Effective Subset Selection Through The Lens of Neural Network Pruning

Anonymous authors
Paper under double-blind review

Abstract

Having large amounts of annotated data significantly impacts the effectiveness of deep neural networks. However, the annotation task can be very expensive in some domains, such as medical data. Thus, it is important to select the data to be annotated wisely, which is known as the subset selection problem. We investigate and establish a relationship between subset selection and neural network pruning, which is more widely studied. Leveraging insights from network pruning, we propose utilizing the norm criterion of neural network features to improve subset selection methods. We empirically validate our proposed strategy on various networks and datasets, demonstrating enhanced accuracy. This shows the potential of employing pruning tools for subset selection.

1 Introduction

An important factor in the success of deep neural networks is having a large set of annotated data. However, it can be difficult to obtain such labeled data. For example, in medical applications, annotation must be done by expert doctors whose time is expensive. Consequently, given limited resources, it is imperative to select the examples to be annotated from a large unlabeled dataset wisely, so as to extract maximum information from the data. Our work focuses on this challenge, known as the *subset selection problem*.

The subset selection problem is closely related to active learning, which involves gradually selecting unlabeled examples for annotation during the learning process. While gradual annotation has benefits, subset selection is the special case where all the samples to be annotated must be selected at once, all together. Subset selection poses several challenges. Firstly, determining the criteria for selecting informative examples is non-trivial, as it requires balancing various factors, such as diversity, relevance, and coverage of the data distribution. Subset selection based on simple criteria such as uncertainty, entropy, and margin between the highest scores has been demonstrated to be ineffective. Moreover, it has been shown that many methods proposed for subset selection fail to outperform random choice, particularly when a very small set of examples is chosen Hacohen et al. (2022); Chen et al. (2022); Guo et al. (2022).

In this study, we explore the relationship between training-data subset selection and neural network pruning, which is a well-researched area Mozer & Smolensky (1988); Molchanov et al. (2016); LeCun et al. (1989); Hassibi et al. (1993); Chauvin (1988); Carreira-Perpinán & Idelbayev (2018); Louizos et al. (2018); Bellec et al. (2018); Mocanu et al. (2018); Mostafa & Wang (2019); Novikov et al. (2015); Jaderberg et al. (2014); Chen et al. (2023; 2021); Neill (2020); van Amersfoort et al. (2020); Lee et al. (2018); de Jorge et al. (2020); Alizadeh et al. (2022); Paul et al. (2022); Lee et al. (2020b); Wang et al. (2020); Zhang et al. (2021b); Su et al. (2020); Liu & Zenke (2020); Sun et al. (2024). Neural pruning reduces the computational costs of training and inference in deep models (see Vadera & Ameen (2022) for a survey).

We propose that input data can be seen as part of the neural network structure, suggesting that methods for pruning weights can also be applied to 'prune' the training examples (see Fig. 1). Motivated by this relation, we focus on migrating to subset selection the use of norms of network features that is practiced in pruning. Structured pruning methods have shown the effectiveness of retaining filters with high norms while discarding those with low norms Li et al. (2017); He et al. (2018; 2019). Similarly, unstructured pruning techniques, particularly at initialization, prioritize weights with higher magnitudes Frankle & Carbin (2018).

Building upon these insights from prior works, we propose leveraging the norm criterion of features in the data to enhance the efficacy of selection methods. We start by investigating how the norm of the features is a contributor to subset performance by using the norms' values as a probability from which the subset is sampled. Our findings reveal that subsets characterized by high norms exhibit superior accuracy following training. Therefore, we suggest a weighted sampling criterion that relies on the norm. Additionally, we tested the accuracy of numerous subsets sampled uniformly at random and observed a correlation between norm and accuracy, indicating the significance of norm in determining the effectiveness of the selected subsets.

Relying solely on examples' feature norms is limited, as it misses data correlations. Thus, we suggest using the Gram-Schmidt process to stably choose examples orthogonal to those already selected. This facilitates selecting examples whose features have the highest norm in the 'remaining subspace' that is not spanned by the previously selected examples. Handling the data correlations in this way promotes a comprehensive coverage of the features' domain.

We present results with features induced by both randomly initialized networks and self-supervised networks, namely, SimCLR Chen et al. (2020) and DINO Caron et al. (2021), on the CIFAR-10/100, Tiny-ImageNet, ImageNet and OrganAMNIST Yang et al. (2023) datasets. We show that combining our norm criterion with other subset selection methods can significantly improve the overall performance, and thus achieve new state-of-the-art results in most cases. We demonstrate that our approach is versatile, performs well across frameworks and is applicable to various feature domains.

Our contributions are summarized as follows: (i) A novel relation between pruning and subset selection; (ii) Proposing the network's feature norm and Gram-Schmidt algorithm as successful subset selection tools; and (iii) Comprehensive experiments demonstrating our framework's advantages.

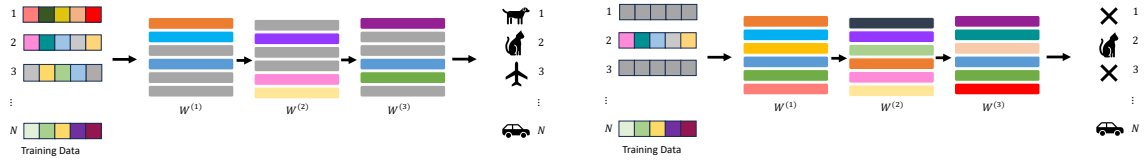
2 Related Work

Subset Selection strategies for choosing examples include uncertainty sampling Gal et al. (2017); Settles (2009), diversity maximization Ash et al. (2019); Citovsky et al. (2021); Beluch et al. (2018); Hsu & Lin (2015); Chang et al. (2017); Chen et al. (2020), and entropy criteria Coleman et al. (2019). Generally, these approaches aim to select examples that best represent the underlying data distribution while minimizing redundancy.

While both subset selection and active learning share the goal of selecting informative examples for labeling, the main difference lies in their approach and methodology. Subset selection focuses on choosing a fixed subset of examples from the entire dataset, while active learning involves dynamically selecting examples for labeling based on the current model's uncertainty or informativeness. Notably, in subset selection, the chosen subsets should ideally be model-agnostic and capable of achieving good performance when training the model from scratch. In active learning framework, numerous methods are tailored to the geometry of the problem and often rely on k-means clustering to select diverse sets Hacohen et al. (2022); Sener & Savarese (2018); Sorscher et al. (2022); Xia et al. (2022). Others aim to cover the embedding space Yehuda et al. (2022); Zheng et al. (2022a). Others adopt an optimization perspective Borsos et al. (2020); Paul et al. (2021); Killamsetty et al. (2021b), suggesting the estimation of the gradient of the entire dataset using a small amount of data Killamsetty et al. (2021a); Mirzasoleiman et al. (2020).

Recent challenges in subset selection focus on choosing examples from a labeled pool rather than an unlabeled one Gadre et al. (2024); Mazumder et al. (2024). Recent challenges in subset selection focus on choosing examples from a labeled pool rather than an unlabeled one Gadre et al. (2024); Mazumder et al. (2024). Moreover, Guo et al. (2022) introduce a framework for subset selection.

The works relevant to our research are the ProbCover Yehuda et al. (2022) and TypiClust Hacohen et al. (2022) algorithms, which to the best of our knowledge are the current state-of-the-art extremely small subset selection approaches for the image classification task. ProbCover is a novel subset selection strategy designed to address the challenges of deep active learning in low-budget regimes. It leverages recent advancements in self-supervised learning to enrich the geometry of data representations, enabling more effective subset selection for annotation. By maximizing Probability Coverage, ProbCover aims to select examples that contribute the most information to the learning process, thereby reducing annotation costs while least harming



(a) **Structured pruning:** Selection of a small subset of neurons.

(b) **Subset Selection:** Choosing a small subset of training examples.

Figure 1: Subset selection is analogous to pruning of the input data, which can be considered the first layer of the network.

performance. TypiClust is a subset selection technique specifically tailored to low-budget scenarios, where only a limited number of labeled examples are available for training. Based on theoretical analysis revealing a phase transition-like behavior, TypiClust employs a querying strategy that prioritizes typical examples when the budget is constrained and unrepresentative examples when the budget is larger. This approach capitalizes on the observed phenomenon that typical examples contribute the most information in low-budget settings, leading to improved model performance. Both of these method depend heavily on the feature domain, and their performance drops when the features are uninformative. Moreover, methods such as ProbCover and TypiClust require the computation of clustering or adjacency graphs. In contrast, we demonstrate that our method does not rely heavily on the domain and is less time-consuming.

Neural Network Pruning has emerged as a technique for reducing model complexity and compute, see Vadera & Ameen (2022) for a comprehensive survey. Various pruning methods have proposed using norm-based techniques, graduality, and randomization. Norm or magnitude based techniques discard parameters with low magnitudes, often achieving significant compression with only a minor harm to performance Han et al. (2015); He et al. (2019; 2018); Li et al. (2016); Frankle & Carbin (2018); See et al. (2016); Guo et al. (2016); Narang et al. (2016); Tung & Mori (2018); Lubana & Dick (2020); Sun et al. (2024). Gradual pruning iteratively removes the least important weights, allowing networks to adapt gradually and maintain performance Frankle & Carbin (2018); Han et al. (2015); Lee et al. (2020a). Finally, using randomization in the pruning process further enhances performance Bar & Giryes (2023); He et al. (2019).

We focus on interpreting subset selection via pruning, utilizing the features' norm and randomization to subset selection together with an algorithm to ensures feature diversity.

3 Pruning and Subset Selection

In subset selection, given N unlabeled examples and their corresponding features, $F_1, ..., F_N$, we select a small subset of examples S containing s examples for labeling. The goal is to maximize the performance of the model after training, utilizing these small labeled subsets.

The key in our work is the analogy we draw between pruning and subset selection: selecting examples from the dataset is analogous to choosing filters in the pruning setting. Specifically, viewing input examples as the very first layer of the network suggests that choosing a filter corresponds to selecting an example. Both, the pruned network weights and the input examples that are not in the selected subset are treated as zero. Neither 'unselected' examples nor pruned weights participate in the training of the network. Fig. 1 provides a visual illustration. Indeed, the correspondence holds when filter pruning is performed in a layer-wise manner and all other layers remain static; thus, this is simply an analogy and there are clear differences between the two cases.

The features' norm is the first tool we migrate from pruning. Extensive research in network pruning has emphasized the significance of high-norm filters for structured pruning He et al. (2018; 2019), high-magnitude weights for unstructured pruning Frankle & Carbin (2018) and the norms of features Li et al. (2017). In Appx. F we further show the bias of pruned network to high feature norms.

Motivated by norm-based pruning, we use the norm of features for subset selection. The features can be obtained through various methods, including pre-training procedures or randomly initialized networks. We

focus on the features of the norms and not the input itself. Clearly, the input impacts the features created throughout the network layers, and hence relying on the features norm is data driven.

Feature Norm. Let $F_1, ..., F_N$ denote the features corresponding to N unlabeled training examples. Then, we randomize the examples according to the following probability:

$$p_i = \frac{\|F_i\|}{\sum_{j=1}^{N} \|F_j\|}, \quad i = 1, ..., N, \tag{1}$$

where $\|\cdot\|$ is the ℓ_2 norm unless otherwise stated. In Sec. 4, we demonstrate that this simple choice of examples performs better than uniform random selection, which is a non-trivial baseline, particularly with extremely small subsets Hacohen et al. (2022); Chen et al. (2022). Moreover, we show that there is an advantage to randomization over determinstic selection of the highest weights.

Gram-Schmidt. In the context of subset selection, it is crucial to emphasize that solely relying on feature norms may not yield optimal results. While feature norms provide valuable information, they may not capture the full complexity of the dataset. The norm values do not contain information about the correlations between data points. Therefore, it is essential to augment norm-based selection methods with additional concepts to enhance their effectiveness. We suggest leveraging techniques from linear algebra and choosing features that span the domain. Namely, we utilize the Gram-Schmidt process described in Alg. 1 to iteratively choose orthogonal features. Initially, we randomly select an example to be labelled according to the norms using the probabilities in Eq. (1). Then, we update the set of chosen examples, S. Finally, we remove the projection onto the chosen feature from the remaining features, which is the third step of the Gram-Schmidt process, ensuring orthogonality of the remaining features to the selected ones. In this way we handle the correlations between the data points. We ensure that the chosen examples not only have high feature norms, but also capture diverse and informative aspects of the dataset.

Algorithm 1 Gram–Schmidt for Subset Selection

Input: $F_1, ..., F_N$ features, s the number of examples to label.
Initialize: $\tilde{F}_1, ..., \tilde{F}_N = F_1, ..., F_N$ and $S = \emptyset$
for $i = 1$ **to** s **do**
  Randomize i according to $p_j = \frac{\|\tilde{F}_j\|}{\sum_{k=1}^{N} \|\tilde{F}_k\|}, j \notin S$
  Update: $S = S \cup \{i\}$
  Projection: For $j \notin S$: $\tilde{F}_j = \tilde{F}_j - \frac{\tilde{F}_j^T \tilde{F}_i}{\|\tilde{F}_i\|^2} \tilde{F}_i$
end for
return S

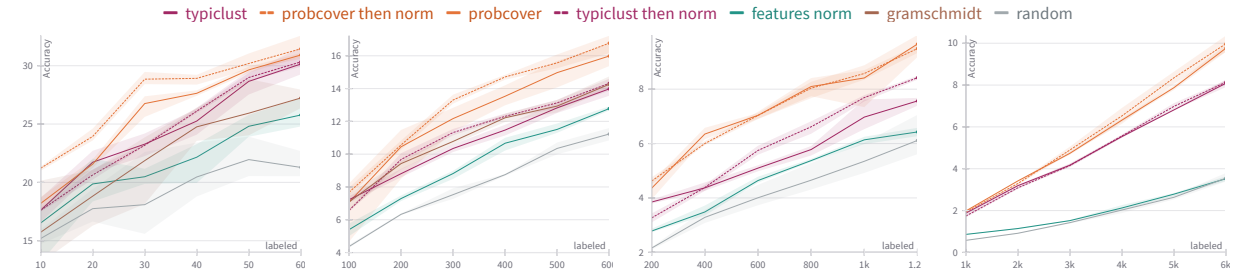
The Gram-Schmidt process is known to be numerically unstable when the projections are calculated with the initial inputs rather than gradually. We address this issue in Alg. 1 by performing the projection with $\tilde{F}_i$ rather than with the initial input vectors F_i, and update the values of the features $\tilde{F}$ in each iteration.

Alg. 1 enjoys low computation complexity. Let d be the dimension of the features. In each iteration, the randomization step takes $O(Nd)$ to calculate the features' norms and normalize them to probabilities. In the projection step, it takes $O(Nd)$ in the worst case for calculating the inner products of the $N-1$ remaining features with the chosen feature. Overall complexity is $O(sNd)$. In comparison, the baseline, ProbCover, takes $O(N^2d)$ computations. We assume that the features are given and we do not include their query to the complexity calculations since both ProbCover and our GramSchmidt rely on the availability of the features.

Randomization. In both suggested methods (norm-based and Gram-Schmidt), we incorporate randomization inspired by pruning, which also boosts performance Bar & Giryes (2023). Instead of selecting examples with the maximal norms, we perform weighted random selection based on the norms. We argue that using randomization is crucial for achieving good performance. In Fig. 6c we demonstrate the role of randomization in subset selection performance.

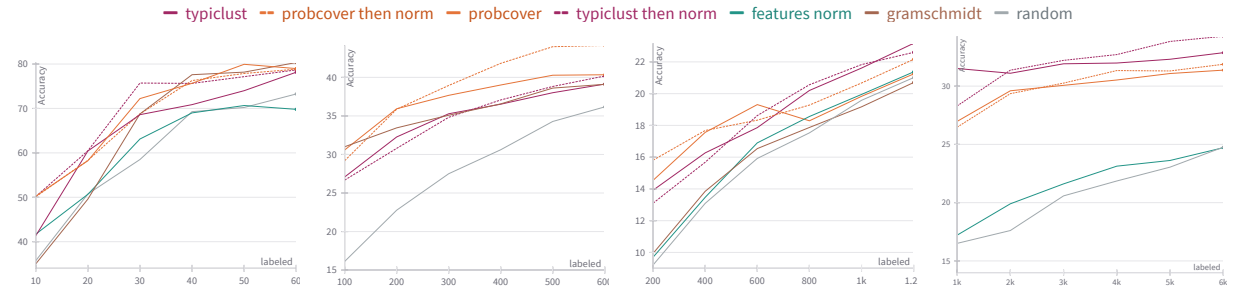
4 Experiments

We validated our proposed strategy empirically across diverse settings and datasets, focusing on extremely small subsets selected from unlabeled pools. Consequently, in many instances, not all classes are represented in the labeled training set, leading to potential class imbalances within the subsets.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny-ImageNet (d) ImageNet

Figure 2: **Fully-supervised framework:** The performance comparison includes both of our methods, randomization with feature norms and Gram-Schmidt, random selection, the baselines TypiClust and ProbCover, and the addition of our norm criterion to these baselines. An average of 3 seeds is presented and the shaded areas correspond to the standard error.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny-ImageNet (d) ImageNet

Figure 3: **Semi-supervised with linear classifier:** Results with a linear classifier trained on top of self-supervised features

We evaluate our method in three frameworks: (i) *Fully supervised*: Training exclusively with annotated data using an initialized ResNet-18 from scratch. (ii) *Semi-supervised with linear probing*: Training a single-layer linear classifier on top of self-supervised features obtained from an unlabeled dataset. This framework aims to leverage semi-supervised learning principles without relying on recent advances in pseudo-labeling techniques. (iii) *Semi-supervised*: Training competitive semi-supervised methods using subsets chosen by the selection algorithms. We employ FlexMatch Zhang et al. (2021a) and SimMatch Zheng et al. (2022b) to assess the effectiveness of our selection algorithm.

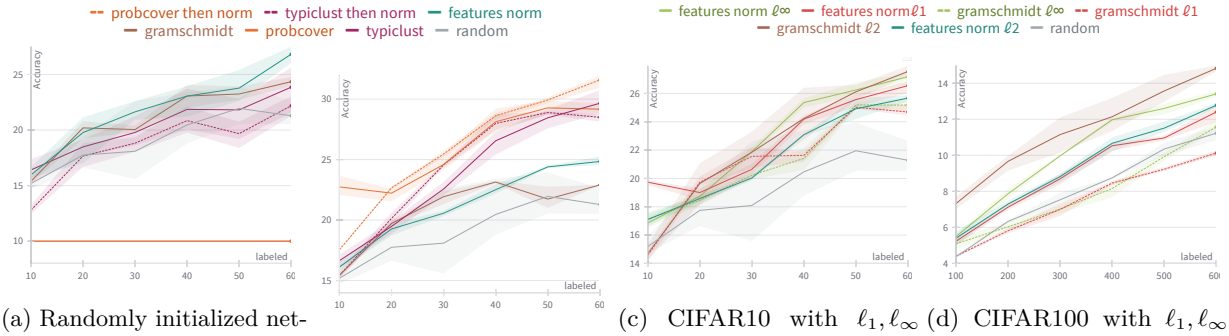
We present results with randomization along the feature norms approach according to Eq. (1), the Gram-Schmidt based strategy (Alg. 1), and a combination of ProbCover Yehuda et al. (2022) and TypiClust Hacohen et al. (2022) with the norm criterion. ProbCover and TypiClust rely on self-supervised embeddings. ProbCover is a method which selects examples that contribute the most information to the learning process and TypyClust prioritizes typical examples to improve the performence with exteremly low budgets. We combined the methods with the norm criterion and GS (Alg. 1), named "*<method> then norm*" or "*<method> then GS*", by selecting twice the required budget, denoted as $2b$, and then randomly selecting b examples based on feature norms. We choose to embed the norm criterion this way since the methods rely on normalized features and are very sensitive to changes in the feature domain. In the paper, we only show the combination of the feature norm criterion with these methods. In Appx. D, we also show results with Gram-Schmidt on top of the baselines. When combined with the baseline, using Gram-Schmidt enhances the performance mainly in setting (ii) with a linear classifier.

For frameworks (i) and (ii), we use code adapted from Yehuda et al. (2022); Hacohen et al. (2022), which is based on prior work Van Gansbeke et al. (2020); Munjal et al. (2020). For the semi-supervised setting (iii), we employ code of the Semi-Supervised benchmark Wang et al. (2022). Our code is attached to the paper.

We compare our method with other subset selection methods: Uniform at random, which is a competitive baseline, TypiClust Hacohen et al. (2022) and ProbCover Yehuda et al. (2022). We do not include other

	Random	Features Norm	ProbCover	ProbCover + Norm	ProbCover + GS
FlexMatch	61.66	63.84	63.7	65.51	**79.9**
SimMatch	38.03	39.68	54.68	57.08	**76.02**
FlexMatch	17.81	22.8	35.68	33.93	**40.04**
SimMatch	17.1	22.3	36.34	34.56	**40.88**

Table 1: Semi-supervised training, FlexMatch Zhang et al. (2021a) and SimMatch Zheng et al. (2022b), with CIFAR-10 (top) and CIFAR-100 (bottom). Using feature norms is better than uniform random choice and adding GS on top of ProbCover enhances accuracy.



(a) Randomly initialized network.

(b) Self-supervised DINO.

(c) CIFAR10 with ℓ_1, ℓ_∞ norms.

(d) CIFAR100 with ℓ_1, ℓ_∞ norms.

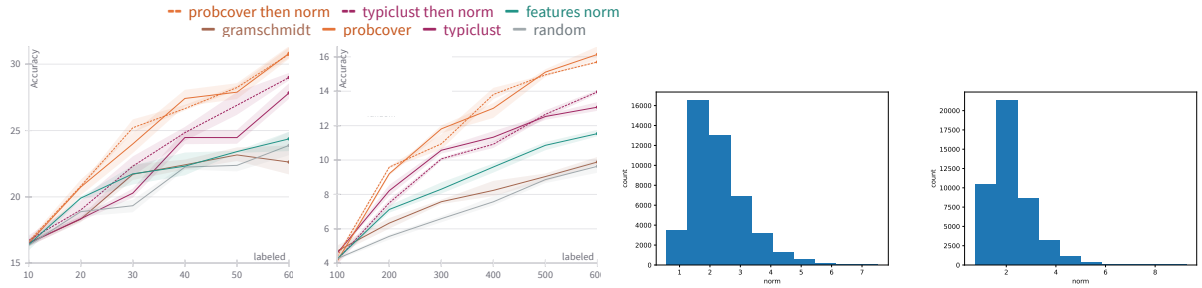
Figure 4: **Feature domain.** Figs. 4a and 4b include the results with randomly initialized NN and DINO features. They show that the benefits of the feature norm apply to various feature domains. **Norms type.** Figs. 4c and 4d compares the results of using randomization based on feature norm and Gram-Schmidt algorithm, where the ℓ_1 and ℓ_∞ are used instead of ℓ_2 for the norm. Gram-Schmidt with ℓ_2 provides the best results. For feature norm-based selection, ℓ_2 and ℓ_∞ are the best.

subset selection methods since many of them do not surpass the random selection and others are only slightly better than random selection, as demonstrated in Guo et al. (2022); Hacohen et al. (2022); Chen et al. (2022).

We tested our method on CIFAR-10, CIFAR-100, Tiny-ImageNet, ImageNet and OrganAMNIST Yang et al. (2023) (see Appx. A for additional details) datasets. Specifically, unless otherwise specified, we utilized SimCLR Chen et al. (2020) embeddings for CIFAR-10/100, Tiny-ImageNet and OrganAMNIST. For ImageNet, we utilized DINO Caron et al. (2021) embeddings. We use the SimCLR implementation from Van Gansbeke et al. (2020) and train ResNet-18 with an MLP projection layer for 500 epochs. Post-training, we extract the 512-dimensional features from the penultimate layer. Our optimization is conducted with SGD with momentum and an initial learning rate of 0.4 with a cosine scheduler. We employ a batch size of 512 and a weight decay of 10^{-4}. We augment the data with random resize and crop, random horizontal flips, color jittering, and random grayscale. For DINO, we use ViT-S/16 model pre-trained on ImageNet.

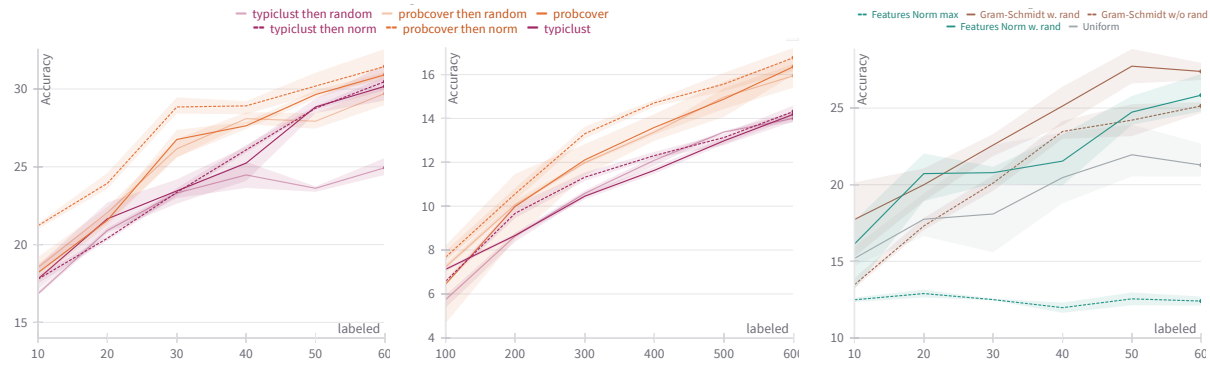
In frameworks (i) and (ii), we evaluated the methods with varying numbers of examples: $[b, 2b, ..., 6b]$, where b represents the number of classes. We report the average results over 3 seeds. For the semi-supervised framework, (iii), we utilized b labeled examples.

In the fully-supervised framework for CIFAR-10/100, Tiny-ImageNet and OrganAMNIST, we train the model for 200 epochs using the SGD optimizer with Nesterov momentum and a cosine learning rate with an initial step-size of 0.025. We utilize a batch size of $\max\{\#\text{labeled}, 100\}$ and a weight decay of 3×10^{-4}. Augmentations include random crop and random horizontal flip. For ImageNet, we use the same hyper-parameters as described above, with the exception that we train for 100 epochs and employ a batch size of 50 due to computational constraints. We employed a single NVIDIA RTX-2080 GPU to undertake the learning processes, encompassing both the acquisition of self-supervised features and training with small subsets of examples.



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

Figure 5: Figs. 5a and 5b include results with an initial random labeled pool. The size of the initial pool is the number of classes. Notice that also in this case, our approach provides benefits for subset selection. Figs. 5c and 5d present the histograms of the feature norm with self-supervised features. The norms are diverse, indicating that the distribution they induce is not vacuous.



(a) CIFAR-10 (b) CIFAR-100 (c) Randomization w. CIFAR-10

Figure 6: Figs. 6a and 6b include results obtained with uniform randomization rather than randomization according to the feature norm. They show that randomization based on feature norms is pivotal for the enhancement in performance. Fig. 6c compares selecting examples according to the feature norm or Gram Schmidt with and without randomization. Note that using randomization is beneficial.

For the semi-supervised framework with a linear classifier ,(ii), we utilize the features of the labeled data and train a $d \times C$ linear classifier, where d is the dimension of the features and C is the number of classes. To train the classifier, we use a learning rate of 2.5 and 400 epochs. For the semi-supervised framework (iii), we train FlexMatch and SimMatch with a Wide-ResNet-28-10 model using SGD with momentum for 1000k iterations. We employ a learning rate of 0.03, a batch size of 64, and a weight decay of 5×10^{-4}. Weak augmentations such as random crop and random horizontal flip are applied, while strong augmentations are obtained using RandAugment Cubuk et al. (2020).

4.1 Results

Fully supervised results. The results in Fig. 2 demonstrate that the simple baseline of randomization according to norm achieves a significant performance boost compared to uniform random selection in CIFAR-10, CIFAR-100 and Tiny-ImageNet. The results also show that our proposed algorithm contributes to CIFAR-10/100. The Gram-Schmidt method contributes an additional enhancement in performance compared to randomization based on norm alone. This is especially notable for CIFAR-100, where the results are comparable with TypiClust. Integrating the norm criterion on top of previous methods yields comparable and even better results, which is particularly evident with CIFAR-10/100 with ProbCover and with Tiny-ImageNet with TypiClust. For ImageNet, using the norm criterion and Gram-Schmidt lead to comparable results with the baselines. In Appx. E, we present results with larger subsets. In these cases, incorporating the norm criterion leads to comparable or improved accuracy.

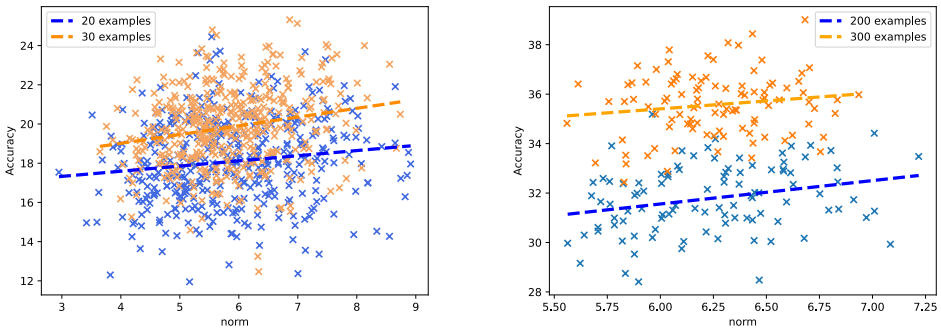


Figure 7: **Correlation with Norm:** Test Accuracy of models trained with 20/30 examples (left) and 200/300 examples (right) uniformly sampled from CIFAR-10 as a function of their self-supervised feature norms. The regression lines indicate a positive correlation between norm and accuracy. Although the correlation is not high, it remains consistent across subset sizes and self-supervised features (see results with DINO in Fig. 15).

Set size	Random	Feature Norms	Gram-Schmidt	TypiClust	TypiClust+Norm	ProbCover	ProbCover + Norm
20	271.62	**251.01**	267.2	244.81	**240.68**	253.29	**250.31**
40	215.95	214.48	**213.52**	211.66	**209.56**	207.85	**207.28**
60	188.98	188.39	**186.63**	185.88	**184.96**	182.04	**179.89**

Table 2: **FID scores:** Measure distributions similarity between subsets and the remaining training set. Low scores indicate higher similarity. Subsets taken from CIFAR-10.

Fig. 8 presents results with OrganAMNIST. The findings suggest that incorporating the norm criterion and Gram-Schmidt algorithm is advantageous particularly for higher budgets. Overall, even though we enhance performance with extremely small labeled sets, there is still a significant drop in performance compared to the more than 90% accuracy achieved with the full dataset. Thus, although we improve over the state-of-the-art, there is still room for improvement.

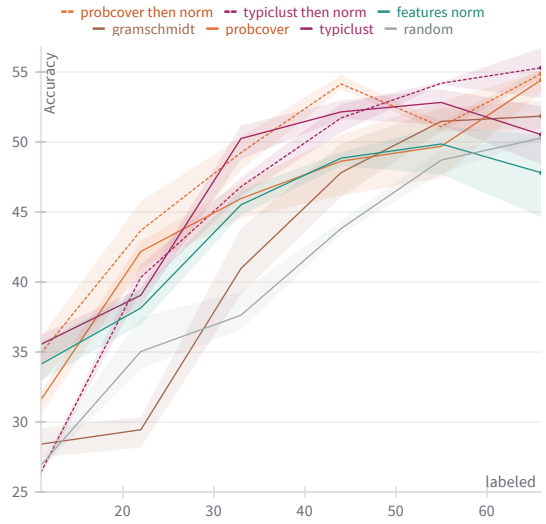


Figure 8: Fully-supervised setting with OrganAMNIST.

Semi-supervised learning with a linear classifier. Fig. 3 displays the results obtained with the semi-supervised framework, where only 1 layer of the neural network is optimized. The results indicate that norm randomization performs similarly to or slightly better than uniform randomization. Our subset selection method shows improvements primarily over norm randomization, especially with higher budgets for CIFAR-10 and with CIFAR-100, where the results are comparable with the TypiClust method. Additionally, integrating the norm criterion with the baselines, ProbCover and TypiClust, generally enhances their performance. It is noteworthy that the results surpass those of the fully supervised setting, indicating that the features used are highly informative. This information can significantly enhance performance, even when only a limited set of labels is available.

Semi-Supervised with full fine-tuning. The results for semi-supervised training, where all the network is fine-tuned and pseudo-labeling is employed, are provided in Tab. 1. Our findings demonstrate a clear improvement in accuracy when utilizing the norm criterion. Randomization according to norm boosts performance and achieves better accuracy than the sophisticated ProbCover method in some cases. Additionally, it appears that incorporating the Gram-Schmidt algorithm on top of ProbCover further enhances performance in this setting. The results do not include TypiClust due to limited availability of computing resources.

Random initial pool. Previous active learning methods for subset selection typically required an initial non-empty labeled pool for training an initial learner, from which the selection process is bootstrapped. Therefore, we include results in Fig. 5 where training is conducted with an initial uniform randomly selected pool of b labeled samples, where b is the number of classes. The results demonstrate that randomization according to norm remains beneficial for performance even in this scenario, when compared with completely random sampling. While our Gram Schmidt method does not enhance performance beyond randomization according to the norm, it does demonstrate advantages over uniform random selection. The results with the addition of the norm criterion to TypiClust and ProbCover show comparable performance to the baselines. For TypiClust, the norm addition yields improved results. Overall, we observe that also in this case, using the feature norm criterion is beneficial for performance.

4.2 Empirical Validation of Assumptions

Correlation of accuracy and norm. Fig. 7 shows the accuracy after training on CIFAR-10 subsets in a fully-supervised framework. To create the figure, we randomly sampled 100 sets of 200 and 300 labeled images and 500 sets of 20 and 30 labeled images and trained a network for them in a supervised manner. For each subset, we present the accuracy as a function of its average feature norm. Having the scatter plot, we added a regression line over the points, which demonstrates that high accuracy correlates with high feature norms, which justifies the assumption we make in this work. In Appx. G, we provide results obtained with DINO features to further support our motivation. Although in some cases the correlation is weaker, there is still a consistent positive correlation between high accuracy and high norm across all tested cases. The fact that in some cases there is a low correlation between feature norm and accuracy suggests that while feature norm is useful, as our experiments show, it can greatly benefit from complementing it with other methods that depend on features. In our work, we proposed using Gram-Schmidt orthogonalization to reduce correlations among features within the chosen subset and also combined our approach with other existing methods.

Distribution of selected examples. To evaluate the alignment of our selected subset with the training set, we employ Fréchet Inception Distance (FID) Heusel et al. (2017) scores, originally presented for numerical evaluation of generative models. FID measures the similarity between the distributions of artificially generated images and real images. We use it to measure the similarity between the chosen subsets and the remaining examples in the training set. Lower FID scores, as detailed in Tab. 2, indicate a closer match to the original dataset, particularly evident when utilizing the norm criterion. We calculate the FID scores with Seitzer (2020). Our approach results in a distribution that closely resembles the original dataset. In Appx. H, we provide qualitative examples chosen with the norm criterion to show that images with high norms are more likely to be easier to recognize for humans.

Distribution of feature norm. In order to demonstrate the sampling distribution, we present in Fig. 5 the histogram of feature norms across the entire datasets. Specifically, we illustrate the feature norms of the SimCLR features for the CIFAR-10/100 datasets. Our analysis reveals diverse values of feature norms, indicating a non-vacuous distribution from which examples are sampled.

4.3 Ablation Study

Dependency on the feature embedding. To ensure that the benefits of our method are not dependent on a specific feature domain, we conducted experiments with other embeddings. Specifically, we employed another self-supervised approach, DINO Caron et al. (2021), which is known for its informative features. We experimented with the fully-supervised framework with CIFAR-10. The results in Fig. 4b indicate that utilizing norm randomization is beneficial. Moreover, the addition of norm on top of existing methods mostly enhances the performance. Also, for the Gram-Schmidt method, we observed a performance gain, particularly at low budgets. In Appx. B, we include results on CIFAR-100.

Given the computational demands and potential unavailability of pre-trained self-supervised models, we conducted experiments with features induced by randomly initialized neural networks with CIFAR-10. The results with initialized neural networks are presented in Fig. 4a. We observed that the performance of ProbCover collapsed and suffered from accuracy comparable to a random classifier.

The benefits of TypiClust over random sampling are limited in this case, and using the norm criterion did not improve this. It is noteworthy that this is consistent with the claim made by Yehuda et al. (2022), they claim that their methods rely on informative features and suffer from poor performance with the RGB space. Our approach demonstrates benefits in both scenarios, which is advantegous. Employing the Gram-Schmidt algorithm yields benefits over random sampling and, notably, using the norm criterion alone induced high accuracy. Even though other methods experience a degradation in accuracy when using less informative features, we observe that randomization according to norm maintains its performance. Appx. B includes results with CIFAR-100 and the raw RGB space.

Othen norm types. Given that some pruning methods are related to norms other than the ℓ_2 norm Li et al. (2017), we present results with ℓ_1 and ℓ_∞ norms with CIFAR-10 and CIFAR-100 in Figs. 4c and 4d. We observe that randomization according to ℓ_∞ yields good results. The performance of Gram-Schmidt with norms other than the ℓ_2 norm harms performance, suggesting that our method suits the Euclidean norm. Randomizing according to ℓ_1 is better than ℓ_2 for CIFAR-10 but slightly worse for CIFAR-100. We employ the ℓ_2 norm due to its consistent performance across scenarios. In Appx. C we include results with ProbCover and TypiClust.

Additionally, we tested the effect of normalizing the feature norms by the norm of the input. The normalization did not significantly change the results, so we have not included them in this paper.

Necessity of randomization. We investigated the necessity of randomization for successful subset selection. In Fig. 6c, we present results, comparing scenarios where examples with the maximal norm are chosen based on feature norms and with our algorithm. Specifically, in our algorithm (Alg. 1), the randomization step is replaced with selecting the example with the highest feature norm. The results indicate that for selection based solely on norms, randomization is crucial for achieving accuracy gains, as accuracy only marginally surpasses that of a random classifier.

Replace the norm criterion with random sampling. To better assess the gain in performance for Typiclust and ProbCover that is achieved by the addition of our norm criterion, we replace the norm criterion with random sampling. Figs. 6a and 6b contain the results with CIFAR-10/100. The results indicate that the use of random sampling with the sets chosen by TypiClust and ProbCover leads to either degradation or comparable results. This is in clear contrast to the benefits of random selection based on the features' norm values.

5 Conclusion and Futute Work

In this work, we have presented a novel approach to addressing the subset selection problem for deep neural network training. Leveraging insights from neural network pruning and focusing on the norm criterion of network features, we have introduced a method that significantly enhances the efficiency and effectiveness of subset selection. Our findings show a correlation between the output features' norm and accuracy, highlighting the importance of feature norms in the selection process. Moreover, we have presented a tailored algorithm that combines the norm criterion with the Gram-Schmidt process to ensure coverage of the feature space.

Furthermore, our evaluations across diverse settings and datasets validate the efficacy of our approach. By comparing our method with existing subset selection techniques, such as uniform random sampling, TypiClust, and ProbCover, we have shown consistent improvements in model performance, particularly when dealing with extremely small subsets selected from large unlabeled pools. Our method achieves new state-of-the-art results in many cases, underscoring its relevance and practical utility in scenarios where labeled data availability is limited, such as medical applications. We discusse the broader impact of our suggested aproach in the appendix.

Looking ahead, our work opens several avenues for future research. One promising direction is using the connection we draw between pruning and subset selection. In this work we used the norm criterion, but future work could explore the migration of other pruning methods to the subset selection.

Overall, we believe that we advance the field of subset selection and lays a foundation for developing more effective and informative annotation strategies in deep learning.

References

Milad Alizadeh, Shyam A Tailor, Luisa M Zintgraf, Joost van Amersfoort, Sebastian Farquhar, Nicholas Donald Lane, and Yarin Gal. Prospect pruning: Finding trainable weights at initialization using meta-gradients. *arXiv preprint arXiv:2202.08132*, 2022.

Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

Noga Bar and Raja Giryes. Pruning at initialization–a sketching perspective. *arXiv preprint arXiv:2305.17559*, 2023.

Guillaume Bellec, David Kappel, Wolfgang Maass, and Robert Legenstein. Deep rewiring: Training very sparse deep networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJ_wN01C-.

William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9368–9377, 2018.

Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. *Advances in neural information processing systems*, 33:14879–14890, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Miguel A Carreira-Perpinán and Yerlan Idelbayev. "learning-compression" algorithms for neural net pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8532–8541, 2018.

Haw-Shiuan Chang, Erik Learned-Miller, and Andrew McCallum. Active bias: Training more accurate neural networks by emphasizing high variance samples. *Advances in Neural Information Processing Systems*, 30, 2017.

Yves Chauvin. A back-propagation algorithm with optimal use of hidden units. *Advances in neural information processing systems*, 1, 1988.

Liangyu Chen, Yutong Bai, Siyu Huang, Yongyi Lu, Bihan Wen, Alan L Yuille, and Zongwei Zhou. Making your first choice: To address cold start problem in vision active learning. *arXiv preprint arXiv:2210.02442*, 2022.

Tianyi Chen, Bo Ji, Tianyu Ding, Biyi Fang, Guanyi Wang, Zhihui Zhu, Luming Liang, Yixin Shi, Sheng Yi, and Xiao Tu. Only train once: A one-shot neural network training and pruning framework. *Advances in Neural Information Processing Systems*, 34:19637–19651, 2021.

Tianyi Chen, Luming Liang, Tianyu DING, Zhihui Zhu, and Ilya Zharkov. OTOv2: Automatic, generic, user-friendly. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=7ynoX1ojPMt.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. *Advances in Neural Information Processing Systems*, 34: 11933–11944, 2021.

Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. *arXiv preprint arXiv:1906.11829*, 2019.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Pau de Jorge, Amartya Sanyal, Harkirat S Behl, Philip HS Torr, Gregory Rogez, and Puneet K Dokania. Progressive skeletonization: Trimming more fat from a network at initialization. *arXiv preprint arXiv:2006.09081*, 2020.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2018.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36, 2024.

Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *International conference on machine learning*, pp. 1183–1192. PMLR, 2017.

Chengcheng Guo, Bo Zhao, and Yanbing Bai. Deepcore: A comprehensive library for coreset selection in deep learning. In *International Conference on Database and Expert Systems Applications*, pp. 181–195. Springer, 2022.

Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient dnns. *Advances in neural information processing systems*, 29, 2016.

Guy Hacohen, Avihu Dekel, and Daphna Weinshall. Active learning on a budget: Opposite strategies suit high and low budgets. In *International Conference on Machine Learning*, pp. 8175–8195. PMLR, 2022.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

Babak Hassibi, David G Stork, and Gregory J Wolff. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, pp. 293–299. IEEE, 1993.

Yang He, Guoliang Kang, Xuanyi Dong, Yanwei Fu, and Yi Yang. Soft filter pruning for accelerating deep convolutional neural networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2018.

Yang He, Ping Liu, Ziwei Wang, Zhilan Hu, and Yi Yang. Filter pruning via geometric median for deep convolutional neural networks acceleration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4340–4349, 2019.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Wei-Ning Hsu and Hsuan-Tien Lin. Active learning by learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Speeding up convolutional neural networks with low rank expansions. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014.

Krishnateja Killamsetty, Sivasubramanian Durga, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, pp. 5464–5474. PMLR, 2021a.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021b.

Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

Jaeho Lee, Sejun Park, Sangwoo Mo, Sungsoo Ahn, and Jinwoo Shin. Layer-adaptive sparsity for the magnitude-based pruning. In *International Conference on Learning Representations*, 2020a.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018.

Namhoon Lee, Thalaiyasingam Ajanthan, Stephen Gould, and Philip H. S. Torr. A signal propagation perspective for pruning neural networks at initialization. In *ICLR*, 2020b.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv:1608.08710*, 2016.

Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rJqFGTslg.

Tianlin Liu and Friedemann Zenke. Finding trainable sparse networks through neural tangent transfer. In *International Conference on Machine Learning*, pp. 6336–6347. PMLR, 2020.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

Ekdeep Singh Lubana and Robert P Dick. A gradient flow framework for analyzing network pruning. In *International Conference on Learning Representations*, 2020.

Mark Mazumder, Colby Banbury, Xiaozhe Yao, Bojan Karlaš, William Gaviria Rojas, Sudnya Diamos, Greg Diamos, Lynn He, Alicia Parrish, Hannah Rose Kirk, et al. Dataperf: Benchmarks for data-centric ai development. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature communications*, 9(1):2383, 2018.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2016.

Hesham Mostafa and Xin Wang. Parameter efficient training of deep convolutional neural networks by dynamic sparse reparameterization. In *International Conference on Machine Learning*, pp. 4646–4655. PMLR, 2019.

Michael C Mozer and Paul Smolensky. Skeletonization: A technique for trimming the fat from a network via relevance assessment. *Advances in neural information processing systems*, 1, 1988.

Prateek Munjal, N. Hayat, Munawar Hayat, J. Sourati, and S. Khan. Towards robust and reproducible active learning using neural networks. *ArXiv*, abs/2002.09564, 2020.

Sharan Narang, Greg Diamos, Shubho Sengupta, and Erich Elsen. Exploring sparsity in recurrent neural networks. In *International Conference on Learning Representations*, 2016.

James O' Neill. An overview of neural network compression. *arXiv preprint arXiv:2006.03669*, 2020.

Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P Vetrov. Tensorizing neural networks. *Advances in neural information processing systems*, 28, 2015.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34:20596–20607, 2021.

Mansheej Paul, Brett W Larsen, Surya Ganguli, Jonathan Frankle, and Gintare Karolina Dziugaite. Lottery tickets on a data diet: Finding initializations with sparse trainable networks. In *NeurIPS*, 2022.

Abigail See, Minh-Thang Luong, and Christopher D Manning. Compression of neural machine translation models via pruning. *arXiv preprint arXiv:1606.09274*, 2016.

Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid, August 2020. Version 0.3.0.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

Burr Settles. Active learning literature survey. 2009.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35: 19523–19536, 2022.

Jingtong Su, Yihang Chen, Tianle Cai, Tianhao Wu, Ruiqi Gao, Liwei Wang, and Jason D Lee. Sanity-checking pruning methods: Random tickets can win the jackpot. *arXiv:2009.11094*, 2020.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. In *ICLR*, 2024.

Frederick Tung and Greg Mori. Clip-q: Deep network compression learning by in-parallel pruning-quantization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7873–7882, 2018.

Sunil Vadera and Salem Ameen. Methods for pruning deep neural networks. *IEEE Access*, 10:63280–63300, 2022.

Joost van Amersfoort, Milad Alizadeh, Sebastian Farquhar, Nicholas Lane, and Yarin Gal. Single shot structured pruning before training. *arXiv preprint arXiv:2007.00389*, 2020.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Proceedings of the European Conference on Computer Vision*, 2020.

Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. *arXiv preprint arXiv:2002.07376*, 2020.

Yidong Wang, Hao Chen, Yue Fan, Wang Sun, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, Heli Qi, Zhen Wu, Yu-Feng Li, Satoshi Nakamura, Wei Ye, Marios Savvides, Bhiksha Raj, Takahiro Shinozaki, Bernt Schiele, Jindong Wang, Xing Xie, and Yue Zhang. Usb: A unified semi-supervised learning benchmark for classification. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. doi: 10.48550/ARXIV.2208.07204. URL https://arxiv.org/abs/2208.07204.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 2022.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

Ofer Yehuda, Avihu Dekel, Guy Hacohen, and Daphna Weinshall. Active learning through a covering lens. *Advances in Neural Information Processing Systems*, 35:22354–22367, 2022.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34:18408–18419, 2021a.

Zeru Zhang, Jiayin Jin, Zijie Zhang, Yang Zhou, Xin Zhao, Jiaxiang Ren, Ji Liu, Lingfei Wu, Ruoming Jin, and Dejing Dou. Validating the lottery ticket hypothesis with inertial manifold theory. *Advances in Neural Information Processing Systems*, 34:30196–30210, 2021b.

Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. Coverage-centric coreset selection for high pruning rates. *arXiv preprint arXiv:2210.15809*, 2022a.

Mingkai Zheng, Shan You, Lang Huang, Fei Wang, Chen Qian, and Chang Xu. Simmatch: Semi-supervised learning with similarity matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14471–14481, 2022b.