequently generates scores on the corresponding complement $s(x_i) = \ell(f_{\theta_k}(x_i), y_i)$, $(x_i, y_i) \in \mathcal{D}_k^c$, establishing a cross-validated assessment where models exclusively evaluate samples excluded from their training scenario. We adopt the basic Entropy score as the scoring function.

Incremental Selection (IS). Given a target pruning rate $p \in (0,1)$, the algorithm selects $M = \lfloor N(1-p) \rfloor$ samples through an incremental refinement process over J stages. It begins with score normalization $\tilde{s}(x_i) = s(x_i)/\sum_{j=1}^N s(x_j)$ and sets per-stage selection sizes M_j such that $\sum_{j=1}^J M_j = M$. At stage j, a scoring network is trained on the current coreset $\mathcal{S}j-1$ and assigns loss scores $s_j(x_i) = \ell(f\theta_j(x_i), y_i)$ to samples in the remaining set $\mathcal{R}j-1 = \mathcal{D} \setminus \mathcal{S}j-1$. The top-$M_j$ samples Δ_j with highest scores are added to the coreset, i.e., $\mathcal{S}j = \mathcal{S}j-1 \cup \Delta_j$, while the residual set is updated accordingly. This process continues until the final coreset $\mathcal{S}_J$ reaches size M.

Algorithm 1: UNSEEN-Incremental Selection

Require: Training dataset $\mathcal{D}$ with N samples, target pruning rate $p \in (0,1)$, number of partitions K, number of refinement stages J

Compute target coreset size: $M \leftarrow \lfloor N(1-p) \rfloor$
Define per-stage selection sizes $\{M_j\}_{j=1}^J$ such that $\sum_{j=1}^J M_j = M$
▷ **Cross-Validated UNSEEN Scoring**
Partition $\mathcal{D}$ into K mutually exclusive subsets: $\{\mathcal{D}_k\}_{k=1}^K$, ensuring $\mathcal{D}_i \cap \mathcal{D}_j = \emptyset$ and $\bigcup_{k=1}^K \mathcal{D}_k = \mathcal{D}$
for $k = 1$ to K **do**
    Train model f_{θ_k} on $\mathcal{D}_k$
    for each sample $(x_i, y_i) \in \mathcal{D}_k^c = \mathcal{D} \setminus \mathcal{D}_k$ **do**
        $s(x_i) = \ell(f_{\theta_k}(x_i), y_i)$ ▷ Compute score
    end for
end for
for each sample $x_i \in \mathcal{D}$ **do**
    $\tilde{s}(x_i) = \frac{s(x_i)}{\sum_{j=1}^N s(x_j)}$ ▷ Normalize scores
end for
Initialize coreset: $\mathcal{S}_1 \leftarrow$ **Select**$(\{\tilde{s}(x_i)\}_{x_i\in\mathcal{D}}, M_1)$
Define pruned dataset: $\mathcal{R}_1 \leftarrow \mathcal{D} \setminus \mathcal{S}_1$
▷ **Incremental Selection**
for $j = 2$ to J **do**
    Train model f_{θ_j} on $\mathcal{S}_{j-1}$
    for each sample $(x_i, y_i) \in \mathcal{R}_{j-1}$ **do**
        $s_j(x_i) = \ell(f_{\theta_j}(x_i), y_i)$
    end for
    $\Delta_j \leftarrow$ **Select**$(\{s_j(x_i)\}_{(x_i,y_i)\in\mathcal{R}_{j-1}}, M_j)$ ▷ Select top M_j samples
    Update coreset: $\mathcal{S}_j \leftarrow \mathcal{S}_{j-1} \cup \Delta_j$
    Update pruned dataset: $\mathcal{R}_j \leftarrow \mathcal{R}_{j-1} \setminus \Delta_j$
end for
return Final coreset $\mathcal{S}_J$ and final model f_{θ_J}

Experiments

Datasets and Settings

Datasets and architecture. We evaluated our framework UNSEEN on CIFAR-10/100 (Krizhevsky, Hinton et al. 2009) and ImageNet-1K (Deng et al. 2009). We further evaluated our approach on three challenging fine-grained visual categorization (FGVC) datasets: CUB-2011 (Wah et al. 2011), Stanford Dogs (Khosla et al. 2011), and Stanford Cars (Krause et al. 2013). To validate cross-architecture generalization, we implement dataset pruning on CIFAR-10 and CIFAR-100 using ResNet-18 (He et al. 2016) as the backbone, followed by evaluation of coreset transfer performance on ResNet-34 and ResNet-50.

Baseline. We compare our approach against eleven baselines, including Random, Entropy (Coleman et al. 2019), Margin (Coleman et al. 2019), Least Confidence (Coleman et al. 2019), EL2N (Paul, Ganguli, and Dziugaite 2021), AUM (Pleiss et al. 2020), Forgetting (Toneva et al. 2019), Moderate (Xia et al. 2022), CCS (Zheng et al. 2022), DynUnc (He et al. 2024), TDDS (Zhang et al. 2024).

Implementation details. We set $K = 4$ and $J = 2$, *i.e.*, only one additional pruning stage, with $M_2 = \lceil N \cdot 10\% \rceil$ as the default, consistent across all pruning rates. All experimental results are averaged over five runs. Other details are in the supplementary materials.

Experiments Results

Performance of general image classification. UNSEEN outperformed previous dataset pruning methods on CIFAR-10, CIFAR-100, and ImageNet-1K at varying pruning rates. As shown in Tables 1 and 2, UNSEEN achieved 73.55% accuracy when pruning 30% samples of ImageNet-1K, with only a 0.06% accuracy gap compared to training on full data.

UNSEEN can be applied to existing methods. Figure 4 showed the performance of random dataset pruning, two classical methods (Margin and Least Confidence), and the previous SOTA method TDDS. We applied UNSEEN to the two classical methods. With UNSEEN, they significantly outperformed baselines and approached TDDS per-

Dataset	CIFAR-10					CIFAR-100				
Prune Rate	30%	40%	50%	60%	70%	30%	40%	50%	60%	70%
Random	94.67	94.15	93.27	92.49	91.04	76.00	74.32	72.37	69.87	66.26
Entropy	94.77$_{\uparrow 0.10}$	94.37$_{\uparrow 0.22}$	93.87$_{\uparrow 0.60}$	92.76$_{\uparrow 0.27}$	90.83$_{\downarrow 0.21}$	76.73$_{\uparrow 0.73}$	74.94$_{\uparrow 0.62}$	72.21$_{\downarrow 0.16}$	68.41$_{\downarrow 1.46}$	62.47$_{\downarrow 3.79}$
Margin	94.83$_{\uparrow 0.16}$	94.45$_{\uparrow 0.30}$	93.75$_{\uparrow 0.48}$	92.76$_{\uparrow 0.27}$	90.35$_{\downarrow 0.69}$	76.64$_{\uparrow 0.64}$	74.71$_{\uparrow 0.39}$	71.95$_{\downarrow 0.42}$	67.96$_{\downarrow 1.91}$	62.15$_{\downarrow 4.11}$
Least Confidence	94.11$_{\downarrow 0.56}$	93.51$_{\downarrow 0.64}$	93.08$_{\downarrow 0.19}$	91.63$_{\downarrow 0.86}$	90.09$_{\downarrow 0.95}$	76.54$_{\uparrow 0.54}$	74.79$_{\uparrow 0.47}$	72.66$_{\uparrow 0.29}$	69.00$_{\downarrow 0.87}$	63.92$_{\downarrow 2.34}$
AUM	95.49$_{\uparrow 0.82}$	95.46$_{\uparrow 1.31}$	95.22$_{\uparrow 1.95}$	94.90$_{\uparrow 2.41}$	92.44$_{\uparrow 1.40}$	77.98$_{\uparrow 1.98}$	75.41$_{\uparrow 1.09}$	68.86$_{\downarrow 3.51}$	56.42$_{\downarrow 13.45}$	38.54$_{\downarrow 27.72}$
EL2N	95.36$_{\uparrow 0.69}$	95.27$_{\uparrow 1.12}$	95.10$_{\uparrow 1.83}$	94.58$_{\uparrow 2.09}$	91.10$_{\uparrow 0.06}$	77.44$_{\uparrow 1.74}$	74.47$_{\uparrow 0.15}$	66.75$_{\downarrow 5.62}$	52.81$_{\downarrow 17.06}$	35.03$_{\downarrow 31.23}$
Forgetting	95.48$_{\uparrow 0.81}$	95.47$_{\uparrow 1.32}$	95.34$_{\uparrow 2.07}$	95.03$_{\uparrow 2.54}$	93.22$_{\uparrow 2.18}$	78.16$_{\uparrow 2.16}$	76.01$_{\uparrow 1.69}$	71.74$_{\downarrow 0.63}$	62.42$_{\downarrow 7.45}$	48.48$_{\downarrow 17.78}$
CCS	95.45$_{\uparrow 0.78}$	95.38$_{\uparrow 1.23}$	95.14$_{\uparrow 1.87}$	94.54$_{\uparrow 2.05}$	91.77$_{\uparrow 0.73}$	75.66$_{\downarrow 0.34}$	73.83$_{\downarrow 0.49}$	70.49$_{\downarrow 1.88}$	66.74$_{\downarrow 3.13}$	60.81$_{\downarrow 5.45}$
DynUnc	95.49$_{\uparrow 0.82}$	95.45$_{\uparrow 1.30}$	95.28$_{\uparrow 2.01}$	94.68$_{\uparrow 2.19}$	93.26$_{\uparrow 2.22}$	76.13$_{\uparrow 0.13}$	74.72$_{\uparrow 0.40}$	72.32$_{\downarrow 0.05}$	70.04$_{\uparrow 0.17}$	66.73$_{\uparrow 0.47}$
TDDS	95.49$_{\uparrow 0.82}$	95.49$_{\uparrow 1.34}$	95.30$_{\uparrow 2.03}$	94.66$_{\uparrow 2.17}$	93.51$_{\uparrow 2.47}$	77.25$_{\uparrow 1.25}$	76.00$_{\uparrow 1.68}$	74.09$_{\uparrow 1.72}$	71.91$_{\uparrow 2.04}$	68.38$_{\uparrow 2.12}$
Moderate	94.37$_{\downarrow 0.30}$	93.89$_{\downarrow 0.26}$	93.22$_{\downarrow 0.05}$	92.42$_{\downarrow 0.07}$	90.86$_{\downarrow 0.18}$	76.06$_{\uparrow 0.06}$	74.74$_{\uparrow 0.42}$	72.84$_{\uparrow 0.47}$	70.74$_{\uparrow 0.87}$	66.38$_{\uparrow 0.12}$
UNSEEN	**95.59**$_{\uparrow 0.92}$	**95.58**$_{\uparrow 1.43}$	**95.35**$_{\uparrow 2.08}$	**95.10**$_{\uparrow 2.61}$	**94.16**$_{\uparrow 3.12}$	**78.61**$_{\uparrow 2.61}$	**76.88**$_{\uparrow 2.56}$	**75.15**$_{\uparrow 2.78}$	**71.99**$_{\uparrow 2.12}$	**68.49**$_{\uparrow 2.23}$

Table 1: Comprehensive comparison on CIFAR-10 and CIFAR-100 datasets with ResNet-18. The accuracy on the full dataset of CIFAR-10 and CIFAR-100 is 95.50% and 79.24%. UNSEEN outperforms full-data training when pruning 30% of CIFAR-10 samples. For CIFAR-100, it prunes 30% of training samples with only a 0.63% accuracy drop.

Dataset	ImageNet-1K		
Prune Rate	30%	50%	70%
Random	72.16	71.07	70.00
Entropy	72.72$_{\uparrow 0.56}$	70.93$_{\downarrow 0.14}$	67.55$_{\downarrow 2.45}$
Margin	72.10$_{\downarrow 0.06}$	70.93$_{\downarrow 0.14}$	67.58$_{\downarrow 2.42}$
Least Confidence	72.32$_{\uparrow 0.16}$	71.18$_{\uparrow 0.11}$	67.47$_{\downarrow 2.53}$
AUM	72.96$_{\uparrow 0.80}$	67.47$_{\downarrow 3.60}$	44.58$_{\downarrow 25.42}$
EL2N	72.65$_{\uparrow 0.49}$	69.61$_{\downarrow 1.46}$	62.78$_{\downarrow 7.22}$
Forgetting	71.96$_{\downarrow 0.20}$	70.26$_{\downarrow 0.81}$	68.14$_{\downarrow 1.86}$
Moderate	72.47$_{\uparrow 0.31}$	70.94$_{\downarrow 0.13}$	67.42$_{\downarrow 2.58}$
CCS	71.65$_{\downarrow 0.51}$	70.09$_{\downarrow 0.98}$	66.30$_{\downarrow 3.70}$
DynUnc	70.28$_{\downarrow 1.88}$	66.39$_{\downarrow 4.68}$	60.69$_{\downarrow 9.31}$
TDDS	71.47$_{\downarrow 0.69}$	68.91$_{\downarrow 2.16}$	64.52$_{\downarrow 5.48}$
UNSEEN	**73.55**$_{\uparrow 1.39}$	**72.13**$_{\uparrow 1.06}$	**70.26**$_{\uparrow 0.26}$

Table 2: Comprehensive Comparison on ImageNet-1K. The accuracy on the full dataset is 73.61%. UNSEEN prunes 30% of the data with only a 0.06% accuracy drop.

formance. For example, Margin with UNSEEN and Least Confidence with UNSEEN achieve accuracy of 78.19% and 78.17% at a pruning ratio of 30% on CIFAR-100, respectively, demonstrating that UNSEEN is a plug-and-play framework that can be incorporated into existing methods.

Performance of fine-grained image classification. We also applied UNSEEN to datasets with subtle image differences. We conduct experiments on CUB-2011, Stanford Dogs, and Stanford Cars. As illustrated in Figure 5, our method constructs informative coresets during fine-tuning across all three datasets, demonstrating UNSEEN's effectiveness in selecting fine-grained samples with minimal variations.

Cross-architecture generalization. We validated the applicability of pruned datasets on larger, unseen network architectures not involved in pruning. Specifically, we pruned CIFAR-10 and CIFAR-100 with ResNet-18, and trained the obtained coreset with ResNet-34 and ResNet-50. As shown in Figure 6, our method significantly surpasses other methods, demonstrating that our method generates a coreset with remarkable cross-architecture generalization capabilities.

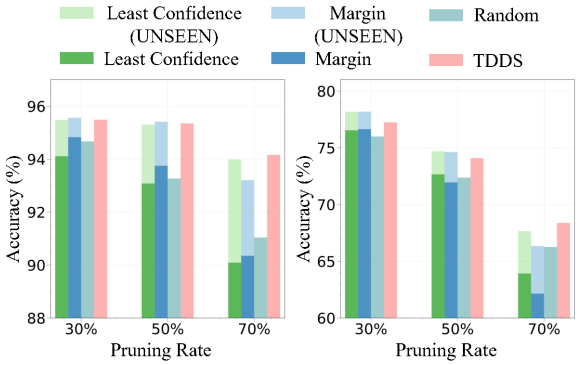


Figure 4: Plug-and-play enhancement of UNSEEN on CIFAR-10 (left) and CIFAR-100 (right). Margin and Least Confidence achieve significant enhancement with UNSEEN and outperform TDDS at low pruning rates.

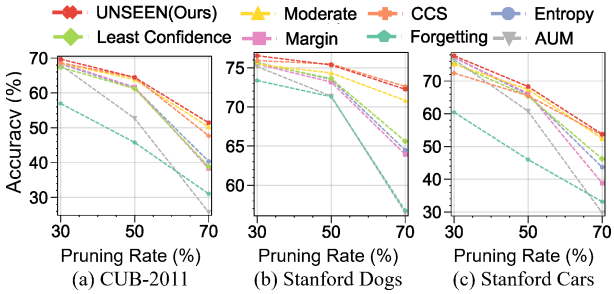
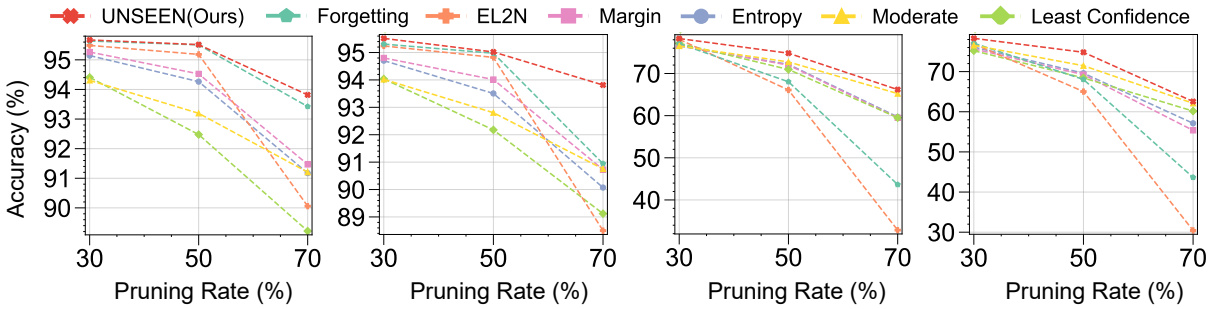


Figure 5: Comparison on fine-grained datasets.



(a) 18 →34 on CIFAR-10 (b) 18 →50 on CIFAR-10 (c) 18 →34 on CIFAR-100 (d) 18 →50 on CIFAR-100

Figure 6: We employed ResNet-18 to perform dataset pruning, and subsequently trained ResNet-34 and ResNet-50 on the pruned datasets. Results demonstrate that the coreset selected by UNSEEN exhibits strong generalization across architectures.

Dataset		**CIFAR-10**			**CIFAR-100**		
UNSEEN	IS	30%	50%	70%	30%	50%	70%
×	×	94.77	93.87	90.83	76.73	72.21	62.47
✓	×	95.39	95.33	94.05	78.24	74.56	62.23
×	✓	95.27	94.82	92.96	76.92	74.31	65.73
✓	✓	**95.59**	**95.35**	**94.16**	**78.61**	**75.15**	**68.49**

Table 3: Ablation study on UNSEEN and Incremental Selection (IS). Both UNSEEN and IS improve the Entropy method, with the optimal results when combined.

Dataset	**CIFAR-10**			**CIFAR-100**		
Prune Rate	30%	50%	70%	30%	50%	70%
UNSEEN ($K = 2$)	95.37	94.81	92.82	77.87	74.51	64.53
UNSEEN ($K = 4$)	**95.39**	**95.34**	**94.05**	**78.24**	**74.56**	**67.90**
UNSEEN ($K = 10$)	95.37	95.24	93.28	76.98	73.45	67.52
UNSEEN ($K = 20$)	95.29	94.51	92.89	76.23	72.11	66.41

Table 4: Ablation study on the number of folders K. UNSEEN yields optimal performance for moderate K values.

Ablation Study

Ablation study on UNSEEN and IS.

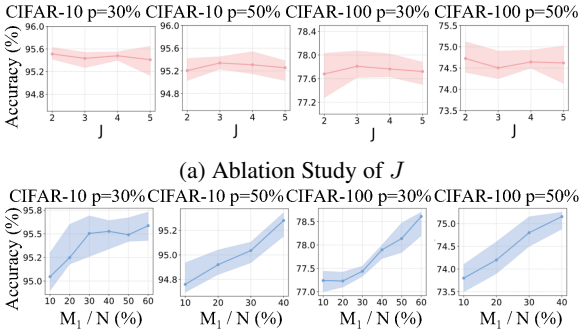
We adopted Entropy as the baseline, then integrated UNSEEN and IS separately. As shown in Table 3, they both exhibit significant improvements, with UNSEEN demonstrating a more pronounced enhancement. The optimal result is achieved when combined.

Ablation study on the number of partitions K. Results showed that models with moderate capability achieve optimal pruning. As analyzed in Table 4, overly strong models (*e.g.*, with extremely small K) tend to fit samples precisely, reducing the discrimination for the difficulty of samples. Weaker models (*e.g.*, with excessively large K) struggle to effectively capture fundamental class characteristics, resulting in suboptimal differentiation.

Abation study on generalization-based scoring. Scoring models of UNSEEN can be viewed as proxy models trained on reduced data. We conducted experiments to validate the effectiveness of UNSEEN from a generalization perspective.

Dataset	**CIFAR-10**			**CIFAR-100**		
Prune Rate	30%	50%	70%	30%	50%	70%
Entropy	94.77	93.87	90.83	76.73	72.21	62.47
Entropy-proxy ($K = 4$)	94.85	94.04	92.04	76.83	73.22	65.08
UNSEEN ($K = 4$)	**95.59**	**95.35**	**94.16**	**78.61**	**75.15**	**67.90**

Table 5: Ablation study on generalization-based scoring. UNSEEN significantly outperforms Entropy-proxy, demonstrating its advantages from a generalization perspective.



(a) Ablation Study of J

(b) Ablation Study of M (J = 2)

Figure 7: Ablation studies of hyperparameters J and M.

For the proxy model based on Entropy, the dataset was randomly partitioned into K equally sized subsets, and K scoring models were trained on these subsets to assign scores to samples within each subset (*i.e.,* Entropy-proxy). We set K = 4, the same as the experimental setting in UNSEEN. Table 5 compares the results of dataset pruning among the original Entropy, Entropy-proxy, and UNSEEN. UNSEEN significantly outperforms Entropy-proxy, demonstrating its advantages from a generalization perspective.

Ablation study for the number of refinement stages J **and per-stage coreset size** M. As shown in Figure 7a, although the cost grows with increasing J, the performance remains nearly unchanged. We also conducted the ablation study on the first-stage coreset size M_1 ($\lceil N \cdot (1-p) \rceil - M_2$) using CIFAR-10 and CIFAR-100 when $J = 2$. Figure 7b

shows that a larger first-stage coreset, supplemented by a smaller second-stage retraining, yields better performance.

Discussion

Computational Cost of UNSEEN and IS

UNSEEN does not introduce any additional cost, while incremental selection incurs only minimal overhead.
(i) Computational Overhead of Previous Methods: Given a batch size of B, previous methods incur N/B iterations to train a scoring model on the full dataset. After pruning p of the data, the model is trained on the selected coreset with $N \cdot (1-p)/B$ iterations. Thus, the total computational overhead of previous methods is $N \cdot (2-p)/B$ iterations.
(ii) Computational Overhead of UNSEEN: UNSEEN trains K scoring models on K equal-sized subsets. Each requires $N/(K \cdot B)$ iterations, summing to N/B iterations. Training on the coreset adds $N \cdot (1-p)/B$ iterations, totaling $N \cdot (2-p)/B$ iterations, which is *equivalent to that of previous methods*. Hence, **UNSEEN does not introduce any additional computational cost.**
(iii) Computational Overhead of Incremental Selection (IS): We set $J = 2$ and $M_2 = \lceil N \cdot 10\% \rceil$, *i.e.*, 10% of the full dataset is incrementally selected in the second step. An additional model is trained before the second stage, incurring $N(1-p-10\%)/B$ iterations. For instance, **when pruning 70% of the data, IS incurs only $0.15\times$ the total cost required by previous methods** (see figure below).

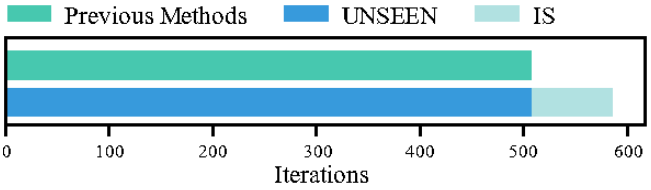
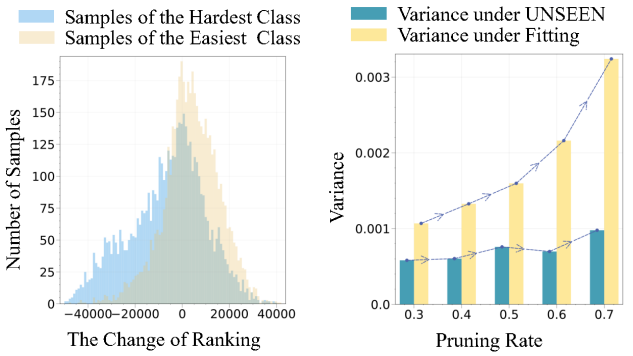


Figure 8: Comparison of Computational Overhead among Previous Methods, UNSEEN, and IS at 70% Pruning Rate.

Class-Level Analysis of UNSEEN

This section explains UNSEEN's strong performance by analyzing dataset pruning at the class level. Given that different classes inherently possess varying levels of complexity, models tend to prioritize learning simpler classes, leading to a lower accuracy for challenging ones and consequently creating an imbalance in performance (Cui et al. 2024). Most existing pruning methods either ignore class information or treat all classes uniformly (Guo, Zhao, and Bai 2022) (*i.e.,* selecting the same number of samples from each class).

Since the scoring models under UNSEEN are not exposed to training samples, they implicitly incorporate amplified class-level difficulty weighting, which leads to a prioritization of hard-class samples. To verify this hypothesis, we selected all samples from the easiest and the hardest classes in the full dataset and plotted the change in their score rankings when transitioning from the fitting framework to UNSEEN (i.e., the difference in rankings between



(a) Ranking Change (b) Inter-class Var Comparison

Figure 9: (a) We computed the differential ranking by subtracting the fitting rankings from the UNSEEN rankings. We observed that most samples in the hardest class experienced rank drops (prior to being selected in UNSEEN), while the easiest class exhibited the opposite trend. (b) We measured variance in classification accuracies across classes (*i.e.,* inter-class variance). With increasing pruning rates, the Entropy method under the fitting framework showed rapid growth in inter-class variance, whereas UNSEEN maintained significantly stable and lower inter-class variance.

the two frameworks). Lower rankings indicate higher scores and greater selection priority. As demonstrated in Figure 9a, a majority of samples from the hardest class exhibited negative differences with lower magnitudes, indicating their increased selection priority under the UNSEEN framework. In contrast, samples from the easiest class exhibited a contrasting pattern with positive differences. The prioritization of hard-class instances facilitates mitigating inter-class discrepancy within the coreset, thereby achieving holistic performance optimization through balanced representation learning. As shown in Figure 9b, we observe that with increasing pruning rates, the inter-class variance *i.e.,* variance of accuracies across different classes, increases rapidly under the fitting framework, while the inter-class variance is stably lower under the UNSEEN framework. The experimental result demonstrates that minimizing the accuracy disparity between classes can enhance overall performance.

Conclusion

In this paper, we identify that existing dataset pruning methods under the fitting framework yield highly dense scores, leading to undiscriminating and unstable selection. Therefore, we introduce a plug-and-play framework, UNSEEN, from the perspective of generalization. To refine the previous single-step pruning method, we scale UNSEEN to multi-step scenarios and propose incremental selection to evaluate samples comprehensively and optimize the coreset dynamically. Finally, we analyze the reason for UNSEEN's outstanding performance by prioritizing samples of hard classes and extend the concept of difficulty from samples to classes. Experiments demonstrate that minimizing inter-class disparity is critical for achieving exceptional performance.

Acknowledgments

This work was supported by Alibaba Group through Alibaba Innovative Research Program, and Shanghai Science and Technology Program (Grant No. 25ZR1402278).

References

Cazenavette, G.; Wang, T.; Torralba, A.; Efros, A. A.; and Zhu, J.-Y. 2022. Dataset distillation by matching training trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4750–4759.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113.

Coleman, C.; Yeh, C.; Mussmann, S.; Mirzasoleiman, B.; Bailis, P.; Liang, P.; Leskovec, J.; and Zaharia, M. 2019. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*.

Cui, J.; Zhu, B.; Wen, X.; Qi, X.; Yu, B.; and Zhang, H. 2024. Classes Are Not Equal: An Empirical Study on Image Recognition Fairness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23283–23292.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Floridi, L.; and Chiriatti, M. 2020. GPT-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30: 681–694.

Guo, C.; Zhao, B.; and Bai, Y. 2022. Deepcore: A comprehensive library for coreset selection in deep learning. In *International Conference on Database and Expert Systems Applications*, 181–195. Springer.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

He, M.; Yang, S.; Huang, T.; and Zhao, B. 2024. Large-scale dataset pruning with dynamic uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7713–7722.

Huang, C.; Yu, W.; Wang, X.; Zhang, H.; Li, Z.; Li, R.; Huang, J.; Mi, H.; and Yu, D. 2025. R-Zero: Self-Evolving Reasoning LLM from Zero Data. *arXiv preprint arXiv:2508.05004*.

Khosla, A.; Jayadevaprakash, N.; Yao, B.; and Li, F.-F. 2011. Novel dataset for fine-grained image categorization: Stanford dogs. In *Proc. CVPR workshop on fine-grained visual categorization (FGVC)*, volume 2.

Killamsetty, K.; Sivasubramanian, D.; Ramakrishnan, G.; and Iyer, R. K. 2021. GLISTER: Generalization based Data Subset Selection for Efficient and Robust Learning. In *AAAI*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Kolesnikov, A.; et al. 2020. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7): 1956–1981.

Liu, X.; Wen, Z.; Wang, S.; Chen, J.; Tao, Z.; Wang, Y.; Jin, X.; Zou, C.; Wang, Y.; Liao, C.; et al. 2025. Shifting ai efficiency from model-centric to data-centric compression. *arXiv preprint arXiv:2505.19147*.

Margatina, K.; Vernikos, G.; Barrault, L.; and Aletras, N. 2021. Active learning by acquiring contrastive examples. *arXiv preprint arXiv:2109.03764*.

Paul, M.; Ganguli, S.; and Dziugaite, G. K. 2021. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34: 20596–20607.

Pleiss, G.; Zhang, T.; Elenberg, E.; and Weinberger, K. Q. 2020. Identifying mislabeled data using the area under the margin ranking. *Advances in Neural Information Processing Systems*, 33: 17044–17056.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Sachdeva, N.; Wu, C.-J.; and McAuley, J. 2021. Svp-cf: Selection via proxy for collaborative filtering data. *arXiv preprint arXiv:2107.04984*.

Toneva, M.; Sordoni, A.; des Combes, R. T.; Trischler, A.; Bengio, Y.; and Gordon, G. J. 2019. An Empirical Study of Example Forgetting during Deep Neural Network Learning. In *Int. Conf. Learn. Represent.*

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wan, Z.; Wang, Z.; Wang, Y.; Wang, Z.; Zhu, H.; and Satoh, S. 2024. Contributing Dimension Structure of Deep Feature for Coreset Selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 9080–9088.

Wang, S.; Jiao, Z.; Zhang, Z.; Peng, Y.; Ze, X.; Yang, B.; Wang, W.; Wei, H.; and Zhang, L. 2025a. Socratic-Zero: Bootstrapping Reasoning via Data-Free Agent Co-evolution. *arXiv preprint arXiv:2509.24726*.

Wang, S.; Jin, X.; Wang, Z.; Wang, J.; Zhang, J.; Li, K.; Wen, Z.; Li, Z.; He, C.; Hu, X.; and Zhang, L. 2025b. Data Whisperer: Efficient Data Selection for Task-Specific LLM Fine-Tuning via Few-Shot In-Context Learning. *Annual Meeting of the Association for Computational Linguistics*.

Wang, S.; Wang, J.; Zhang, J.; Wang, C.; Min, Y.; Wen, Z.; Huang, F.; Jiang, H.; Lin, J.; Liu, D.; et al. 2025c. Winning the pruning gamble: A unified approach to joint sample and token pruning for efficient supervised fine-tuning. *arXiv preprint arXiv:2509.23873*.

Wang, S.; Yang, Y.; Wang, Q.; Li, K.; Zhang, L.; and Yan, J. 2025d. Not All Samples Should Be Utilized Equally: Towards Understanding and Improving Dataset Distillation. *Synthetic Data for Computer Vision Workshop at CVPR*.

Welling, M. 2009. Herding dynamical weights to learn. In *Proceedings of the 26th annual international conference on machine learning*, 1121–1128.

Xia, X.; Liu, J.; Yu, J.; Shen, X.; Han, B.; and Liu, T. 2022. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*.

Yang, S.; Cao, Z.; Guo, S.; Zhang, R.; Luo, P.; Zhang, S.; and Nie, L. 2024. Mind the boundary: Coreset selection via reconstructing the decision boundary. In *Forty-first International Conference on Machine Learning*.

Yu, R.; Liu, S.; and Wang, X. 2023. Dataset distillation: A comprehensive review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhang, X.; Du, J.; Li, Y.; Xie, W.; and Zhou, J. T. 2024. Spanning training progress: Temporal dual-depth scoring (tdds) for enhanced dataset pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26223–26232.

Zhao, B.; Mopuri, K. R.; and Bilen, H. 2020. Dataset condensation with gradient matching. *arXiv preprint arXiv:2006.05929*.

Zheng, H.; Liu, R.; Lai, F.; and Prakash, A. 2022. Coverage-centric coreset selection for high pruning rates. *arXiv preprint arXiv:2210.15809*.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6): 1452–1464.