Not All Models Are Equal: Predicting Model Transferability in a Self-challenging Fisher Space

Wenqi Shao[1,2⋆], Xun Zhao[2], Yixiao Ge[2⋆], Zhaoyang Zhang[1], Lei Yang[3], Xiaogang Wang[1], Ying Shan[2], Ping Luo[4]

[1] The Chinese University of Hong Kong [2] ARC Lab, Tencent PCG
[3] Applied Model Center, Tencent PCG [4] The University of Hong Kong
weqish@link.cuhk.edu.hk pluo.lhi@gmail.com
{yixiaoge,yingsshan}@tencent.com
⋆ Corresponding authors: Wenqi Shao, Yixiao Ge

Abstract. This paper addresses an important problem of ranking the pre-trained deep neural networks and screening the most transferable ones for downstream tasks. It is challenging because the ground-truth model ranking for each task can only be generated by fine-tuning the pre-trained models on the target dataset, which is brute-force and computationally expensive. Recent advanced methods proposed several lightweight transferability metrics to predict the fine-tuning results. However, these approaches only capture static representations but neglect the fine-tuning dynamics. To this end, this paper proposes a new transferability metric, called **S**elf-challenging **F**isher **D**iscriminant **A**nalysis (**SFDA**), which has many appealing benefits that existing works do not have. First, SFDA can embed the static features into a Fisher space and refine them for better separability between classes. Second, SFDA uses a self-challenging mechanism to encourage different pre-trained models to differentiate on hard examples. Third, SFDA can easily select multiple pre-trained models for the model ensemble. Extensive experiments on 33 pre-trained models of 11 downstream tasks show that SFDA is efficient, effective, and robust when measuring the transferability of pre-trained models. For instance, compared with the state-of-the-art method NLEEP, SFDA demonstrates an average of 59.1% gain while bringing 22.5x speedup in wall-clock time. The code will be available at https://github.com/TencentARC/SFDA.

Keywords: Transfer Learning, Model Ranking, Image Classification

1 Introduction

Due to the wide applications of DNNs [18, 17, 11], an increasing number of pre-trained models are produced by training on different source datasets (*e.g.*, ImageNet [23]), with different learning strategies (*e.g.*, self-supervised learning [16, 15]). These pre-trained models are crucial in providing a good warm start for fine-tuning on target tasks in transfer learning. An interesting question naturally emerges: given the numerous pre-trained models, how can we properly and

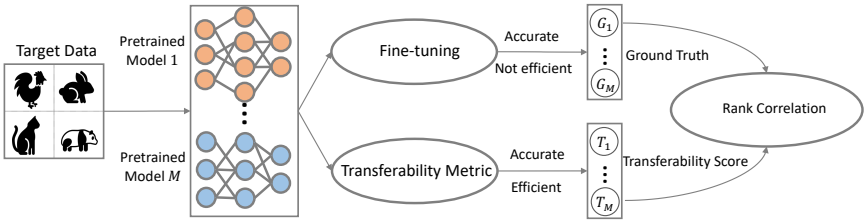


Fig. 1. The diagram of the task of ranking pre-trained models. The ground truth of the problem is to fine-tune pre-trained models and collect fine-tuning accuracy $\{G_m\}_{m=1}^{M}$. However, it is inefficient to enumerate all models in a fine-tuning procedure with a hyper-parameters sweep. An efficient approach is to adopt several lightweight transferability metrics to predict the fine-tuning results. Each metric produces transferability scores $\{T_m\}_{m=1}^{M}$ for all pre-trained models. An evaluation based on rank correlation such as Kendall's tau assesses the transferability metric.

quickly rank the models thereby selecting the best ones for a certain target task.

Ranking pre-trained models is critical for transfer learning from a practical perspective. Due to the lack of sufficient labelled data in various domains (*e.g.*, object detection), a pre-trained model with expert knowledge is highly desired for initialization of downstream tasks in these domains. Ranking pre-trained models is also challenging, as the optimal pre-trained model is usually task-specific [25]. The pre-trained model ranking method should be efficient and generic enough so that the best model can be quickly selected for downstream tasks.

The ground-truth ranking of pre-trained models is obtained by fine-tuning pre-trained models on the target dataset and ranking them by the test accuracies, as shown in Fig.1. Brute-force fine-tuning is obviously time-consuming and computationally expensive. For instance, fine-tuning one pre-trained model on one target dataset usually costs several GPU hours, selecting the optimal pre-trained model out of 10 models on 10 target tasks would cost many GPU days. Recent works propose several lightweight transferability metrics such as LEEP [29] and LogME [48] to substitute cumbersome fine-tuning procedure. Although these metrics are efficient to obtain, they simply measure the quality of pre-trained models by their static features, preventing themselves from characterizing the dynamics of the fine-tuning process, as shown in Fig.2.

In this paper, we propose a new transferability metric for ranking pre-trained models, namely **S**elf-challenging **F**isher **D**iscriminant **A**nalysis (**SFDA**). To approximate the fine-tuning process, SFDA captures two significant features of fine-tuning, i.e. classes separability and discrimination on hard examples, as shown in Fig.3. Towards this goal, we equip SFDA with a regularized FDA (Reg-FDA) module and a self-challenging mechanism. On the one hand, the Reg-FDA module projects the static features to a Fisher space where the updated features present better class separability. On the other hand, the self-challenging mechanism challenges SFDA by increasing the difficulty in separating classes,

encouraging different pre-trained models to discriminate on hard examples. Due to the above designs, SFDA behaves more like a fine-tuning process than prior arts [25, 48, 29], characterizing itself as an effective, efficient and robust transferability assessment method. Moreover, SFDA can be naturally extended to multiple pre-trained model ensembles selection because features in the Fisher space for different pre-trained models have homogeneous dimensionality. Such homogeneity makes it possible to consider the complementarity between models.

The **contributions** of this work are three-fold. (1) We propose a new transferability metric named Self-challenging Fisher Discriminant Analysis (SFDA) which is effective, efficient and robust for ranking pre-trained models. (2) SFDA can be naturally extended to multiple pre-trained model ensembles selection. (3) Extensive experiments on 33 pre-trained models produced by different types of architectures and training strategies over 11 downstream classification tasks demonstrate the effectiveness of SFDA. For example, SFDA shows an average of 59.1% gain in the rank correlation with the ground truth fine-tuning accuracy while bringing 22.5x speedup in wall-clock time compared with previous state-of-the-art method NLEEP.

2 Related Work

Transfer learning. Transfer learning [39, 37] has been extensively investigated in the form of different learning tasks such as domain adaptation [41, 26] and task transfer learning [49]. In deep learning, transfer learning mainly comes in the form of inductive transfer, which usually refers to the paradigm of adapting the pre-trained models to target tasks [46, 20]. With the rapid development of deep learning in applications, numerous pre-trained models have been stored, such as HuggingFace Transformers [42]. This paper focuses on ranking pre-trained models and selecting the best performing model for the target task. Unlike task transfer learning, where the prior knowledge in the source domain is known, ranking pre-trained models only has access to the pre-trained model itself. Hence, the method for ranking pre-trained models should be generic and efficient enough to apply to various pre-trained models and target tasks.

Transferability of pre-trained models. Measuring the transferability of pre-trained models on a target task has recently attracted much attention due to the broad practicability. LEEP [29] is a well-known early work studying this problem. It estimates the empirical joint probability of source and target label space. Hence, LEEP only works when the pre-trained model has a classification head that outputs source label probability. To pursue a more generic transferability method, NLEEP [25] generate a pseudo classification head by a Gaussian Mixture Model and LogME [48] directly model the relationship between features extracted from the pre-trained models and their labels by marginalized likelihood. Although these metrics are easy to compute, they fail to characterize the dynamics of fine-tuning process, making the performance far from idealism. This paper proposes a new transferability metric that behaves more like fine-tuning by projecting features into a self-challenging Fisher space.

3 Preliminaries

In this section, we first present the setup, ground truth and evaluation protocol of the task of ranking pre-trained models and then introduce existing transferability metrics.

Problem Setup. A target dataset with N labeled data samples denoted as $\mathcal{T} = \{(x_n, y_n)\}_{n=1}^{N}$ and M pre-trained models $\{\phi_m = (\theta_m, h_m)\}_{m=1}^{M}$ are given. Each model ϕ_m consists of a feature extractor θ_m producing a D-dimension feature (i.e. $\hat{x} = \theta_m(x) \in \mathbb{R}^D$) and a classification head h_m outputting label prediction probability given input x [17, 11]. The task of pre-trained model ranking is to generate a score for each pre-trained model thereby the best model can be identified according to the ranking list.

Fine-tuning as ground truth. After fine-tuning all pre-trained models with hyper-parameters sweep on the target training dataset, the highest scores of evaluation metrics (e.g. test accuracy) are returned. The fine-tuning accuracies of different pre-trained models are denoted as $\{G_m\}_{m=1}^{M}$, which are recognized as the ground truth of pre-trained model ranking [25, 48].

Transferability metric. For each pre-trained model ϕ_m, a transferability metric outputs a scalar score T_m by

$$T_m = \sum_{i=1}^{N} \log p(y_i|x_i; \theta_m, h_m) \tag{1}$$

where (x_i, y_i) denotes the i-th data point in target dataset $\mathcal{T}$. A larger T_m indicates that model ϕ_i can perform better on target $\mathcal{T}$. As we can see from Fig.2, transferability metrics differ in modelling label prediction probability $p(y_i|x_i; \theta_m, h_m)$. For instance, prior arts such as LEEP [29] obtain the probability based on static features $\hat{x}$, while ignoring that fine-tuning would update $\hat{x}$. Instead, our proposed SFDA is carefully designed to behave more like fine-tuning, as will be introduced in Sec.4.

Evaluation protocol. Basically, if ϕ_i has higher classification accuracy than ϕ_j (i.e. $G_i > G_j$), $T_i > T_j$ is also expected. Hence, a rank-based correlation between $\{T_m\}_{m=1}^{M}$ and $\{G_m\}_{m=1}^{M}$ is competent to evaluate the effectiveness of transferability metrics. We use *weighted Kendall's* τ_w (detailed in Appendix Sec.A.1) by following the common practice [48, 25], where larger τ_w indicates better correlation and better transferability metric.

4 Self-challenging Fisher Discriminant Analysis

We propose a Self-challenging Fisher Discriminant Analysis (SFDA) for the problem of pre-trained model ranking. SFDA consists of two critical ingredients: a module of Regularized Fisher Discriminant Analysis (Reg-FDA) and a self-challenging mechanism motivated by two important observations in fine-tuning. As shown in Fig.3(a & b), fine-tuning pre-trained model would update static features $\{\hat{x}_i\}_{i=1}^{N}$ for better classes separability. Moreover, Fig.3(d) show that

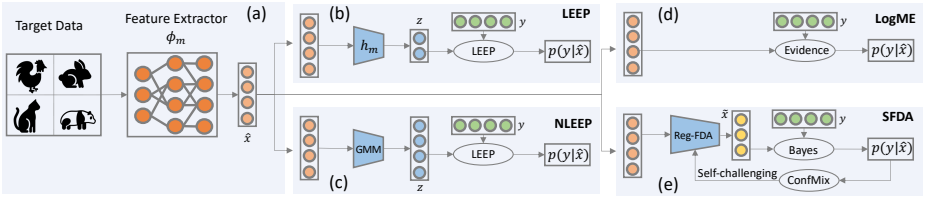


Fig. 2. Comparison of different transferability metrics including LEEP [29], NLEEP [25], LogME [48] and our proposed SFDA. (a) shows a transferability metric rely on the static representation $\hat{x}$. (b-d) show that these metrics differentiate in modeling label prediction $p(y|\hat{x})$. Specifically, LEEP, NLEEP, and LogME calculate transferability score by static features $\hat{x}$, making it difficult to approximate fine-tuning as fine-tuning would update $\hat{x}$. Our proposed SFDA behaves more like fine-tuning in terms of classes separability and discrimination on hard examples, which are achieved by the module of Reg-FDA and a self-mechanism with the proposed ConfMix noise in (e).

different pre-trained models discriminate on hard examples during fine-tuning. Hence, the Reg-FDA and self-challenging mechanism are carefully designed to encourage classes separability and discrimination on hard examples. Algorithm 1 (see Appendix Sec.A.2) illustrates the whole pipeline of our proposed SFDA.

Notation. SFDA operates on static features $\hat{x} = \theta_m(x)$. Assume the target dataset $\mathcal{T} = \{(\hat{x}_n, y_n)\}_{n=1}^{N}$ has C classes, we split it into classes. Then we have $\mathcal{T} = \{\hat{x}_n^{(1)}\}_{n=1}^{N_1} \cup \cdots \cup \{\hat{x}_n^{(C)}\}_{n=1}^{N_C}$, where $\hat{x}_n^{(c)}$ denotes the n-th instance of the c-th class and N_c denote the sample size of the c-th class ($\sum_{c=1}^{C} N_c = N$).

4.1 Regularized FDA

To approximate fine-tuning procedure, the primary thing is to transform $\{\hat{x}_i\}_{i=1}^{N}$ to a space such that the classes on target $\mathcal{T}$ are separated as much as possible as shown in Fig.3(b). Therefore, we propose Regularized Fisher Discriminant Analysis (Reg-FDA) for two reasons. First, Reg-FDA promotes classes separability by inheriting the merits from conventional FDA [28] that can maximize between scatter of classes and minimize within scatter of each class. Second, the optimization problem for Reg-FDA has a straightforward solution, which avoids gradient optimization and allows for efficient computation.

Formulation of Reg-FDA. Reg-FDA defines a transformation that projects $\hat{x} \in \mathbb{R}^D$ into $\tilde{x} \in \mathbb{R}^{D'}$ by a projection matrix $U \in \mathbb{R}^{D \times D'}$. We have $\tilde{x} := U^\mathsf{T}\hat{x}$, where

$$U = \arg\max_{U} \frac{d_B(U)}{d_W(U)} \stackrel{\text{def}}{=} \frac{|U^\mathsf{T} S_B U|}{|U^\mathsf{T}[(1-\lambda)S_W + \lambda I]U|} \tag{2}$$

In Eqn.(2), $d_B(U) = |U^\mathsf{T} S_B U|$ and $d_W(U) = |U^\mathsf{T} \tilde{S}_W U|$ represent between scatter of classes and within scatter of every class, where $\tilde{S}_W = (1-\lambda)S_W + \lambda I$ and I is an identity matrix. Moreover, $S_B = \sum_{c=1}^{C} N_c(\mu_c - \mu)(\mu_c - \mu)^\mathsf{T}$ and

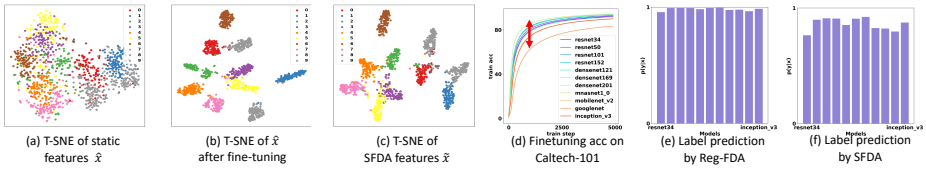


Fig. 3. (a-c) show that similar to fine-tuning our SFDA can update static features for better classes separability on the CIFAR-10 dataset. Deep models often fit easy examples in the early training stage while learning hard examples in the later training stage [50, 2]. Hence, (d) presents that different pre-trained models discriminate on hard examples. (e) and (f) show that SFDA can produce more discriminative label prediction results than Reg-FDA because the self-mechanism in SFDA encourages the different pre-trained models to discriminate on hard examples.

$S_W = \sum_{c=1}^{C} \sum_{n=1}^{N_c} (\hat{x}_n^{(c)} - \mu_c)(\hat{x}_n^{(c)} - \mu_c)^\mathsf{T}$ are between and within scatter matrix respectively. Here $\mu = \sum_{n=1}^{N} \hat{x}_n$ and $\mu_c = \sum_{n=1}^{N_c} \hat{x}_n^{(c)}$ represent the mean of data and the mean of c-th class respectively. $\lambda \in [0, 1]$ in Eqn.(2) is a regularization coefficient used to trade off the inter-class separation and intra-class compactness. Reg-FDA degrades into FDA [28] when $\lambda = 0$.

Adaptive regularization strength. We treat $\lambda \in [0, 1]$ as an regularization strength adaptive to different feature distribution. The motivation is that Reg-FDA can deal with diverse distribution of features $\{\hat{x}_i\}_{i=1}^{N}$ extracted from different pre-trained models when λ is adaptively varied. For example, self-supervised ResNet-50 with Infomin has a larger within scatter of classes than its supervised counterpart on CIFAR-10 dataset as shown in Fig.5 of Appendix Sec.A.3, implying that ResNet-50 with Infomin needs stronger supervision on minimizing within scatter of every class for better classes separation. Motivated by this intuition we instantiate λ as follows,

$$\lambda = \exp^{-a\sigma(S_W)} \text{ where } \sigma(S_W) = \max_{u \in \mathbb{R}^D, \|u\|_2 = 1} u^\mathsf{T} S_W u. \tag{3}$$

In Eqn.(3), a is a positive constant and $\sigma(S_W)$ is the largest eigenvalue of S_W by definition. A larger $\sigma(S_W)$ indicates the larger within scatter. Hence, a smaller λ should be used for the stronger supervision on minimizing within scatter.

Efficient computation of Reg-FDA. The optimization problem in Eqn.(2) can be solved efficiently as follows,

$$S_B u_k = v_k \tilde{S}_W u_k \tag{4}$$

Eqn.(4) is a generalized eigenvalues problem with v_k being the k-th eigenvalue and u_k being the corresponding eigenvector. Note that u_k constitutes the k-th column vector of the matrix U. The generalized eigenvalues problem can be efficiently solved by existing machine learning packages. Moreover, $\sigma(S_B)$ in Eqn.(3) is also easily obtained by the Iteration method, which only involves a matrix-vector product as provided in Appendix Sec.A.2.

Classification by Bayes. After obtaining projection matrix U, we then acquire updated feature representations $\{\tilde{x}_n = U^\mathsf{T}\hat{x}_n\}_{n=1}^N$ which exhibits better class separability as shown in Fig.3(c). For each class, we assume $\tilde{x}_n^{(c)} \sim \mathcal{N}(U^\mathsf{T}\mu_c, \Sigma_c)$ where Σ_c is the covariance matrix of $\{\tilde{x}_n^{(c)}\}_{n=1}^{N_c}$. Here the linear version of FDA that assumes $\Sigma_c = I$ for all $c \in [C]$ is utilized for simplicity. By Bayes theorem, given a sample $\tilde{x}_n$, the score function for label c is acquired by

$$\delta_c(\hat{x}_n) = \hat{x}_n{}^\mathsf{T} U U^\mathsf{T}\mu_c - \frac{1}{2}\mu_c{}^\mathsf{T} U U^\mathsf{T}\mu_c + \log q_c \tag{5}$$

which is a linear equation in terms of $\hat{x}_n$. In Eqn.(5), q_c is the prior probability of the c-th class, which is estimated by $q_c = N_c/N$. We normalize $\delta_c(\hat{x}_n)$, $c = 1 \cdots C$ with softmax function to obtain the final class prediction probability

$$p(y_n|\hat{x}_n) = \frac{\exp^{\delta_{y_n}(\hat{x}_n)}}{\sum_{c=1}^{C}\exp^{\delta_c(\hat{x}_n)}} \tag{6}$$

Substituting Eqn.(6) into Eqn.(1) gives us the transferability metric score of Reg-FDA.

4.2 Self-Challenging Mechanism by Noise Augmentation

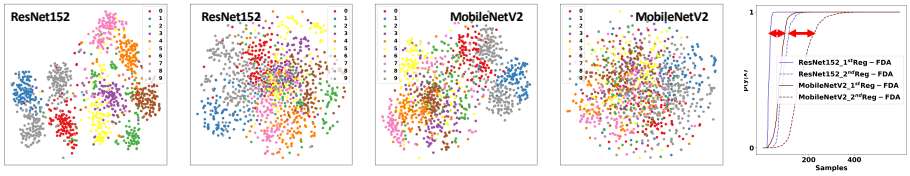
Though Reg-FDA can project static features for better classes separability, pre-trained models perform differently on hard examples during fine-tuning (Fig.3(d)). We propose a self-challenging mechanism augmented by the proposed Confidential Mix (ConfMix) noise to encourage pre-trained models to discriminate on hard examples. The self-challenging framework consists of a two-stage Reg-FDA. In the first stage, the Reg-FDA provides a confidential probability of correctly classifying each sample through Eqn.(6). In the second stage, the Reg-FDA challenges the classification accuracy in the first stage by ConfMix.

Formulation of ConfMix. We denote $p(y_n|\hat{x}_n)$ in Eqn.(6) as p_n. Since p_n represents the confidential probability of correctly classifying n-th sample in target dataset, a smaller p_n indicates larger difficulty in classifying $(\hat{x}_n, y_n)$. To increase the difficulty of classification, ConfMix builds Reg-FDA in the second stage on the convex combination of the underlying sample and the mean of its outer classes by p_n, as written by

$$\bar{x}_n = p_n\hat{x}_n + (1 - p_n)\mu_{c\neq y_n} \tag{7}$$

where $\mu_{c\neq y_n} = \frac{1}{N-N_{y_n}}\sum_{n=1,y_n\neq c}^{N}\hat{x}_n$ denotes the outer classes mean relative to the c-th class.

Analysis of ConfMix. We show that ConfMix increases classification difficulty and improves the discrimination on hard examples for pre-trained models. Through Eqn.(7), $\hat{x}_n$ is moved to its outer classes by ConfMix. Hence, different classes would be closer to each other, increasing the difficulty in separating classes by Reg-FDA. For example, Fig.4 show that ConfMix encourages separate



(a) T-SNE of features $\hat{x}$ (b) T-SNE of features $\hat{x}$ with ConfMix (c) T-SNE of features $\hat{x}$ (d) T-SNE of features $\hat{x}$ with ConfMix (e) Label prediction of Reg-FDA in two stages

Fig. 4. (a-d) show that our proposed ConfMix noise reduces separability between classes on both ResNet-152 and MobileNetV2. (e) shows that ConfMix increases the classification difficulty and encourages pre-trained models to differentiate on hard examples. The results are obtained on a fraction of the CIFAR-10 dataset.

feature distribution between classes. Moreover, the final classification probability $p(y_n|\bar{x}_n)$ obtained by Reg-FDA in the second stage is lower than $p(y_n|\hat{x}_n)$ in the first stage, which means the classification by Reg-FDA becomes more difficult. In addition, we also observe from Fig.3(e & f) and Fig.4(e) that the difference between pre-trained models on hard examples are enlarged.

4.3 Extension to Top-k Model Ensembles Selection

Top-k model ensembles selection utilizes k pre-trained models to perform ensemble transfer learning. The k models are selected to obtain the best performance on target tasks. The main challenge of top-k model selection is the heterogeneous feature representations extracted from different pre-trained models. For example, ResNet-50 outputs a feature with the dimension of 2048 while the features extracted from ResNet-18 are 512-D. Such heterogeneity makes it difficult to compare different pre-trained models. To simplify the problem, recent works propose to select k models by top-k ranked transferability metrics. Despite the simplicity, it fails to consider the relationship between models. In other words, transferability metrics can only measure how well a single model performs on a target task. However, the complementarity between models should also be considered in ensemble transfer learning [6, 30].

Homogeneous features by SFDA. Fortunately, our proposed SFDA can deal with heterogeneous feature representations by the projection of U. By SFDA, we have $\tilde{x} = U^{\mathsf{T}}\hat{x} \in \mathbb{R}^{D'}$ where $D' = \min\{D, C-1\}$. Note that $\hat{x} \in \mathbb{R}^{D}$ where D varies for different models. Fortunately, the projected features of different models by SFDA have homogeneous dimension of $D' = C - 1$. Here we assume $D \geq C - 1$ which is a common case in practice.

Complementarity score. By SFDA, we can collect an ensemble of homogeneous features on a model basis given a sample, as denoted by $F^{\text{ens}} = [\tilde{x}^1, \cdots, \tilde{x}^M] \in \mathbb{R}^{M \times D'}$. Hence, each model is now represented by each row of F^{ens}. An ablative approach [51] is employed to evaluate how a model is complementary to other models, as given by

$$T_m^{\text{com}} = \|F^{\text{ens}}\|_* - \|F^{\text{ens}} \odot 1_m\|_* \tag{8}$$

where T_m^{com} denotes the complementarity score of the m-th model, $1_m\mathbb{R}^M$ is a mask vector with m-th entry of 0 and 1 elsewhere. In Eqn.(8), $\|\cdot\|_*$ denotes nuclear norm which is a relaxation of rank of a matrix Therefore T_m^{com} measures how the m-th model influences the rank of F^{ens}. A larger T_m^{com} implies that the m-th model is more important for the complementarity in F^{ens}. Evaluation of the complementarity score is illustrated in Algorithm 2 of Appendix Sec.A.4, where the final score is averaged over a fraction of target samples.

Ensemble score. The total ensemble score for selecting top-k models is determined by combining SFDA and complementarity scores. The former evaluates the transferability of a single model, and the latter measures the complementarity between models. Putting them together yields

$$T_m^{\text{ens}} = rT_m^{\text{SFDA}} + (1-r)T_m^{\text{com}} \tag{9}$$

where T_m^{SFDA} and T_m^{com} denotes the transferability score and complementarity score. r is a weight ratio to balance two scores. Empirically, we set $r = 0.5$. We select top-k ranked models by T_m^{ens} to perform ensemble transfer learning.

5 Experiments

This section evaluates our method SFDA on different categories of pre-trained models, including both supervised and self-supervised Convolutional Neural Network (CNN) models and recently-popular vision transformer models [11, 34] (see Appendix B.2). Moreover, we also show the effectiveness of SFDA in the top-k ensembles selection of pre-trained models. The evaluation is performed on 11 standard classification benchmark in transfer learning. Finally, we conduct an ablation study to analyze our SFDA (also see Appendix B.4).

5.1 Benchmarks

Target datasets. We adopt 11 classification benchmarks widely used in the transfer learning study, including FGVC Aircraft [27], Caltech-101 [14], Stanford Cars [21], CIFAR-10 [22], CIFAR-100 [22], DTD [10], Oxford 102 Flowers [31], Food-101 [4], Oxford-IIIT Pets [32], SUN397 [44], and VOC2007 [13]. These datasets cover a broad range of classification tasks, which includes scene, texture, and coarse/fine-grained image classification.

Ground truth. We can obtain the ground-truth ranking by fine-tuning all pre-trained models with hyper-parameters sweeping on target datasets. Here we consider the normal training setting which is more general in practice than other sparse training methods [53]. Details of target datasets and fine-tuning schemes are described in Appendix Sec.B.1

5.2 Evaluation on Supervised CNN Models

Models. We first evaluate the performance of transferability metrics on ranking pre-trained supervised CNN models. We select 11 widely-used models including

Table 1. Comparison of different transferability metrics on supervised CNN models in rank correlation τ_w with the ground truth and the wall-clock time. A larger τ_w indicates the better transferability assessment of pre-trained models. The best results are denoted in bold. A shorter wall-clock time suggests that the transferability metric is more efficient to obtain. Our proposed SFDA achieves the best transferability prediction over 11 target tasks while being much more efficient than NLEEP.

	Aircraft	Caltech	Cars	CF-10	CF-100	DTD	Flowers	Food	Pets	SUN	VOC
Weighted Kendall's tau τ_w											
LEEP	-0.234	0.605	0.367	0.824	0.677	0.486	-0.243	0.491	0.389	0.701	0.446
LogME	0.506	0.435	**0.576**	0.852	0.692	0.647	0.111	0.385	0.411	0.511	0.478
NLEEP	0.495	0.661	0.265	0.806	0.823	**0.777**	0.215	0.624	0.599	**0.807**	0.654
SFDA	**0.615**	**0.737**	0.487	**0.949**	**0.866**	0.597	**0.542**	**0.815**	**0.734**	0.703	**0.763**
Wall-Clock Time (s)											
LEEP	5.1	4.9	8.3	22.3	23.8	3.5	3.8	37.1	3.9	21.1	4.8
LogME	11.5	11.0	18.3	23.5	36.9	8.7	10.9	54.0	9.4	51.8	9.7
NLEEP	253.8	488.7	973.8	1.1e4	1.7e4	146.0	294.0	2.0e4	580.8	8.6e3	678.8
SFDA	91.9	246.2	274.6	576.6	617.9	171.9	167.2	703.7	181.2	560.5	75.0

ResNet-34 [17], ResNet-50 [17], ResNet-101 [17], ResNet-152 [17], DenseNet-121 [19], DenseNet-169 [19], DenseNet-201 [19], MNet-A1 [38], MobileNetV2 [33], GoogleNet [35], and InceptionV3 [36]. All these models are trained on ImageNet dataset[23]. We fine-tune these models on the 11 target datasets to obtain the ground truth (original results are shown in Appendix Sec.B.1). In addition, the transferability scores of metrics including LEEP, NLEEP, LogME and our SFDA are also calculated by following the paradigm in Fig.2. We re-implement these algorithms in our framework for a fair comparison.

Performance Comparison. We assess transferability metrics by weighted Kendall's tau τ_w that measures the rank correlation between ground truth and metrics scores. We compare SFDA with previous LEEP, LogME, and NLEEP. As shown in Table 1, SFDA achieves the best rank correlation τ_w with the ground truth on 8 target datasets. For example, SFDA outperforms NLEEP by 0.327, 0.191, and 0.160 rank correlation τ_w on Flowers, Food, and Pets, respectively. On these three datasets, the relative improvements are 152.1%, 30.6%, and 26.7%, respectively, showing the effectiveness of our SFDA in measuring the transferability of pre-trained models. On the other hand, for the remaining 3 target datasets (i.e. Cars, DTD, and SUN397), our SFDA still has a marginal gap compared to the best-performing transferability metric.

Wall-clock time comparison. We also provide wall-clock time comparison in Table 1. The wall-clock time measures how long it takes for each metric to calculate the transferability scores of all models on a target dataset. We can see that both LEEP and LogME are fast to obtain on all target tasks. However, LEEP and LogME are not stable in measuring the transferability score of pre-trained models for all target tasks. For example, the rank correlation τ_w of

Table 2. Comparison of different transferability metrics on self-supervised CNN models regarding τ_w and the wall-clock time. Our proposed SFDA achieves the best transferability assessment over 11 target tasks and exhibits higher efficiency than NLEEP.

	Aircraft	Caltech	Cars	CF-10	CF-100	DTD	Flowers	Food	Pets	SUN	VOC
Weighted Kendall's tau τ_w											
LogME	0.223	0.051	0.375	0.295	-0.008	0.627	0.604	0.570	0.684	0.217	0.158
NLEEP	-0.029	**0.525**	0.486	-0.044	0.276	0.641	0.534	0.574	**0.792**	0.719	-0.101
SFDA	**0.254**	0.523	**0.515**	**0.619**	**0.548**	**0.749**	**0.773**	**0.685**	0.586	0.698	**0.568**
Wall-Clock Time (s)											
LogME	33.5	33.0	72.7	89.3	116.0	15.0	41.0	140.1	40.9	112.5	16.7
NLEEP	581.7	787.7	1.6e3	8.0e3	2.1e4	332.7	322.3	1.1e4	186.5	3.9e4	678.8
SFDA	300.7	316.4	553.4	504.1	753.1	170.2	335.1	980.7	157.7	992.5	134.7

LogME and LEEP on multiple target datasets such as Flowers, Food, and Pets are lower than 0.5, implying that they are ineffective in measuring transferability on these downstream tasks. Moreover, NLEEP performs well on a large number of target tasks. But the computation of NLEEP on some target datasets costs even several hours, which is comparable with the fine-tuning procedure.

As we can see from Table 1, our SFDA can measure transferability of models well while only requiring several hundred seconds for computation on all target tasks, achieving a better trade-off between assessment of transferability and computation consumption than other methods.

5.3 Evaluation on Self-Supervised CNN Models

Models. In transfer learning, self-supervised pre-trained models generally have better transferability than their supervised counterpart [16, 12]. Moreover, the different self-supervised algorithms would provide diverse feature representation for a downstream task [12]. Hence, it is essential to investigate selecting the best self-supervised pre-trained model for a target task. To this end, we build a pool of self-supervised pre-trained models with ResNet-50 [17], including BYOL [15], Deepclusterv2 [5], Infomin [40], MoCov1 [16], MoCov2 [9], Instance Discrimination [43], PCLv1 [24], PCLv2 [24], Selav2 [3], SimCLRv1 [7], SimCLRv2 [8], and SWAV [10]. Our goal is to rank these self-supervised models on 11 downstream tasks in Sec.5.1.

Performance Comparison. We compare our SFDA with LogME and NLEEP on transferability assessment in rank correlation τ_w with the ground truth. Because self-supervised models usually do not have a classifier, LEEP is not included for comparison as it relies on a classifier to calculate transferability score. As shown in Table 2, SFDA still performs consistently well in measuring the transferability of self-supervised models. On the contrary, LogME and NLEEP fail on some target tasks. For example, LogME and NLEEP have a negative of small positive rank correlation on Aircraft and CIFAR-100. Averaging τ_w over

Table 3. Separate effects of components in SFDA. Three variants of SFDA are considered: (1) LogME + ConfMix; (2) SFDA w/o ConfMix; (3) SFDA with $\lambda = 0.5$ in Eqn.(3). We see that both Reg-FDA and the self-challenging mechanism achieved by ConfMix are crucial to SFDA.

Variants	Aircraft	Caltech	Cars	CF-10	CF-100	DTD	Flowers	Food	Pets	SUN	VOC
(1)	0.408	0.324	0.365	0.924	0.571	0.328	0.023	0.466	0.390	0.419	0.695
(2)	0.481	0.676	0.403	0.887	0.773	0.471	**0.668**	0.812	0.652	0.606	0.721
(3)	0.424	0.627	0.178	0.931	0.828	0.458	0.228	0.802	0.409	0.651	0.650
SFDA	**0.615**	**0.737**	**0.487**	**0.949**	**0.866**	**0.597**	0.542	**0.815**	**0.734**	**0.703**	**0.763**

Table 4. Comparison between SFDA and re-training head in terms of τ_w. Our proposed SFDA generally performs better than re-training head over 11 target tasks.

	Aircraft	Caltech	Cars	CF-10	CF-100	DTD	Flowers	Food	Pets	SUN	VOC
Weighted Kendall's tau τ_w											
Re-Head	-0.008	**0.590**	**0.666**	0.583	0.501	0.721	0.661	**0.787**	0.**703**	0.637	0.490
SFDA	**0.254**	0.523	0.515	**0.619**	**0.548**	**0.749**	**0.773**	0.685	0.586	**0.698**	**0.568**
Wall-Clock Time (s)											
Fine-tune	3.3e5	2.8e5	2.7e5	2.5e5	2.5e5	3.1e5	3.8e5	2.6e5	2.9e5	3.9e5	2.9e5
Re-Head	2211	2198	2246	2219	2215	2210	2173	2211	2232	2228	2375
SFDA	300.7	316.4	553.4	504.1	753.1	170.2	335.1	980.7	157.7	992.5	134.7

11 target tasks, the improvement of SFDA (0.593) over LogME (0.345) and NLEEP (0.397) are 71.9% and 49.4%, respectively. The results demonstrate the effectiveness of our SFDA in transferability assessment.

Wall-clock time comparison. As shown in Table 2, LogME and SFDA usually take tens of seconds to calculate transferability score on all target tasks, while NLEEP may cost several hours on some target datasets. Though LogME is fast, it is not stable, as mentioned above. Instead, our SFDA adapts to different pre-trained models and requires several hundred seconds on all target tasks. Hence, SFDA is still effective and efficient in evaluating the transferability of self-supervised models.

5.4 Extension to Top-k Model Ensembles Selection

Our proposed SFDA is competent for the problem of top-k model ensembles selection as SFDA makes features extracted from different models homogeneous. Due to the homogeneity, the complementarity between models is considered by SFDA through Eqn.(8), which we denoted as SFDA$^{\text{com}}$ as shown in Table 10 of Appendix. To verify the effectiveness of SFDA$^{\text{com}}$ in top-k model ensembles selection, we compare it with baselines [1, 47] that select k models by top-k ranked LogME, NLEEP. We also experimented on selecting k models by top-k

Table 5. Comparison on all CNN models in Sec.5.2 and Sec.5.3. The weighted Kendall's tau τ_w is used to assess transferability metrics. Our SFDA is still better at measuring the transferability of pre-trained models than LogME and NLEEP.

Method	Aircraft	Caltech	Cars	CF-10	CF-100	DTD	Flowers	Food	Pets	SUN	VOC
LogME	0.168	0.033	**0.506**	0.687	0.507	0.580	0.301	0.535	0.629	0.284	0.531
NLEEP	-0.026	0.714	0.099	0.491	0.653	**0.766**	0.373	0.233	0.768	**0.716**	0.637
SFDA	**0.350**	**0.791**	0.450	**0.748**	**0.803**	0.544	**0.741**	**0.743**	**0.788**	0.537	**0.798**

ranked SFDA. When k models are selected, we perform ensemble fine-tuning on 11 target tasks respectively by following the paradigm in [1]. The final ensemble fine-tuning accuracy is used to compare different ensemble transferability metrics. Wee can see from Table 10 in Appendix B.3 that SFDA$^{\text{com}}$ leads to higher fine-tuning accuracy on most target tasks than other metrics.

5.5 Ablation Analysis

The effect of Reg-FDA and self-challenging mechanism. SFDA consists of a Reg-FDA module and a self-challenging mechanism achieved by ConfMix. Here we study their separate effects on transferability assessment. To this end, we evaluate the transferability of supervised CNN models in Sec.5.2 with the following variants of SFDA, (1) SFDA with Reg-FDA replaced by LogME because LogME can also output a label prediction probability; (2) SFDA w/o ConfMix; and (3) SFDA with a fixed regularization coefficient $\lambda = 0.5$ in Eqn.(2). The results are reported in Table 3. From (1), we see that 'LogME + ConfMix' performs worse than SFDA ('Reg-FDA + ConfMix'), indicating that the ConfMix cooperates better with Reg-FDA than LogME. The main reason is that our Reg-FDA projects features for better class separability, and ConfMix increases the difficulty in separating classes features. But LogME has nothing to do with classes separability. Moreover, comparing (2) and SFDA, we conclude that self-mechanism achieved by ConfMix can consistently improve the performance on 11 downstream tasks. Lastly, an adaptive regularization term in Eqn.(3) helps SFDA deal with various feature distributions. Fixing the regularization strength leads to worse transferability assessment as shown in Table 3 (3).

Comparison with re-training Head. Re-training head is a widely-adopted tool to measure how well the features extracted from the pre-trained model can predict their labels by a classification head. It freezes the extracted features and trains the classification head only. After training, the head produces the labels of features. Hence, we can obtain re-training head accuracy. Though re-training head simplifies the fine-tuning process, it is still built on static representations and requires a grid search for hyper-parameters such as learning rate and regularization strength. Hence, re-training head is less efficient and effective than our SFDA. Table 4 shows that SFDA is more effective in measuring the transfer-

ability of pre-trained models than the re-training head. In particular, our SFDA is more than 600x faster than brute-force fine-tuning in running time.

Performance on total pre-trained model hubs. In practice, we may have a large-scale pre-trained model hubs rather than categorized ones in Sec.5.2 - 5.3. In this case, we need to rank a variety of pre-trained models. To this end, we consolidate the pre-trained models from supervised CNN models in Sec.5.2 and self-supervised CNN models in Sec.5.3 into one group (including 23 pre-trained models in total), then applying the ranking methods on it. The results are shown in Table 5. The results further reveal that our SFDA performs consistently well on a larger group of pre-trained models compared to LogME and NLEEP.

6 Conclusions and Discussions

The rapid development of deep learning produces many deep models pre-trained on different source datasets with various learning strategies. Given numerous pre-trained models, it is practical to consider how to rank them and screen the best ones for target tasks. In this work, we answer this question by proposing a new transferability metric named Self-challenging Fisher Discriminant Analysis (SFDA). SFDA build upon a regularized Fisher Discriminate Analysis and a self-challenging mechanism. Compared with prior arts, SFDA behaves more like fine-tuning in terms of classes separability and discrimination on hard examples, characterizing itself as an efficient, effective and robust transferability assessment method. Moreover, SFDA can be naturally extended to multiple pre-trained model ensemble selection where complementarity between models is essential for better ensemble performance on downstream tasks. SFDA measures transferability by mimicking the fine-tuning procedure for downstream classification tasks.

Discussions. Although SFDA is fast and effective in measuring transferability of pre-trained models, it can only used in downstream classification tasks. Several future works are worth investigating in pursuit of a more universal metric. Firstly, an interesting future work would be how to extend SFDA in other types of target tasks such as regression tasks by characterizing the features of fine-tuning on these tasks. Secondly, investigating SFDA in out-of-distribution setting [45, 52] could also be a fruitful future direction.

Acknowledgement. We thank anonymous reviewers from the venue ECCV 2022 for their valuable comments. We also thank Xiuzhe Wu and Ruichen Luo for their helpful discussions. Ping Luo is supported by the General Research Fund of HK No.27208720, No.17212120, and No.17200622.

References

1. Agostinelli, A., Uijlings, J., Mensink, T., Ferrari, V.: Transferability metrics for selecting source model ensembles. arXiv preprint arXiv:2111.13011 (2021)
2. Arazo, E., Ortego, D., Albert, P., O'Connor, N., McGuinness, K.: Unsupervised label noise modeling and loss correction. In: International conference on machine learning. pp. 312–321. PMLR (2019)
3. Asano, Y.M., Rupprecht, C., Vedaldi, A.: Self-labelling via simultaneous clustering and representation learning. arXiv preprint arXiv:1911.05371 (2019)
4. Bossard, L., Guillaumin, M., Gool, L.V.: Food-101–mining discriminative components with random forests. In: European conference on computer vision. pp. 446–461. Springer (2014)
5. Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: Proceedings of the European conference on computer vision (ECCV). pp. 132–149 (2018)
6. Chang, K.H.: Complementarity in data mining. University of California, Los Angeles (2015)
7. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
8. Chen, T., Kornblith, S., Swersky, K., Norouzi, M., Hinton, G.E.: Big self-supervised models are strong semi-supervised learners. Advances in neural information processing systems pp. 22243–22255 (2020)
9. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
10. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3606–3613 (2014)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. Ericsson, L., Gouk, H., Hospedales, T.M.: How well do self-supervised models transfer? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5414–5423 (2021)
13. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision (2), 303–338 (2010)
14. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: 2004 conference on computer vision and pattern recognition workshop. pp. 178–178. IEEE (2004)
15. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al.: Bootstrap your own latent-a new approach to self-supervised learning. Advances in Neural Information Processing Systems pp. 21271–21284 (2020)
16. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9729–9738 (2020)

17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
18. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
19. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
20. Kornblith, S., Shlens, J., Le, Q.V.: Do better imagenet models transfer better? In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2661–2671 (2019)
21. Krause, J., Deng, J., Stark, M., Fei-Fei, L.: Collecting a large-scale dataset of fine-grained cars (2013)
22. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems (2012)
24. Li, J., Zhou, P., Xiong, C., Hoi, S.C.: Prototypical contrastive learning of unsupervised representations. arXiv preprint arXiv:2005.04966 (2020)
25. Li, Y., Jia, X., Sang, R., Zhu, Y., Green, B., Wang, L., Gong, B.: Ranking neural checkpoints. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2663–2673 (2021)
26. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International conference on machine learning. pp. 97–105. PMLR (2015)
27. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013)
28. Mika, S., Ratsch, G., Weston, J., Scholkopf, B., Mullers, K.R.: Fisher discriminant analysis with kernels. In: Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468). pp. 41–48. Ieee (1999)
29. Nguyen, C., Hassner, T., Seeger, M., Archambeau, C.: Leep: A new measure to evaluate transferability of learned representations. In: International Conference on Machine Learning. pp. 7294–7305. PMLR (2020)
30. Nguyen, H., Chang, M.: Complementary ensemble learning. arXiv preprint arXiv:2111.08449 (2021)
31. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing. pp. 722–729. IEEE (2008)
32. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.: Cats and dogs. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3498–3505. IEEE (2012)
33. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
34. Shao, W., Ge, Y., Zhang, Z., Xu, X., Wang, X., Shan, Y., Luo, P.: Dynamic token normalization improves vision transformer. arXiv preprint arXiv:2112.02624 (2021)
35. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9 (2015)

36. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2818–2826 (2016)
37. Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., Liu, C.: A survey on deep transfer learning. In: International conference on artificial neural networks. pp. 270–279. Springer (2018)
38. Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V.: Mnasnet: Platform-aware neural architecture search for mobile. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2820–2828 (2019)
39. Thrun, S., Pratt, L.: Learning to learn: Introduction and overview. In: Learning to learn, pp. 3–17. Springer (1998)
40. Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., Isola, P.: What makes for good views for contrastive learning? Advances in Neural Information Processing Systems pp. 6827–6839 (2020)
41. Wang, M., Deng, W.: Deep visual domain adaptation: A survey. Neurocomputing pp. 135–153 (2018)
42. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al.: Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771 (2019)
43. Wu, Z., Xiong, Y., Yu, S.X., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3733–3742 (2018)
44. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: 2010 IEEE computer society conference on computer vision and pattern recognition. pp. 3485–3492. IEEE (2010)
45. Yang, J., Zhou, K., Li, Y., Liu, Z.: Generalized out-of-distribution detection: A survey. arXiv preprint arXiv:2110.11334 (2021)
46. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? (2014)
47. You, K., Liu, Y., Wang, J., Jordan, M.I., Long, M.: Ranking and tuning pre-trained models: A new paradigm of exploiting model hubs. arXiv preprint arXiv:2110.10545 (2021)
48. You, K., Liu, Y., Wang, J., Long, M.: Logme: Practical assessment of pre-trained models for transfer learning. In: International Conference on Machine Learning. pp. 12133–12143. PMLR (2021)
49. Zamir, A.R., Sax, A., Shen, W., Guibas, L.J., Malik, J., Savarese, S.: Taskonomy: Disentangling task transfer learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3712–3722 (2018)
50. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning (still) requires rethinking generalization. Communications of the ACM **64**(3), 107–115 (2021)
51. Zhou, K., Yang, Y., Qiao, Y., Xiang, T.: Domain adaptive ensemble learning. IEEE Transactions on Image Processing pp. 8008–8018 (2021)
52. Zhou, X., Lin, Y., Pi, R., Zhang, W., Xu, R., Cui, P., Zhang, T.: Model agnostic sample reweighting for out-of-distribution learning. In: International Conference on Machine Learning. pp. 27203–27221. PMLR (2022)
53. Zhou, X., Zhang, W., Xu, H., Zhang, T.: Effective sparsification of neural networks with global sparsity constraint. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3599–3608 (2021)