Contributing Dimension Structure of Deep Feature for Coreset Selection

Zhijing Wan[1,2], Zhixiang Wang[3,4], Yuran Wang[1,2], Zheng Wang[1,2]*, Hongyuan Zhu[5], Shin'ichi Satoh[4,3]

[1]National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence, School of Computer Science, Wuhan University, China
[2]Hubei Key Laboratory of Multimedia and Network Communication Engineering
[3]The University of Tokyo, Japan
[4]National Institute of Informatics, Japan
[5] Institute for Infocomm Research (I2R) & Centre for Frontier AI Research (CFAR), A*STAR, Singapore

Abstract

Coreset selection seeks to choose a subset of crucial training samples for efficient learning. It has gained traction in deep learning, particularly with the surge in training dataset sizes. Sample selection hinges on two main aspects: a sample's representation in enhancing performance and the role of sample diversity in averting overfitting. Existing methods typically measure both the representation and diversity of data based on similarity metrics, such as $L2$-norm. They have capably tackled representation via distribution matching guided by the similarities of features, gradients, or other information between data. However, the results of effectively diverse sample selection are mired in sub-optimality. This is because the similarity metrics usually simply aggregate dimension similarities without acknowledging disparities among the dimensions that significantly contribute to the final similarity. As a result, they fall short of adequately capturing diversity. To address this, we propose a feature-based diversity constraint, compelling the chosen subset to exhibit maximum diversity. Our key lies in the introduction of a novel Contributing Dimension Structure (CDS) metric. Different from similarity metrics that measure the overall similarity of high-dimensional features, our CDS metric considers not only the reduction of redundancy in feature dimensions, but also the difference between dimensions that contribute significantly to the final similarity. We reveal that existing methods tend to favor samples with similar CDS, leading to a reduced variety of CDS types within the coreset and subsequently hindering model performance. In response, we enhance the performance of five classical selection methods by integrating the CDS constraint. Our experiments on three datasets demonstrate the general effectiveness of the proposed method in boosting existing methods.

1 Introduction

Coreset selection is a long-standing learning problem that aims to select a subset of the most informative training samples for data-efficient learning (Das et al. 2021; Wan et al. 2023a). Early coreset selection methods were designed to accelerate the learning and clustering of machine algorithms, such as k-means and k-medians (Har-Peled and Kushal 2007), support vector machines (Tsang et al. 2005),

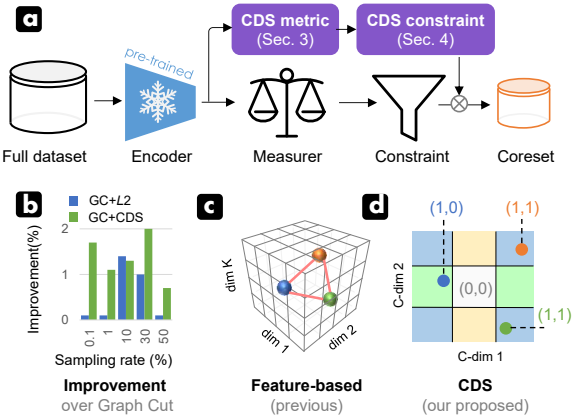


Figure 1: Our method and motivation. (a) We combine the proposed CDS metric and constraint with the current coreset selection pipeline. (b) CDS metric and constraint enhance the performance of SOTA—GC (Iyer et al. 2021). Although replacing the CDS metric with $L2$ distance employed by previous feature-based methods can improve GC, integrating our proposed CDS metric is more effective since it can capture more diverse, informative samples. (c) Previous feature-based methods using $L2$ metric could treat three distinct samples as equivalent, while (d) our CDS metric effectively distinguishes these samples by pruning the feature space and representing the space in different partitions. Note that here, we set the pruned dimension (C-dim) to 2 for demonstration.

and Bayesian inference (Campbell and Broderick 2018). However, they are designed for specific models and problems, and have limited applications in deep learning.

Recently, with the rapid development of deep learning (Xie et al. 2023; Yang et al. 2023; Yuan et al. 2023), research on coreset selection for deep learning has emerged, including geometry-based methods (Sener and Savarese 2017; Agarwal et al. 2020), uncertainty-based methods (Coleman et al. 2019), submodularity-based methods (Kothawade et al. 2022; Rangwani et al. 2021), gradient matching-based methods (Mirzasoleiman, Bilmes, and Leskovec 2020; Killamsetty et al. 2021a) and others (Toneva et al. 2018; Swayamdipta et al. 2020). They typically rely on a pre-trained model to obtain information, *e.g.,* features,

*Corresponding Author

gradients and predicted probabilities, for measuring the importance of data in the full training set through designed measurer and constraint. Budget-sized samples will then be selected based on importance to form a coreset that can be used for data-efficient machine learning (Pooladzandi, Davini, and Mirzasoleiman 2022), compute-efficient hyperparameter tuning (Killamsetty et al. 2022), continual learning (Tiwari et al. 2022; Yoon et al. 2021), *etc*.

The importance of data in coreset selection is assessed in two main aspects, that is, representation and diversity. Representation is guaranteed through the distribution matching between subsets and the full set under the guidance of similarities between data in features, gradients, or other information. Meanwhile, diversity is assured through the imposition of penalties upon similar data. Information similarity is commonly computed by similarity metrics, such as $L2$-norm and cosine distance. However, these similarity metrics obtain the final similarity by simply aggregating dimension similarities without evaluating the impact of different dimensions. This risks treating some distinct samples as equally important, leading to an ineffective assessment of the diversity. For example, in the case of feature-based selection methods, there is redundancy in the feature dimensions extracted by the pre-trained model (Li et al. 2017), and similarity metrics directly add a lot of useless information, which is not conducive to measuring the diversity of data. Besides, and most importantly, the data dimensions that contribute significantly to the final similarity are different for different data (such dimensions are called *contributing dimensions*), and this difference is also ignored in the similarity metrics. Our empirical study further shows that similarity metrics tend to select a larger number of samples whose features have the same Contributing Dimension Structure (CDS) for coreset selection, resulting in fewer types of CDS in the coreset, which inhibits model performance. Therefore, it is urgent to design a metric that can evaluate the impact of different dimensions of the data information.

In this paper, we draw inspiration from the above observation and consequently design a Contributing Dimension Structure (CDS) metric and a feature-based CDS diversity constraint (abbreviated as CDS constraint) to improve the current coreset selection pipeline, as shown in Figure 1(a), to compel the selected subset to showcase maximum diversity. Firstly, we propose a CDS metric to select helpful feature dimensions and divide the pruned feature space into different partitions, obtaining the CDS of each data. Throughout our paper, CDS is defined as an indication of whether each dimension of the pruned feature contributes significantly to the overall similarity measure. Each dimension is analyzed in turn, with 1 indicating that the dimension contributes and 0 indicating that it does not. By comparing the CDS of different data, we can classify them as having the same CDS or different CDS. Afterward, an effective strategy *CDS constraint* is proposed to enrich the diversity of CDS in subsets.

The main contribution of our work is threefold:

- For the first time in coreset selection, we explicitly introduce information on the Contributing Dimension Structure (CDS) via the proposed *CDS metric* to enrich the diversity of CDS in the coreset.
- *CDS constraint*, which aims to constrain the selected subset to have as many different CDS as possible, is proposed to improve existing SOTA methods. We propose two implementations of the CDS constraint, namely the Hard CDS Constraint and the Soft CDS Constraint, and apply them to five classical coreset selection methods.
- Extensive experiments on three image classification datasets with two data sampling modes (class-balanced sampling and class-imbalanced sampling) show that our method can effectively improve SOTA methods.

2 Problem Formulation

Coreset Selection We focus on the traditional computer vision task of image classification. In an image classification task with C classes, we work with a sizable training set $\mathcal{D} = (x_i, y_i)_{i=1}^n$ defined across a joint distribution $\mathcal{X} \times \mathcal{Y}$, where n denotes the quantity of training data, $\mathcal{X}$ pertains the input space and $\mathcal{Y}$ is the label space $\{1, \ldots, C\}$. In scenarios where there's a specified budget b, coreset selection aims to select a subset $\mathcal{S} \subset \mathcal{D}$ containing the most informative training samples. This is done with the intention that the model $\theta^{\mathcal{S}}$ trained on $\mathcal{S}$ can achieve performance comparable to that of the model $\theta^{\mathcal{D}}$ trained on the full training set $\mathcal{D}$. The size of the subset $|\mathcal{S}| = b$, where $b < n$. It is common to convert the coreset selection problem into the design of a monotonic objective function T and to find the optimal subset

$$\mathcal{S}^* = \underset{\mathcal{S} \subset \mathcal{D}}{\arg\max}\, T(\mathcal{S}) \quad \text{s.t.} \quad |\mathcal{S}| \leq b, \tag{1}$$

where budget-sized $\mathcal{S}^*$ is selected before training, with the expectation that the accuracy of models trained on this subset will be maximized.

Representativeness and diversity of subsets are the two main factors that coreset selection focuses on when measuring the importance of data. Most existing coreset selection methods usually design and implement the objective function based on similarity metrics (*e.g.,* $L2$-norm). They ensure representativeness through distribution matching guided by similarities of features, gradients, or other information between data, while pursuing diversity by penalising data with similar information. However, conventional similarity metrics simply aggregate dimension similarities without acknowledging disparities among the dimensions that significantly contribute to the final similarity. As a result, they fall short of adequately capturing diversity. We will dissect the problems in the following.

Diversity Measurement Informative and easily accessible deep features are often adopted in selection methods (Guo, Zhao, and Bai 2022; Margatina et al. 2021; Wan et al. 2023b, 2022), which typically use the overall similarity metrics to calculate the similarity between deep features to measure the importance of the data, such as $L1$-norm, $L2$-norm and cosine distance metric. Figure 2 illustrates their characteristics. For convenience, we focus on the most commonly adopted $L2$-norm as an example. It can be formulated as:

$$d\left(\boldsymbol{F}(x_i), \boldsymbol{F}(x_j)\right) = \sqrt{\sum_{k=0}^{K-1} \left(f_i^k - f_j^k\right)^2}, \tag{2}$$

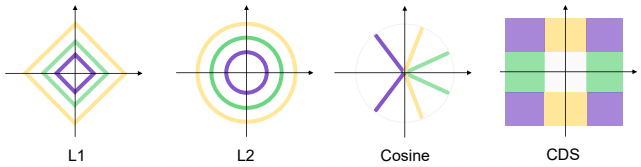


Figure 2: Different metrics. $L1$ and $L2$ quantify the magnitude between samples, while the Cosine distance evaluates the direction between samples. In contrast, our introduced CDS metric evaluates the impact of different dimensions, making it an effective tool for assessing diversity. Note that the reference sample is positioned at the origin for $L1$, $L2$, and CDS, while it is located along the positive horizontal axis for Cosine distance. Regions with the same color indicate the same score.

where $\boldsymbol{F}(x_i) = \left[f_i^0, f_i^1, \ldots, f_i^{K-1}\right] \in \mathbb{R}^K$ denotes the feature of sample x_i extracted by the model, and K means the number of dimensions of the deep feature.

Unfortunately, the calculation of the similarity metric shown in Equation 2 does not explicitly reflect the numerical difference between $\boldsymbol{F}(x_i)$ and $\boldsymbol{F}(x_j)$ in *each* dimension. This may lead to an ineffective assessment of the diversity of the data, making the selection of the coreset insufficiently diverse. For example, there is a phenomenon: Given candidate data x_i, x_j, and x_h, when measuring their importance, we compare them with a selected data x_q. If their distance $d\left(\boldsymbol{F}(x_i), \boldsymbol{F}(x_q)\right)$, $d\left(\boldsymbol{F}(x_j), \boldsymbol{F}(x_q)\right)$, and $d\left(\boldsymbol{F}(x_h), \boldsymbol{F}(x_q)\right)$ are close, feature-based methods could treat them as equally important and select them with equal probability. However, when analyzed in terms of each dimension difference, there may exist a dimension $\ddot{k}$ that makes the difference between x_i and x_q equals zero, while that between x_j and x_q, x_h and x_q are significantly larger than zero, *i.e.,*

$$|f_i^{\ddot{k}} - f_q^{\ddot{k}}| \to 0\,, \quad |f_j^{\ddot{k}} - f_q^{\ddot{k}}| \gg 0\,, \quad |f_h^{\ddot{k}} - f_q^{\ddot{k}}| \gg 0\,.$$

This can be interpreted as the $\ddot{k}^{th}$ dimension of data x_i does not contribute in calculating the overall similarity with x_q, while the $\ddot{k}^{th}$ dimension of data x_j and data x_h do. In other words, x_i has a different contributing dimension structure (CDS) from x_j and x_h. At this point, the CDS diversity of x_i is different from that of x_j and x_h while their diversities calculated by $L2$ are the same.

It is natural to ask the **question**: what is the relationship between the diversity of the CDSs in the selected samples and the performance of the models trained on those selected samples? In light of this question, we introduce a metric to measure the CDS of each data in the next section.

3 CDS Metric of Deep Feature

The previous section introduces the concept of CDS. Moving forward, we will introduce the CDS metric in this section. This metric is designed to quantify the CDS of individual samples, as depicted in Figure 3. Additionally, we will delve into an analysis of the connection between CDS diversity and the performance of models.

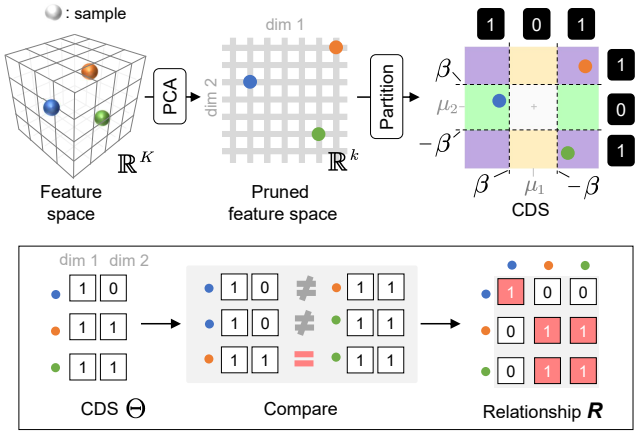


Figure 3: CDS metric for deep features. Given the high-dimensional feature matrix, we first reduce its dimension from K to k using PCA. Then, we compute the central feature $[\mu_0, \mu_1, \ldots, \mu_{k-1}]$ of the dimension reduced feature matrix. Next, we obtain the CDS for each data by comparing the difference between each data feature and the central feature in each dimension with a threshold β to divide the feature space into different partitions. Finally, the CDS relationship matrix $\boldsymbol{R}$ used for the subsequent CDS constraint is obtained by comparing the CDS between each data individually to see if they are the same.

3.1 CDS Metric

Dimension Reduction Inspired by the work (Li et al. 2017), we consider that for the analysis of CDS relationships between data, there is redundancy in the feature dimensions output by the network, *i.e.,* the dimensions that are helpful need to be selected before analysis. Therefore, we first perform dimension selection using the classical PCA algorithm (Pearson 1901) to reduce K to k. The choice of k will be elaborated upon in Section 5.5. After that, two problems lie ahead when it comes to actually analyzing the CDS relationships between the data.

Deviation from the Mean One problem is that the relationship between the contributing dimension structures of the data is relative, *i.e.,* in the above phenomenon, the contributing dimension structure relationship between x_i, x_j and x_h may change as x_q changes to $x_{\dot{q}}$. Therefore in sampling data from each class, we set $\boldsymbol{F}(x_q)$ to be the class prototype, *i.e.,* the central feature $\tilde{\boldsymbol{F}}_c = [\mu_0, \mu_1, \ldots, \mu_{k-1}]$ of the class (Xie et al. 2022), to consistently analyze the relationships of all training data[1] with

$$\mu_{k'} = \left(\sum_{i=0}^{N_c-1} f_i^{k'}\right) / N_c\,, \quad k' \in \{0, 1, \ldots, k-1\}\,.$$

Then for each data point within class c, its deviation from the central feature $\tilde{\boldsymbol{F}}_c$ in each dimension can be expressed

[1]For class-imbalanced sampling settings, $\boldsymbol{F}(x_q)$ can be set to the central feature of the dataset.

as $\boldsymbol{\sigma} = \left[|f_i^0 - \mu_0|, |f_i^1 - \mu_1|, \ldots, |f_i^{k-1} - \mu_{k-1}|\right] \in \mathbb{R}^k$, where $i \in \{0, 1, \ldots, N_c - 1\}$.

Partition The other problem is determining if each dimension contributes to the similarity calculation. We intuitively set a threshold parameter β to deal with it. Then, the CDS Θ of data x_i can be represented as:

$$\Theta(x_i) = \left[\mathbb{I}(|f_i^0 - \mu_0|), \ldots, \mathbb{I}(|f_i^{k-1} - \mu_{k-1}|)\right], \quad (3)$$

where $\mathbb{I}(\triangle f)$ is a binary decision function. When the feature difference $\triangle f > \beta$, then $\mathbb{I}(\triangle f) = 1$, otherwise $\mathbb{I}(\triangle f) = 0$.

Comparison After that, the relationship between data can be subdivided into two types from the CDS perspective. Take x_i and x_j as an example:

- if $\forall k' \in \{0, 1, \ldots, k-1\}, \mathbb{I}(|f_i^{k'} - \mu_{k'}|) = \mathbb{I}(|f_j^{k'} - \mu_{k'}|)$, then x_i and x_j have the same contributing dimension structure, noted as $R_{ij} = 1$;
- if $\exists k' \in \{0, 1, \ldots, k-1\}, \mathbb{I}(|f_i^{k'} - \mu_{k'}|) \neq \mathbb{I}(|f_j^{k'} - \mu_{k'}|)$, then x_i and x_j have different contributing dimension structures, noted as $R_{ij} = 0$.

To achieve this division, we adopt the cosine similarity to measure the similarity between the CDS of x_i and the CDS of x_j. If the cosine similarity of $\Theta(x_i)$ and $\Theta(x_j)$ is equal to 1, then it means that x_i and x_j have the same CDS; otherwise, it means that x_i and x_j have different CDS. Following this approach, the CDS relationship between each data is calculated and analyzed individually. Then, we can obtain a relationship matrix $\boldsymbol{R}^c \in \mathbb{R}^{N_c \times N_c}$ for the class c, where $R_{ij}^c \in \{0, 1\}$. The relationship matrix $\boldsymbol{R}^c$ will be used in the subsequent CDS Constraint algorithm.

3.2 Analyses

We performed experiments to analyze the connection between CDS diversity and the performance of models. Firstly, we computed the CDS relationship matrix for each class in the dataset using the CDS metric. Then, guided by the relationship matrix, two classes of data were sampled according to the sampling rate, *i.e.,* more data with the same CDS (more S-CDS) and more data with different CDS (more D-CDS). They were used to train the models separately and then we compared their performances with that of the Random method. Please refer to the *Supplementary Material* for details of the experimental setup. Figure 4(a) shows the comparison results. It shows that the more D-CDS strategy outperforms the more S-CDS strategy when sampling 0.1%-10% of CIFAR-10; the two strategies are evenly matched as the sampling rate increases. Therefore, for data with the same overall similarity, selecting a subset with different CDS impacts performance differently than selecting a subset with the same CDS at low sampling rates. Specifically, more data with different CDSs need to be sampled.

When further using the CDS metric to analyze the CDS relationships of the data sampled by existing SOTA methods, we find that they tended to select data with the same CDS, as evidenced in Figure 4(b). With reference to as shown in Figure 4(a), it can be deduced that the coresets selected by the existing SOTA methods are sub-optimal.

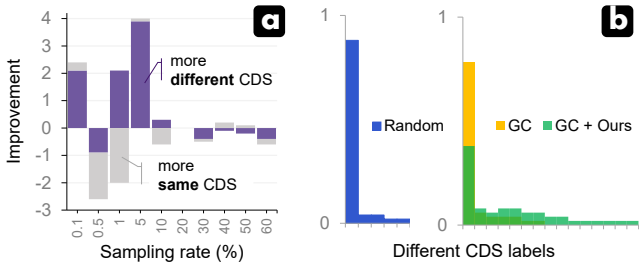


Figure 4: Analyses. (a) The improvement of sampling more same CDS strategy or more different CDS strategy over random sampling. The strategy of sampling more of the same CDS performs worse than random sampling. The strategy of sampling more different CDSs performs better than random sampling, especially with low sampling rates of 0.1%-10%. The result motivates us to select more samples with different CDS. (b) We compare the CDS distribution of coresets (1% of the CIFAR-10) selected by the baseline method and our improved counterparts. It exhibits that previous methods tend to choose a few certain CDSs, which could lead the trained model to perform worse than random sampling. Integrating our proposed constraint explicitly increases the diversity of CDS in the selected coreset.

4 Coreset Selection with CDS Constraints

Inspired by the results in Figure 4, in this section, we propose to improve the SOTA methods by using a feature-based CDS diversity constraint (a.k.a. CDS constraint). The key idea of the CDS constraint is to sample a subset with as many different CDS as possible. We present two implementations of the CDS constraint: the *hard* version, which can be directly applied to existing methods, and the *soft* version, which requires custom design aligned with the objective function of the targeted coreset selection methods.

4.1 Implementation I: Hard CDS Constraint

The coreset selection algorithm with Hard CDS Constraint consists of a two-stage clustering process and a data selection process. The data are first clustered based on the feature distance and CDS relationships. Then, the data are selected using a baseline selection method among data clusters with the same feature distance and CDS. We depict the details in the following:

1. **1st stage clustering:** the distance between each reduced feature and the central feature (of the class or dataset) is calculated and the data are clustered according to the spacing α. That is, samples with feature distance values $v \in [h \times \alpha, (h+1) \times \alpha)$ are clustered into one group. In our experiments, $\alpha = 0.5$ and $h \in N^+$. At this point, the sampling budget for each cluster is calculated based on the cluster density.
2. **2nd stage clustering:** for each cluster in the 1st stage, the CDS relationship between the data within the cluster is calculated and the data with the same CDS are re-clustered to obtain multiple clusters t. At this point, to achieve the CDS constraint, we constrain the sampling budget for each cluster t to be as consistent as possible.

3. **data selection**: each cluster t is then sampled according to the baseline method. Please refer to the *Supplementary Material* for a detailed algorithm of coreset selection with Hard CDS Constraint.

4.2 Implementation II: Soft CDS Constraint

Although Hard CDS Constraint does not need to be redesigned and can be used directly with any of the coreset selection methods, in practice, it has been found to improve only some methods (see the *Supplementary Material* for related experiments). This means that for other methods, a special design is required. For this reason, we have further proposed the Soft CDS Constraint, which is integrated into the objective function T of existing methods, *i.e.,* CRAIG and Graph Cut. We expect to constrain the original objective function T by designing a constraint function H related to the relationship matrix $\boldsymbol{R}$ to find:

$$\mathcal{S}^{\star} = \underset{\mathcal{S}\subset\mathcal{D}}{\arg\max}\, T\left(\mathcal{S}\right) \times H(\boldsymbol{R}) \quad \text{s.t.} \quad |\mathcal{S}| \leq b, \tag{4}$$

where $\mathcal{S}^{\star}$ satisfies the CDS constraint:

$$\Psi(\mathcal{S}^{\star}) \geq \Psi(\mathcal{S}^{*}). \tag{5}$$

$\Psi(\mathcal{S})$ calculates the number of CDS types in subset $\mathcal{S}$.

Since finding the optimal subset $\mathcal{S}^{\star}$ is NP-hard in general, we use the simple greedy algorithm (Minoux 2005) to approximately solve the coreset selection problem, as in the CRAIG and Graph Cut. The greedy algorithm starts with the empty set $\mathcal{S}_0 = \emptyset$, and at each selection iteration i, it selects an element $e \in V$ that maximizes $T\left(e|\mathcal{S}_{i-1}\right) \times H(\boldsymbol{R})$ and enriches the diversity of CDS, *i.e.,* $\mathcal{S}_i = \mathcal{S}_{i-1} \cup \arg\max_{e\in V} T\left(e|\mathcal{S}_{i-1}\right) \times H(\boldsymbol{R})$, where $V = \mathcal{D} \setminus \mathcal{S}_{i-1}$ and $\Psi(\mathcal{S}_i) \geq \Psi(\mathcal{S}_{i-1})$. Due to page limitations, the detailed algorithm is provided in the *Supplementary Material*.

CRAIG with the CDS Constraint CRAIG (Mirzasoleiman, Bilmes, and Leskovec 2020) is a gradient matching based method that tries to find an optimal coreset $\mathcal{S}$ that approximates the full dataset gradients under a maximum error ϵ by converting gradient matching problem to the maximization of a monotone submodular facility location function T. It uses the greedy algorithm to select data. To satisfy the CDS constraint, we design a function H to constrain the monotone submodular facility location function:

$$e = \underset{i\in\mathcal{D}\setminus\mathcal{S}_{l-1}}{\arg\max}\, T\left(i|\mathcal{S}_{l-1}\right) \times H(\boldsymbol{R})_i, \tag{6}$$

our designed constraint function H for CRAIG is:

$$H\left(\boldsymbol{R}\right)_i = 1/\left(\sum_{j\in\mathcal{S}_{l-1}} R_{ij} + 1\right). \tag{7}$$

Graph Cut with the CDS Constraint Graph Cut (Iyer et al. 2021) is a submodularity-based method that naturally measures the information and diversity of the selected subset $\mathcal{S}$. At selection iteration l, a greedy algorithm is used to find:

$$e = \underset{i\in\mathcal{D}\setminus\mathcal{S}_{l-1}}{\arg\max}\left(\sum_{o\in\mathcal{D}} s(G_i, G_o) - \lambda \times \sum_{j\in\mathcal{S}_{l-1}} s(G_i, G_j)\right), \tag{8}$$

where $s(\cdot,\cdot)$ is a similarity metric that measures the gradient similarity between data. The parameter λ captures the trade-off between diversity and representativeness, and $\lambda = 2$ in our implementation. Considering the item $\lambda \times \sum_{j\in\mathcal{S}_{l-1}} s(G_i, G_j)$ is responsible for measuring the diversity of each data, we thus propose that the CDS diversity constraint function H constrains only this item: $\lambda \times \sum_{j\in\mathcal{S}_{l-1}} s(G_i, G_j) \times H(\boldsymbol{R})_{ij}$. Our designed constraint function H is given by:

$$H\left(\boldsymbol{R}\right)_{ij} = \begin{cases} 2, & \text{if } R_{ij} = 1 \\ 1, & \text{if } R_{ij} = 0 \end{cases}. \tag{9}$$

If data x_i have the same CDS as the data x_j from $\mathcal{S}_{l-1}$, $H\left(\boldsymbol{R}\right)_{ij}$ can increase the penalty value, reducing the probability of data x_i being selected.

5 Experiments

5.1 Datasets, Model and Experimental Setup

We evaluate our method with two common data sampling modes, *i.e.,* class-balanced and class-imbalanced sampling. We perform experiments on three common image classification datasets, including CIFAR-10, CIFAR-100 (Krizhevsky and Hinton 2009), and TinyImageNet (TIN, a subset of ImageNet (Krizhevsky, Sutskever, and Hinton 2012)). We provide details of datasets in the *Supplementary Material*.

In all experiments, we utilize the 18-layer residual network (ResNet-18) (He et al. 2016) as the backbone of the pre-trained model and target model, and use the deep features extracted before the final fully connected layer for the CDS metric, which is 512-dimensional (512-D). We follow the experimental setup of work (Guo, Zhao, and Bai 2022). Specifically, we use SGD as the optimizer with batch size 128, initial learning rate 0.1, Cosine decay scheduler, momentum 0.9, weight decay 5×10^{-4}, 10 pre-trained epochs and 200 training epochs. For data augmentation, we apply random crop with 4-pixel padding and random flipping on the 32×32 training images. We use classification accuracy as the evaluation metric. For each selection method, we repeat the same experiment 5 times with random seeds and report the performance mean and standard deviation. All experiments were run on Nvidia Tesla V100 GPUs.

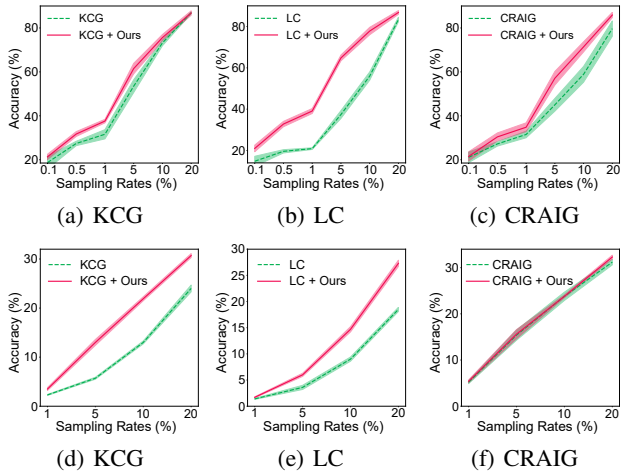
5.2 Comparison Methods

We reproduce nine selection methods ourselves based on the open source database[2], including Random, K-Center Greedy (KCG) (Sener and Savarese 2017), Forgetting (Toneva et al. 2018), Least Confidence (LC) (Coleman et al. 2019), CRAIG (Mirzasoleiman, Bilmes, and Leskovec 2020), Cal (Margatina et al. 2021), Glister (Killamsetty et al. 2021b), Graph Cut (GC) (Iyer et al. 2021), and Moderate-DS (M-DS) (Xia et al. 2023). To adequately demonstrate the effectiveness of the CDS constraint, we apply the CDS constraint to four classical coreset selection methods of different types and an up-to-date selection method (each as a baseline):

[2]https://github.com/PatrickZH/DeepCore

Method	Sampling rates					
	0.1%	0.5%	1%	5%	10%	20%
Random	18.9±0.2	29.5±0.4	39.3±1.5	62.4±1.7	74.7±1.9	86.9±0.3
KCG	18.7±2.9	27.4±1.0	31.6±2.1	53.5±2.9	73.2±1.3	86.9±0.4
Forgetting	21.8±1.7	29.2±0.7	35.0±1.1	50.7±1.7	66.8±2.5	86.0±1.2
LC	14.8±2.4	19.6±0.8	20.9±0.4	37.4±1.9	56.0±2.0	83.4±1.1
CRAIG	21.1±2.4	27.2±1.0	31.5±1.5	45.0±2.9	58.9±3.6	79.7±3.5
Cal	20.8±2.8	32.0±1.9	39.1±3.2	60.7±0.8	72.2±1.5	79.9±0.5
Glister	19.5±2.1	29.7±1.1	33.2±1.1	47.1±2.6	65.7±1.7	83.4±1.7
GC	22.9±1.4	34.0±1.3	42.0±3.0	66.2±1.0	75.6±1.4	84.3±0.4
M-DS	21.0±3.0	31.8±1.2	37.7±1.4	63.4±2.2	78.0±1.3	87.9±0.5
GC+Ours	**24.6±1.7**	**36.4±1.0**	**43.1±1.8**	**67.1±0.6**	76.9±0.2	85.2±0.6
Δ	1.7 ↑	2.4 ↑	1.1 ↑	0.9 ↑	1.3 ↑	0.9 ↑
M-DS+Ours	22.0±2.0	33.0±1.3	40.7±1.0	64.9±0.8	**79.6±0.4**	**87.9±0.2**
Δ	1.0 ↑	1.2 ↑	3.0 ↑	1.5 ↑	1.6 ↑	0.0 ↑

Table 1: Comparison on the class-balanced sampling setting. We train randomly initialized ResNet-18 on coresets of CIFAR-10 selected by different methods and then test them on the test set of CIFAR-10. Bold emphasizes the best performance at each sampling rate. Δ denotes the improvement of baseline+Ours over baseline.



(a) KCG (b) LC (c) CRAIG

(d) KCG (e) LC (f) CRAIG

Figure 5: Performance improvement over baselines. We improve current methods with our proposed CDS metric and constraint. We compare the improved versions with respective baselines on CIFAR-10 (a–c) and TinyImageNet (d–f) under the class-balanced sampling setting.

- **KCG —** a feature distribution-based selection method;
- **LC —** an uncertainty-based selection method;
- **CRAIG —** a gradient matching-based selection method;
- **GC —** a submodularity-based selection method;
- **M-DS —** a feature distribution-based selection method.

We improve baseline methods KCG, LC, and M-DS using the Hard CDS constraint and the greedy sampling baselines CRAIG and GC using the Soft CDS constraint.

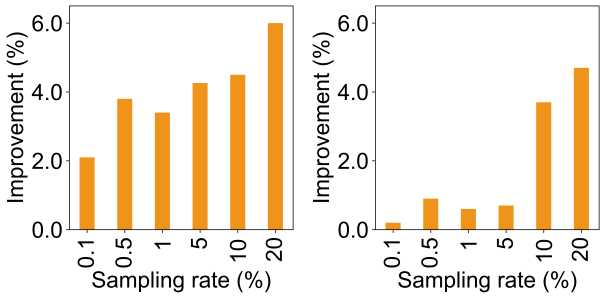
5.3 Results of Class-balanced Sampling

Table 1 and Figure 5 show the experimental results on CIFAR-10 and TinyImageNet. Please refer to *Supplementary Material* for the results for CIFAR-100. We select subsets of CIFAR-10 with fractions of 0.1%, 0.5%, 1%, 5%, 10%, 20% of the full training dataset respectively. For the best experiments of methods KCG, CRAIG, GC and M-DS on the CIFAR-10, we use the PCA algorithm to select the 10 most relevant dimensions of the features extracted from the network for the CDS metric and set β = 1e-4; for the best experiments of LC on the CIFAR-10, we directly use the 512-dimensional features extracted from the network for the CDS metric and set β = 1e-3. To ensure that a minimum of 5 images are sampled from each class of TIN dataset, we start with a 1% sampling rate and select subsets of the TIN with fractions of 1%, 5%, 10%, 20% of the full training dataset. We select the 10 least relevant dimensions for the best experiments of all baselines on TIN, with β = 1e-1.

In Table 1, we first report the improved results based on two baselines with optimal performance at different sampling rates, *i.e.,* GC and M-DS. When comparing the baseline+Ours to the baseline, it is observed that our method enhances the performance of baselines at all sampling rates. When further comparing baseline+Ours with other coreset selection methods, it can be seen that our method further strengthens the leading power of the GC at the 0.1%-5% sampling rates, and even makes the overall performance of M-DS better than the Random method on CIFAR-10. The results prove the effectiveness of our method.

In Figure 5, we show the improved results of baselines KCG, LC, CRAIG on CIFAR-10 and TIN datasets. The results consistently show that our method effectively improves these three types of coreset selection methods, proving the general effectiveness of our method. In particular, taking the LC as an example, LC+Ours outperforms LC by an average of 15.0% accuracy when sampling 0.1% to 20% of CIFAR-10, while LC+Ours outperforms LC by an average of 4.3% accuracy when sampling 1% to 20% of TIN.

It needs to be emphasized that KCG, CRAIG, GC, and M-DS employ the overall similarity metric in the measurement



(a) CIFAR-10 (b) CIFAR-100

Figure 6: Performance improvement of GC+Soft CDS over GC in the class imbalanced way

	dim reduction	partition	CDS-r	cons-traint	
(v1)	✗	✗	✗	✗	34.0±1.3
(v2)	✗	✗	✗	✓	32.8±0.7
(v3)	✓	✗	✗	✓	34.3±2.5
(v4)	✓	✓	✗	✓	33.5±1.1
full	✓	✓	✓	✓	36.4±1.0

Table 2: Ablation study on 0.5% of the CIFAR-10

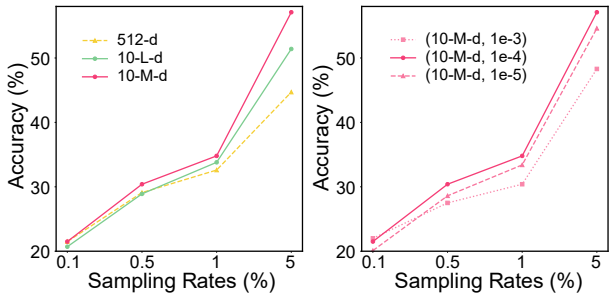
stage, whereas LC uses the predicted probability directly without using the overall similarity metric. This means that our method not only improves baselines that use the overall similarity metric but is also effective for other baselines. We reveal by visualizing the distributions that existing methods do not handle subset diversity well, while our method motivates them to capture diversity adequately, thus boosting model performance remarkably well. For length reasons, the visualisations are shown in the *Supplementary Material*.

5.4 Results of Class-imbalanced Sampling

In this subsection, we choose the method GC to perform class imbalanced sampling on CIFAR-10 and CIFAR-100. Soft CDS is used to improve it. For the experiments on the CIFAR-10, we use the PCA algorithm to select the 10 most relevant dimensions of the features extracted from the network for the CDS metric and set β = 1e-4. We select the 10 least relevant dimensions for the experiments on CIFAR-100, with β = 1e-2. We show the performance improvement of GC+Soft CDS over GC in Figure 6. It can be seen that soft CDS effectively improves the performance of GC on both datasets. For example, Soft CDS improves the performance of GC by an average of 4.0% when sampling 0.1% to 20% of the CIFAR-10.

5.5 Ablation and Parameter Studies

Ablation Study Our method consists of four parts: dimension reduction (dim. reduction), partition, CDS relationship (CDS-r), and CDS diversity constraint (constraint). Since the validity of the constraint has been demonstrated in Figure 4(a), we evaluate the effectiveness of the other three parts



(a) k (b) β

Figure 7: Parameter analysis. It shows that our method achieves the best improvement compared to the baseline method (CRAIG) when K =10-M-D and β =1e-4.

based on the GC (denoted as v1) in Table 2. When introducing the feature into the GC using $L2$-norm and constraining the CDS diversity based on the feature similarity (as v2), it gets a performance drop of 1.2%. When the dim. reduction part is added (as v3), it obtains a 1.5% accuracy improvement, exceeding the performance of baseline GC. When the partition part (as v4) is then introduced, the performance is harmed. However, when $L2$-norm is replaced with CDS metric to compute CDS relationship for CDS constraint (as full), the performance is optimal, *i.e.,* achieving 36.4±1.0 accuracy. It can prove the effectiveness of dim. reduction and CDS-r, while the partition is valid in the CDS metric.

Parameter Study The dimension k of the pruned feature and the contribution threshold β are important parameters for our method. We first study the effect of k in Figure 7(a), and then study the effect of β based on the best choice of k in Figure 7(b). We have tried three kinds of k, namely (1) original feature extracted before the final fully connected layer, where k = 512-D; (2) the 10 most relevant dimensions (10-M-D) of the feature extracted from the network; (3) the 10 least relevant dimensions (10-L-D) of the feature extracted from the network. Each type of k corresponds to one $\tilde{\beta}$, which is the maximum value that satisfies $\sum_{i\in n}\sum_{j\in k}\Theta_{ij}/(n\times k)\geq 0.9$. Based on the optimal k, we empirically set $\beta\in\{10\times\tilde{\beta},\tilde{\beta},0.1\times\tilde{\beta}\}$ to find the optimal β.

6 Conclusion

This paper introduces CDS to the coreset selection and a novel CDS metric for evaluating diversity. Utilizing this metric, we propose a CDS constraint to augment diversity within coreset selection methods. Our extensive experimental results affirm the effectiveness of our approach across a spectrum of methods. The current pipeline does not take the time cost of selecting data into account. In future work, we hope to design a more efficient and robust baseline for coreset selection.

Acknowledgements

We thank our anonymous reviewers for valuable feedback. This research was supported by Hubei Key R&D (2022BAA033), National Natural Science Foundation of China (62171325), A*STAR AME Programmatic Funding A18A2b0046, RobotHTPO Seed Fund under Project C211518008, EDB Space Technology Development Grant under Project S22-19016-STDP, JSPS KAKENHI JP22H03620, JP22H05015, and the Value Exchange Engineering, a joint research project between Mercari, Inc. and RIISE. The Supercomputing Center of Wuhan University supports the supercomputing resource.

References

Agarwal, S.; Arora, H.; Anand, S.; and Arora, C. 2020. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, 137–153. Springer.

Campbell, T.; and Broderick, T. 2018. Bayesian coreset construction via greedy iterative geodesic ascent. In *International Conference on Machine Learning*, 698–706. PMLR.

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*.

Das, S.; Patibandla, H.; Bhattacharya, S.; Bera, K.; Ganguly, N.; and Bhattacharya, S. 2021. TMCOSS: Thresholded Multi-Criteria Online Subset Selection for Data-Efficient Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6341–6350.

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.

Har-Peled, S.; and Kushal, A. 2007. Smaller coresets for k-median and k-means clustering. *Discrete & Computational Geometry*, 37(1): 3–19.

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.

Iyer, R.; Khargoankar, N.; Bilmes, J.; and Asanani, H. 2021. Submodular combinatorial information measures with applications in machine learning. In *Algorithmic Learning Theory*, 722–754. PMLR.

Killamsetty, K.; Abhishek, G. S.; Lnu, A.; Ramakrishnan, G.; Evfimievski, A.; Popa, L.; and Iyer, R. 2022. Automata: Gradient based data subset selection for compute-efficient hyper-parameter tuning. *Advances in Neural Information Processing Systems*, 35: 28721–28733.

Killamsetty, K.; Durga, S.; Ramakrishnan, G.; De, A.; and Iyer, R. 2021a. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, 5464–5474. PMLR.

Killamsetty, K.; Sivasubramanian, D.; Ramakrishnan, G.; and Iyer, R. 2021b. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 8110–8118.

Kothawade, S.; Kaushal, V.; Ramakrishnan, G.; Bilmes, J.; and Iyer, R. 2022. Prism: A rich class of parameterized submodular information measures for guided data subset selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10238–10246.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases*, 1(4).

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25.

Li, J.; Cheng, K.; Wang, S.; Morstatter, F.; Trevino, R. P.; Tang, J.; and Liu, H. 2017. Feature selection: A data perspective. *ACM computing surveys*, 50(6): 1–45.

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

Minoux, M. 2005. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques: Proceedings of the 8th IFIP Conference on Optimization Techniques Würzburg, September 5–9, 1977*, 234–243. Springer.

Mirzasoleiman, B.; Bilmes, J.; and Leskovec, J. 2020. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, 6950–6960. PMLR.

Pearson, K. 1901. LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11): 559–572.

Pooladzandi, O.; Davini, D.; and Mirzasoleiman, B. 2022. Adaptive second order coresets for data-efficient machine learning. In *International Conference on Machine Learning*, 17848–17869. PMLR.

Rangwani, H.; Jain, A.; Aithal, S. K.; and Babu, R. V. 2021. S3vaada: Submodular subset selection for virtual adversarial active domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7516–7525.

Sener, O.; and Savarese, S. 2017. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*.

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

Tiwari, R.; Killamsetty, K.; Iyer, R.; and Shenoy, P. 2022. Gcr: Gradient coreset based replay buffer selection for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 99–108.

Toneva, M.; Sordoni, A.; Combes, R. T. d.; Trischler, A.; Bengio, Y.; and Gordon, G. J. 2018. An empirical study

of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*.

Tsang, I. W.; Kwok, J. T.; Cheung, P.-M.; and Cristianini, N. 2005. Core vector machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, 6(4): 363–392.

Wan, Z.; Wang, Z.; Chung, C.; and Wang, Z. 2023a. A survey of dataset refinement for problems in computer vision datasets. *ACM computing surveys*.

Wan, Z.; Xu, X.; Wang, Z.; Wang, Z.; and Hu, R. 2023b. From multi-source virtual to real: Effective virtual data search for vehicle re-identification. *IEEE Transactions on Intelligent Transportation Systems*.

Wan, Z.; Xu, X.; Wang, Z.; Yamasaki, T.; Zhang, X.; and Hu, R. 2022. Efficient virtual data search for annotation-free vehicle reidentification. *International Journal of Intelligent Systems*, 37(5): 2988–3005.

Xia, X.; Liu, J.; Yu, J.; Shen, X.; Han, B.; and Liu, T. 2023. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 1–20.

Xie, H.; Yang, Z.; Zhu, H.; and Wang, Z. 2023. Striking a balance: Unsupervised cross-domain crowd counting via knowledge diffusion. In *Proceedings of the 31st ACM International Conference on Multimedia*, 6520–6529.

Xie, P.; Xu, X.; Wang, Z.; and Yamasaki, T. 2022. Sampling and re-weighting: Towards diverse frame aware unsupervised video person re-identification. *IEEE Transactions on Multimedia*, 24: 4250–4261.

Yang, Z.; Zhong, X.; Zhong, Z.; Liu, H.; Wang, Z.; and Satoh, S. 2023. Win-win by competition: Auxiliary-free cloth-changing person re-identification. *IEEE Transactions on Image Processing*.

Yoon, J.; Madaan, D.; Yang, E.; and Hwang, S. J. 2021. Online coreset selection for rehearsal-based continual learning. *arXiv preprint arXiv:2106.01085*.

Yuan, X.; Xu, X.; Wang, X.; Zhang, K.; Liao, L.; Wang, Z.; and Lin, C.-W. 2023. OSAP-Loss: Efficient optimization of average precision via involving samples after positive ones towards remote sensing image retrieval. *CAAI Transactions on Intelligence Technology*.