ZooD: Exploiting Model Zoo for Out-of-Distribution Generalization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent advances on large-scale pre-training have shown great potentials of leveraging a large set of Pre-Trained Models (PTMs) for improving Out-of-Distribution (OoD) generalization, for which the goal is to perform well on possible unseen domains after fine-tuning on multiple training domains. However, maximally exploiting a zoo of PTMs is challenging since fine-tuning all possible combinations of PTMs is computationally prohibitive while accurate selection of PTMs requires tackling the possible data distribution shift for OoD tasks. In this work, we propose ZooD, a paradigm for PTMs ranking and ensemble with feature selection. Our proposed metric ranks PTMs by quantifying inter-class discriminability and inter-domain stability of the task data features extracted by the PTMs in a leave-one-domain-out cross-validation manner. The top-K ranked models are then aggregated for the target OoD task. To avoid accumulating noise induced by model ensemble, we propose an efficient variational EM algorithm to select informative features. We evaluate our paradigm on a diverse model zoo consisting of 35 models for various OoD tasks and demonstrate: (i) model ranking is better correlated with fine-tuning ranking than previous methods and up to 9859x faster than brute-force fine-tuning; (ii) OoD generalization outperforms the state-of-the-art methods and accuracy on most challenging task DomainNet is improved from 46.5% to 50.6%.

1 Introduction

Training and test data being Independent and Identically Distributed (IID) is a primary assumption behind most machine learning systems. However, this assumption does not hold in many real-world scenarios as real-world is marred with continuous distribution shifts [26]. Machine learning models encounter serious performance degradation [8, 20, 22] in such Out-of-Distribution (OoD) scenarios. To alleviate the accuracy degradation caused by distribution shifts, numerous algorithms have been proposed [4, 1, 27, 31, 5, 28, 45, 19, 13, 33, 6]. Recently, Gulrajani and Lopez-Paz [18] have argued for the systematic comparisons of OoD algorithms and introduced a standard and rigorous test bed called DomainBed. Their experimental comparison has raised some doubts about the effectiveness of OoD algorithms since they often fail to outperform the simple empirical risk minimization.

On the other hand, recent works [21, 2, 53, 42] have shown the advantages of pre-training for improving OoD generalization, i.e., learning from multiple training domains and being well applied to an unseen domain. The availability of a large set of Pre-Trained Models (PTMs) provides a possibility for solving various OoD tasks. However, it is challenging to sufficiently exploit the power of a model zoo (a large set of PTMs). One naive approach could be fine-tuning all possible combinations of PTMs on the target dataset and choosing the best performing one. However, naive fine-tuning is a costly and inflexible method with the risk of over-fitting [55]. Fine-tuning may also require exhaustive hyper-parameters search. Besides, fine-tuning becomes computationally prohibitive for a model zoo consisting of several hundred models and a dataset containing a large number of examples, making it impossible to use at any practical scale.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

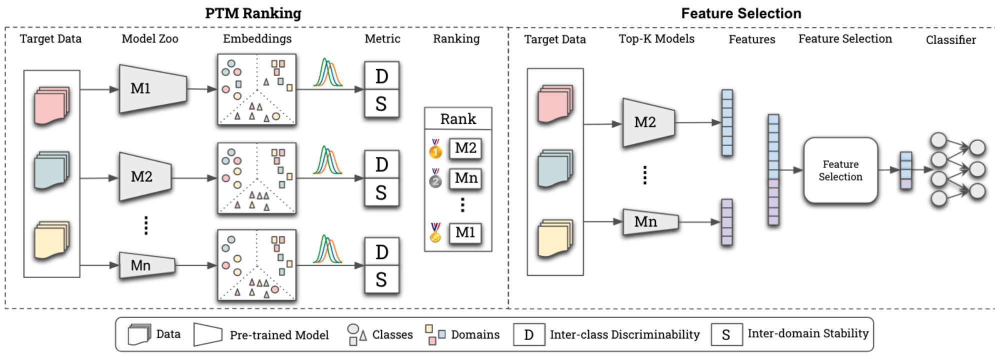

Figure 1: An overview of ZooD. Given a task with multiple training domains, the model ranking component evaluates and selects the top-K models that generalize well on this task. The features from selected models are then aggregated and denoised based on the feature selection component.

Recently, many ranking metrics have been proposed to estimate the transferability of models under IID assumption [7, 47, 37, 55, 54]. However, ranking a zoo of models for generalization on unseen distribution shifts is more challenging compared with IID setting. Moreover, even if a metric can correctly evaluate the transferability of each PTM, simply using the best model will not fully utilize rich knowledge present in a zoo of models. But the problem is even more serious that the most transferable model will include some noise, because noise and invariant features are undistinguishable in the sense that they are all stable across domains. Previous study [52] also pointed this out and emphasized the necessity of feature denoising. Therefore, if we leverage the model zoo by assembling relatively transferable models, the accumulation of noise features may increase memory use and hurt the predictive performance.

To solve the aforementioned problems, we propose ZooD, a paradigm to rank and aggregate a **Zoo** of PTMs for **OoD** generalization. An overview of our method is shown in Figure 1. Given a classification task with multiple training domains, to evaluate the generalization capability of each model, we quantify both the inter-class discriminability and inter-domain stability of the features extracted from each PTM in a leave-one-domain-out cross-validation manner, i.e., choosing one domain as the validation domain and each domain rotating as the validation domain, which is critical for identifying models that can extract domain-invariant features. Each PTM in the zoo is ranked by this quantification. ZooD then continues with model aggregation consisting of model ensemble and feature selection. By introducing latent masks over candidate features, an efficient EM algorithm is proposed to select informative features. To tackle the intractability of the posterior, variational approximation to the true posterior using a factorizable distribution is derived. We further extent it to large-scale datasets by building a local estimator under the stochastic approximation [43].

To demonstrate the efficacy of our method, we have performed extensive experiments with 35 diverse PTMs and 7 OoD datasets. First, we show that our ranking metric is strongly correlated with the fine-tuning performance of PTMs compared with existing IID metrics. Second, we illustrate the outstanding performance of ZooD on OoD datasets. For instance, on Office-Home, we get 85.1% average accuracy compared with previous SOTA of 70.6%. Lastly, we show the speedup of our method compared with brute-force fine-tuning. ZooD gives a maximum speedup of $\approx 10000\times$ (0.27 GPU hours vs 2662.27 GPU hours), making it practical and scalable.

Finally, to speed-up research and make our work more reproducible, we have devised a test bench consisting of extracted features, fine-tuning accuracy results, and ranking scores for all 35 PTMs in our model zoo. This testbed can help future research as the process of getting fine-tuning accuracy results based on DomainBed [18] for a zoo of models is computationally expensive. For instance, fine-tuning 35 models on all 7 OoD datasets costed approximately 35140 GPU hours (equivalent to 1464 GPU days or 4 GPU years). Concisely, our contributions are as follows:
(i) We propose an efficient and scalable ranking metric to gauge the generalization-ability of PTMs for unseen domains.
(ii) Using EM, we propose a method for selecting informative features and discarding invariant but noisy features in an ensemble of models.
(iii) We have established a test bed for PTMs on 7 OoD datasets, including features extracted by 35 PTMs in our model zoo, fine-tuning accuracy results and model ranking scores by different methods.

2 Related Work

Pre-training for OoD generalization. To tackle the problem of distribution shifts between training and test data, various OoD methods [4, 1, 27, 31, 15, 11, 5, 28, 45, 13, 33, 6] have been proposed with the aim to learn invariant representations across different environments. However, a standard evaluation [18] of many OoD algorithms shows that they do not significantly outperform simple ERM. On the other hands, recent works have shown effectiveness of pre-trained models for OoD generalization. Yi et al. [53] theoretically showed that adversarially pre-trained models also perform better for OoD generalization. Anonymous [3] performed a large-scale empirical analysis and show that the right choice of pre-trained models can achieve SOTA results. They also showed IID performance is not a good indicator of OoD performance and emphasized on the importance of model selection. Albuquerque et al. [2] showed the importance of feature extractor by proposing a new OoD-based pretext task for SSL pre-training that can outperform supervised training. CLIP [42] demonstrated that large-scale pre-training on a dataset of image-text pairs results in much more robust models for downstream tasks with various distribution shifts. Our work is based on these observations and we aim to facilitate utilization of PTMs by proposing an efficient metric as well as efficient feature ensemble and selection method.

Ranking pre-trained models by metric design. Large-scale, ever-increasing and evolving nature of PTMs requires a low-cost and flexible selection metric. Recently, a number of metrics have been introduced to estimate transferability of source-task-learned representations for target task under IID conditions. H-score [7] estimates the transferability by finding the relationship between extracted features and target class labels. NCE [47] proposes to estimate transferability via measuring conditional entropy between source and target labels. LEEP [37] simplifies NCE by using the joint distribution of source and target labels to estimate log expected empirical prediction. LogME [55, 54] estimates maximum value of label evidence given features from pre-trained models. The use of features instead of labels makes LogME more generalizable as it can be employed beyond classification. However, these transferability metrics focus on determining the compatibility of source-task-learned representations for the target task. We, on the other hand, aim to compute stability of these features across domains in addition to source-target transferability.

Ensemble and feature selection. Early works have shown that model ensemble can significantly improve predictive performance [14]. In the age of deep learning, Lakshminarayanan et al. [29] propose deep ensemble to measure predictive uncertainty. Similar works [39, 40] on uncertainty estimation focus on the context of outlier detection and reinforcement learning. When facing a zoo of PTMs, it's natural to leverage the rich knowledge by assembling multiple PTMs. In prior works, Liu et al. [34] propose using PTMs as teacher models that distill knowledge to a target model for downstream tasks. Shu et al. [46] propose Zoo-Tuning that learns to aggregate the parameters of multiple PTMs to a target model. However, these methods require the target model must have the identical architecture as the PTMs, thus sacrificing flexibility.

Our proposed paradigm involves selecting informative features from assembled feature extractors. In the related works of Bayesian variable selection, a prior is introduced over potential predictor subsets and subsequent method estimates posterior to identify promising subset models. Here we mainly focus on Stochastic search variable selection (SSVS) [38]. Meuwissen and Goddard [36] introduce a random effects variant of SSVS for gene mapping. Li and Zhang [32] consider regression modeling in high-dimensional spaces incorporating structural information. Ročková and George [44] propose EMVS for high-dimentional SSVS promising sparse high posterior probability submodels. Note that all aforementioned feature selection methods are only effective under the IID assumption, while in our paradigm, invariant and informative features can be selected from aggregated PTMs, which improves predictive performance for OoD tasks.

3 ZooD for OoD Generalization

3.1 Model Transferability Ranking

Assume that we have a domain distribution $\mathcal{D}$ from which we observe m domains: $\left\{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_m\right\}$. Each domain $\mathcal{D}_i$ is a set of (label, data) pairs, i.e. $\mathcal{D}_i = \left\{(y_{ij}, x_{ij}), 1 \leq j \leq n_i\right\}$. Meanwhile, we have a zoo of pre-trained feature extractors: $\mathcal{M} = \{\phi_1, \phi_2, \cdots, \phi_k, \cdots\}$. Our objective is to train a predictor f, along with one of the selected feature extractors from $\mathcal{M}$ (e.g., ϕ_k), such that the composed model $f \circ \phi_k$ performs well on both the m observed domains and unseen domains from $\mathcal{D}$.

In this work, we propose an algorithm that facilitates model selection *without* carrying out the fine-tuning step. For every model in $\mathcal{M}$, the algorithm produces an associated score, by which we can *rank* the models, such that the higher-ranked ones have a better chance to deliver stronger results after fine-tuning.

The proposed algorithm is a combination of 1) a model transferability metric and 2) a leave-one-domain-out cross-validation scheme. More specifically, we evaluate each feature extractor m times, and each time we treat the data from the held-out domain as validation data $\{(y'_j, x'_j)\}_{j=1}^{n'}$, while aggregating all remaining $(m-1)$ domains' data as the training data $\{(y_i, x_i)\}_{i=1}^{n}$. In the end, we average the m values of the model transferability metric. Finally, we rank all feature extractors in descending order of the average.

The transferability of each ϕ can be quantified in terms of *inter-class discriminability* and *inter-domain stability*. First, we denote the aggregated domain's label and feature as $\mathbf{y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$ and $\Phi = \big(\phi(x_1), \ldots, \phi(x_n)\big)^\top \in \mathbb{R}^{n\times d}$, respectively. We use $\mathbf{y}' \in \mathbb{R}^{n'}$ and $\Phi' \in \mathbb{R}^{n'\times d}$ for the held-out domain. Inter-domain stability is referring to correlation shift and covariate shift. Therefore, we formulate the objective as the following density function:

$$p(\mathbf{y}', \Phi' \big| \mathbf{y}, \Phi) = p(\mathbf{y}' | \Phi', \mathbf{y}, \Phi) p(\Phi' | \Phi),$$

where $p(\mathbf{y}' | \Phi', \mathbf{y}, \Phi)$ measures discriminability and correlation shift between features Φ' and labels $\mathbf{y}'$, given the aggregated training data. Meanwhile, $p(\Phi'|\Phi)$ measures covariate shift between features Φ and Φ'. Given a hypothetical space $\mathcal{F}$ of classifiers, we can write $p(\mathbf{y}|\Phi) = \int_{f\in\mathcal{F}} p(\mathbf{y}|\Phi, f) p(f) \mathrm{d}f$. According to the Laplace approximation [35], if $p(\mathbf{y}|\Phi, f)$ is unimodal at $\boldsymbol{\mu}$, we can take Taylor expansion of the log-likelihood at the mode $\log p(\mathbf{y}|\Phi, f) \approx \log p(\boldsymbol{\mu}|\Phi, f) - \frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})^\top \Lambda (\mathbf{y}-\boldsymbol{\mu})$, where $\Lambda = -\nabla_{\mathbf{y}^\top} \nabla_{\mathbf{y}} \log p(\mathbf{y}|\Phi, f)\big|_{\mathbf{y}=\boldsymbol{\mu}}$. The quadratic term implies that $p(\mathbf{y}|\Phi, f)$ can be approximated with a Gaussian distribution. Similar to You et al. [54], we consider a linear classifier, i.e. $f \circ \phi(\mathbf{x}) = \mathbf{w}^\top \phi(\mathbf{x})$ with a Gaussian prior of $\mathbf{w}$:

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1}\mathbb{I}_d), \quad \mathbf{y} \big| \Phi, \mathbf{w} \sim \mathcal{N}(\Phi\mathbf{w}, \beta^{-1}\mathbb{I}_n),$$

where α and β are two positive parameters. We estimate $\hat{\alpha}$ and $\hat{\beta}$ by maximizing the model evidence

$$p(\mathbf{y}|\Phi; \alpha, \beta) = \int_{\mathbf{w}\in\mathbb{R}^d} p(\mathbf{y}|\Phi, \mathbf{w}; \beta) p(\mathbf{w}; \alpha) \mathrm{d}\mathbf{w}$$

according to Algorithm 3 in You et al. [54] and compute the likelihood of $\mathbf{y}'$ as follows:

$$p(\mathbf{y}'|\Phi', \mathbf{y}, \Phi; \hat{\alpha}, \hat{\beta}) = \frac{p(\mathbf{y}', \mathbf{y}|\Phi', \Phi; \hat{\alpha}, \hat{\beta})}{p(\mathbf{y}|\Phi; \hat{\alpha}, \hat{\beta})}.$$

For measuring covariate shift, we approximate the distribution of $\phi(x)$ with a Gaussian distribution $\mathcal{N}(\hat{\mu}_\phi, \hat{\Sigma}_\phi)$, where $\hat{\mu}_\phi$ and $\hat{\Sigma}_\phi$ are estimated from the training data Φ. Then we compute the density $p(\Phi'|\Phi) = p(\Phi'|\hat{\mu}_\phi, \hat{\Sigma}_\phi)$ to quantify the covariate shift.

Finally, we compute the density at the logarithmic scale and this defines the proposed metric

$$\text{Metric} = \log p(\mathbf{y}'|\Phi', \mathbf{y}, \Phi) + \log p(\Phi'|\Phi). \tag{1}$$

Please refer to Appendix B.3 and B.4 for more details.

One distinctive aspect of our selection process is the cross-domain validation, embodied in the first term of (1). Across different domains, there are domain-invariant and domain-specific features, where overfitting to the latter can severely harm the OoD generalization. By evaluating on held-out domains, we are able to filter out models that fixate on domain-specific features. To provide theoretical justification, an explicit analysis in the linear regression setting is conducted, where we show that the model with the optimal Metric is the one that select all domain-invariant features. Despite the over-simplification, it does reflect the essence of our approach. Due to page limit, the technical details are presented in Appendix B.5.

3.2 Model Ensemble with Feature Selection

The top-ranked PTMs in Section 3.1 are preferred for solving the OoD generalization task. To further aggregate different PTMs, we consider assembling the top-ranked feature extractors and rewrite $\Phi = \big[\Phi^{(1)}, \ldots, \Phi^{(k)}\big]$, where $\Phi^{(i)}$ is the feature matrix from the i-th ranked feature extractor.

As we show in experiments, in most cases, aggregating features from multiple models can significantly outperform any single model. However, simply concatenating features inevitably introduces more noise. As found in [52], non-informative but invariant features from training domains may only bring some noise, that is irrelevant to the classification problem, and the accumulation of noise hurts the learnability of the OoD generalization task while increasing the memory and computation cost. Therefore, we modify previous top linear model and present a feature selection tool under the Bayesian linear model framework in Section 3.1.

First, we impose a binary mask $\mathbf{z} = (\mathrm{z}_1, \mathrm{z}_2, \ldots, \mathrm{z}_d)^\top$ for the weight vector $\mathbf{w} = (\mathrm{w}_1, \mathrm{w}_2, \ldots, \mathrm{w}_d)^\top$, where $\mathrm{z}_i = 1$ indicates that w_i is an active weight in the top linear model, i.e. $\mathrm{w}_i \neq 0$, meaning the corresponding feature is informative, while $\mathrm{w}_i \approx 0$ if $\mathrm{z}_i = 0$, indicating a noisy feature that should be screened. Therefore the Bayesian feature selection is formulated by estimating the probability π_i of z_i with $\pi_i := p(\mathrm{z}_i = 1)$ and $\boldsymbol{\pi} = \{\pi_1, \pi_2, \ldots, \pi_d\}$.

To facilitate the utility of the mask, we assume that the weights $\{\mathrm{w}_i\}$ are independent of each other and each weight w_i is drawn from either a slab prior or a spike prior [24] with the mean of zero:

$$p(\mathrm{w}_i | \mathrm{z}_i, \alpha_{i,1}, \alpha_{i,2}) = \begin{cases} \mathcal{N}(0, \alpha_{i,1}^{-1}) & \text{if } \mathrm{z}_i = 1; \\ \mathcal{N}(0, \alpha_{i,2}^{-1}) & \text{if } \mathrm{z}_i = 0. \end{cases}$$

We make the Bayesian treatment to linear model in Section 3.1 by introducing gamma priors for all inverse variance terms:

$$\alpha_{i,1} \sim \mathrm{Gamma}(\nu_{i,1}, \nu_{i,2}), \quad \alpha_{i,2} \sim \mathrm{Gamma}(\nu_{i,3}, \nu_{i,4}), \quad \beta \sim \mathrm{Gamma}(\nu_{0,1}, \nu_{0,2}),$$

and denote all hyper-parameters as $\boldsymbol{\nu} = \{\nu_{i,j}\}$. In addition, we denote all latent variables as $\boldsymbol{\xi} = \left\{\beta, \{\mathrm{w}_i, \mathrm{z}_i, \alpha_{i,1}, \alpha_{i,2}\}_{i=1}^d\right\}$. Under certain conditions, maximizing marginal likelihood provably leads to consistent selection and obeys Occam's razor phenomenon [17, 51], and thus screens non-informative features. To estimate π_i, the maximum marginal likelihood estimator of $(\boldsymbol{\pi}, \boldsymbol{\nu})$ is given by

$$\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\nu}} = \underset{\boldsymbol{\pi}, \boldsymbol{\nu}}{\mathrm{argmax}} \log p(\mathbf{y} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu}) = \underset{\boldsymbol{\pi}, \boldsymbol{\nu}}{\mathrm{argmax}} \log \int_{\boldsymbol{\xi}} p(\mathbf{y}, \boldsymbol{\xi} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu}) \mathrm{d}\boldsymbol{\xi}.$$

However, direct maximization of (2) is intractable due to the integration over $\boldsymbol{\xi}$. EM algorithm might be a solution here [44]. In the E-step, we compute the conditional expectation:

$$\mathbb{E}_{\boldsymbol{\xi}} \left[\log p(\mathbf{y}, \boldsymbol{\xi} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu}) \middle| \mathbf{y}, \Phi; \boldsymbol{\pi}^{old}, \boldsymbol{\nu}^{old}\right].$$

Notice that evaluating the expectation involving the posterior distribution of $\boldsymbol{\xi}$. However in our case, it is not straightforward to obtain an analytical form of the true posterior distribution. We instead approximate it using Variational Inference [10] by introducing a tractable distribution Q. Considering the following objective function:

$$\mathcal{L}(Q) = \int_{\boldsymbol{\xi}} Q(\boldsymbol{\xi}; \boldsymbol{\pi}, \boldsymbol{\nu}) \log \frac{p(\mathbf{y}, \boldsymbol{\xi} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu})}{Q(\boldsymbol{\xi}; \boldsymbol{\pi}, \boldsymbol{\nu})} d\boldsymbol{\xi},$$

which is a lower bound of $\log p(\mathbf{y} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu})$. It has been shown the maximizer of $\mathcal{L}(Q)$ is the optimal approximator of $p(\boldsymbol{\xi} | \mathbf{y}, \Phi; \boldsymbol{\pi}, \boldsymbol{\nu})$ under the KL divergence. To obtain an explicit solution, we factorize Q into

$$Q(\boldsymbol{\xi}) = Q(\beta) \prod_{i=1}^d \Big[Q(\mathrm{z}_i) Q(\mathrm{w}_i) Q(\alpha_{i,1}) Q(\alpha_{i,2})\Big], \tag{2}$$

which holds for the classical mean-field family. After all variational parameters in (2) are updated by running one-step coordinate gradient descent [10], in the M-step, we update $\boldsymbol{\pi}^{new}$ and $\boldsymbol{\nu}^{new}$ by maximizing:

$$\mathbb{E}_{\boldsymbol{\xi} \sim Q(\boldsymbol{\xi}; \boldsymbol{\pi}^{old}, \boldsymbol{\nu}^{old})} \left[\log p(\mathbf{y}, \boldsymbol{\xi} | \Phi; \boldsymbol{\pi}, \boldsymbol{\nu})\right].$$

By repeating the E and M step, the estimator $(\boldsymbol{\pi}^{new}, \boldsymbol{\nu}^{new})$ converges to an optimal solution. We then screen those variables with converged prior π_i smaller than the predefined threshold τ. Our derivations for variational approximations and prior hyper-parameters optimization are listed in Appendix C.3.

However, the proposed algorithm still suffers from heavy computational cost: each iteration costs $\mathcal{O}(nd^2)$. To address this problem, we propose an efficient version based on Stochastic Variational Inference [23]. A local estimator $Q^s(\boldsymbol{\xi})$ is established under stochastic approximation that enjoys less computational complexity and guarantees convergence to global optimum [43]. We successfully reduce the computation cost to $\mathcal{O}(n^s d^2)$ with $n^s \ll n$. The complete algorithm is presented in Appendix C.4.

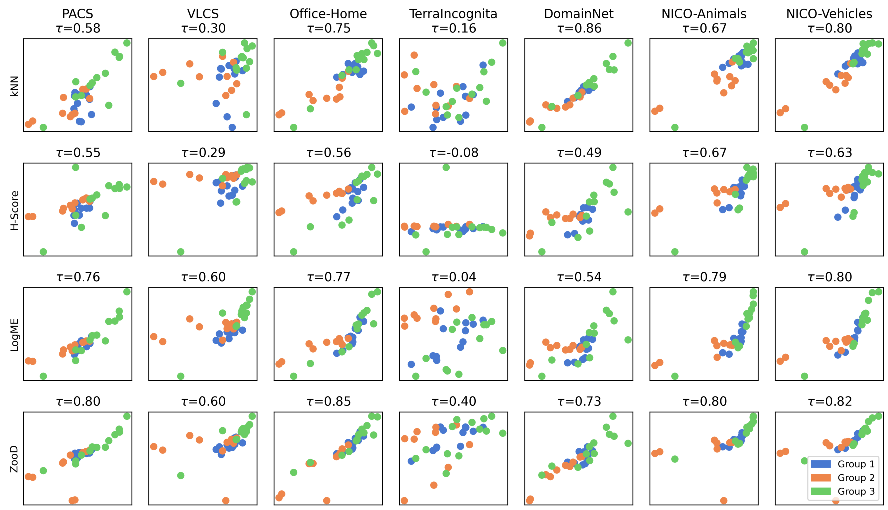

Figure 2: Comparison of ZooD ranking scores with three features-based ranking methods. The plots illustrate ground-truth out-of-domain accuracies (x-axis), ranking scores (y-axis) and Kendall's coefficient τ for 35 PTMs on seven datasets.

4 Experiments

In this section, we demonstrate the effectiveness of ZooD. First, we evaluate the ability of our ranking metric to estimate OoD performance and compare it with ground-truth performance and several existing IID ranking methods. Second, we show that our aggregation methods achieves significant improvements and SOTA results on several OoD datasets. Finally, we demonstrate that ZooD requires significantly less computation, and, therefore, is practically scalable compared with naive fine-tuning.

Setup Details. We use 35 PTMs with diverse architectures, pre-training methods and pre-training datasets. We divide the PTMs into three groups. Group 1 consists of models with different architectures, Group 2 consists of models pre-trained with different training methods, and Group 3 consists of models pre-trained on large-scale datasets. We conduct experiments on six OoD datasets: PACS [30], VLCS [16], Office-Home [48], TerraIncognita [9], DomainNet [41], and NICO (NICO-Animals & NICO-Vehicles) [19]. Each of the datasets has multiple domains. The standard way to conduct experiment is to choose one domain as test (unseen) domain and use the remaining domains as training domains, which is named leave-one-domain-out protocol. The top linear classifier is trained on the training domains only and tested on the test domain. Each domain rotates as the test domain and the average accuracy is reported for each dataset. To get ground-truth performance, we follow DomainBed [18] to fine-tune top linear classifiers for the PTMs on these OoD datasets. We adopt the leave-one-domain-out cross-validation setup in DomainBed with 10 experiments for hyper-parameter selection and run 3 trials. We triple the number of iterations for DomainNet (5000 to 15000) as it is a large-scale dataset requiring more iterations [12] and decrease the number of experiments for hyper-parameter selection from 10 to 5. More details on the experimental setup are in Appendix A.1.

4.1 Comparison with IID Ranking Metrics

IID ranking methods. We divide existing ranking methods into two groups. One group consists of methods that employ PTM's classification layer for ranking. These methods include NCE [47] and LEEP [37]. The other group consists of approaches that only use PTM's extracted features. These methods include H-Score [7] and LogME [55]. Additionally, we also use kNN with k=200 [50] as a baseline.

Evaluation metrics. To evaluate PTMs on OoD datasets with ranking methods, we follow leave-one-domain-out validation protocol [30]. For ZooD and kNN, we further adopt leave-one-domain-out validation for training domains and take average results as the performance prediction for the held-out test domain. To compute the correlation between ranking scores and ground-truth performance, we use two metrics. First, to compare the ranking of a transferability metric with accuracy, we employ Kendall's coefficient τ [25]. Unlike Pearson's correlation, τ measures correlation based on the order of

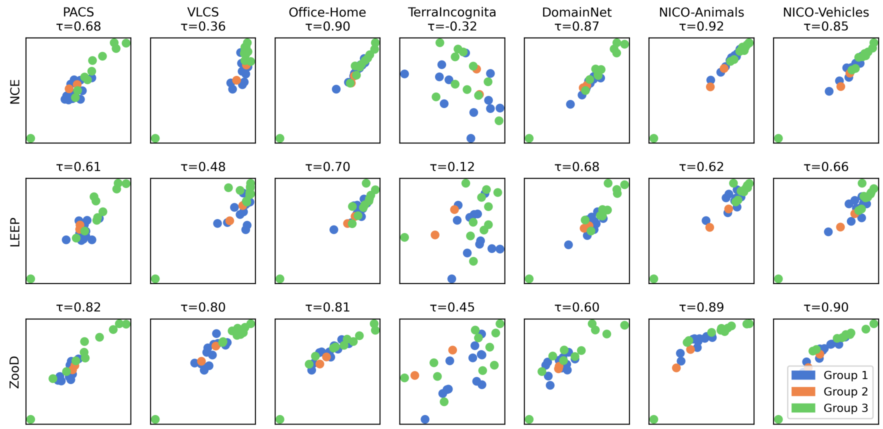

Figure 3: Comparison of ZooD ranking scores with two classification-layer based ranking methods. The plots illustrate ground-truth out-of-domain accuracies (x-axis), ranking scores (y-axis) and Kendall's coefficient τ for 25 PTMs that have classification layers on seven datasets.

Table 1: Comparisons: (a) τ_w between ZooD and feature-based transferability estimation methods using all of our PTMs. (b) τ_w between ZooD and classification-based transferability estimation methods. For this comparison, we consider 25 models that have classification heads. (c) Our method v.s. brute-force fine-tuning in terms of computing cost. For this comparison, we consider all 35 models.

(a) τ_w for feature based

	kNN	H-Score	LogME	ZooD
PACS	0.76	0.57	0.88	**0.91**
VLCS	0.49	0.45	0.79	**0.80**
Office-Home	0.78	0.68	**0.86**	**0.86**
TerraIncognita	0.40	-0.20	0.02	**0.46**
DomainNet	**0.89**	0.62	0.65	0.76
NICO-Animals	0.73	0.72	0.89	**0.90**
NICO-Vehicles	0.82	0.75	0.90	**0.92**

(b) τ_w for Classification based

	LEEP	NCE	ZooD
PACS	0.76	0.81	**0.89**
VLCS	0.57	0.32	**0.88**
Office-Home	0.76	**0.94**	0.86
TerraIncognita	0.02	-0.44	**0.59**
DomainNet	0.77	**0.87**	0.72
NICO-Animals	0.58	0.92	**0.94**
NICO-Vehicles	0.69	0.92	**0.95**

(c) Speed-up over brute-force

GPU Hours	ZooD	Fine-tuning	Speed Up
PACS	0.27	2662.27	9859×
VLCS	0.29	2706.67	9332×
Office-Home	0.39	3089.87	7922×
TerraIncognita	0.49	3920.27	8000×
DomainNet	11.24	17055.33	1516×
NICO-Animals	0.32	2914.40	9107×
NICO-Vehicles	0.30	2794.13	9313×

two measures. Consequently, it is a better criterion for ranking. Second, to measure the performance of transferability metric for top-model selection, we utilize weighted Kendall's coefficient τ_w [49]. The τ_w gives more weight to the ranking of top-performing models compared with the rest of the models. Therefore, it is a better comparative criterion for top model selection.

Results. First, we compare our method with feature-based scoring methods: kNN, H-Score, and LogME. These methods, similar to our method, rank models based on the penultimate layer. We compare ZooD with these methods for the full set of 35 PTMs. We plot ranking scores and ground-truth accuracies in Figure 2. For quantitative comparison, we also provide τ values. It can be seen that ZooD is better correlated with fine-tuning accuracy than other ranking methods on most of the datasets. For example, our method has a τ of 0.85 compared with LogME's τ of 0.77 on Office-Home and a τ of 0.40 compared with LogME's τ of 0.04 on TerraIncognita.

Furthermore, our metric is more stable and consistent. Precisely, τ of ZooD varies between 0.40 $\sim$ 0.85 compared with 0.04 $\sim$ 0.80 for LogME, -0.08 $\sim$ 0.67 for H-Score, and 0.16 $\sim$ 0.86 for kNN. The consistency of transferability metric across different datasets is critical since the purpose of a transferability metric is to estimate performance on a new dataset without having access to ground-truth accuracy. Whenever an estimation metric is inherently unstable, it is hard to determine its reliability for a new dataset.

Note that our method uses a linear model with Gaussian error to approximate the top classifier. This helps us achieve efficient model assessment, especially on small and medium-sized datasets in which the bias caused by model approximation is negligible compared with the estimation error due to insufficient data. However, on DomainNet, things may be different. The bias caused by model approximation dominants the evaluation performance on large datasets. Therefore, our method does not outperform kNN on DomainNet.

Second, we compare our method with classification-layer based methods: NCE and LEEP. For this comparison, we select a subset of our PTMs that have classification layers. The results are illustrated

Table 2: Comparison of out-of-domain accuracies between ZooD and SOTA OoD methods. The results of MixStyle [56] and SWAD [12] are from SWAD, and other results are from Gulrajani and Lopez-Paz [18] (denoted with †). Our results are average of three trials.

Method	PACS	VLCS	Office-Home	TerraInc.	Domain	Avg
ERM†	85.5	77.5	66.5	46.1	40.9	63.3
IRM†	83.5	78.6	64.3	47.6	33.9	61.6
GroupDRO†	84.4	76.7	66.0	43.2	33.3	60.7
I-Mixup†	84.6	77.4	68.1	47.9	39.2	63.4
MLDG†	84.9	77.2	66.8	47.8	41.2	63.6
MMD†	84.7	77.5	66.4	42.2	23.4	58.8
DANN†	83.7	78.6	65.9	46.7	38.3	62.6
CDANN†	82.6	77.5	65.7	45.8	38.3	62.0
MTL†	84.6	77.2	66.4	45.6	40.6	62.9
SagNet†	86.3	77.8	68.1	48.6	40.3	64.2
ARM†	85.1	77.6	64.8	45.5	35.5	61.7
VREx†	84.9	78.3	66.4	46.4	33.6	61.9
RSC†	85.2	77.1	65.5	46.6	38.9	62.7
MixStyle	85.2	77.9	60.4	44.0	34.0	60.3
SWAD	88.1	79.1	70.6	50.0	46.5	66.9
ZooD						
Single	96.0	79.5	84.6	37.3	48.2	69.1
Ensemble	95.5	80.1	85.0	38.2	50.5	69.9
F. Selection	**96.3**	**80.6**	**85.1**	42.3	**50.6**	**71.0**
F. Ratio (%)	24.3	24.5	62.5	76.8	99.8	

in Figure 3. It can be seen that ZooD is also more stable and consistent than NCE and LEEP. Moreover, Our method achieves superior performance on the difficult real-world TerraIncognita dataset. This dataset consists of obscure and blurry images captured by WildCams installed in different territories. NCE has a negative correlation for this dataset. On the other hand, our method, although not perfect, captures the relation in a better way. For this challenging dataset, our method has a τ of 0.45 compared with 0.12 and -0.32 for LEEP and NCE, respectively.

Third, we compare weighted Kendall's coefficient of our method with other ranking methods. The weighted Kendall's coefficient is a better metric to gauge the performance of a metric for top model selection. We also divide these results into two groups: comparison with feature-based scoring methods in Table 1a and comparison with classification-based scoring methods in Table 1b. Our method outperforms feature-based scoring methods on 6 out of 7 datasets. Similarly, it also outperforms both LEEP and NCE on 5 out of 7 datasets. Moreover, our ranking method is more stable as it performs better on challenging datasets. For example, it has τ_w of 0.46 $\sim$ 0.92 compared with LogME's τ_w of 0.02 $\sim$ 0.90 and H-Score's τ_w of -0.20 $\sim$ 0.75.

In summary, transferability estimation of ZooD correlates better with ground-truth accuracy on most of the OoD datasets compared with previous ranking methods. It also outperforms most feature-based metrics for model selection in terms of τ_w. Additionally, it is more stable and consistent across datasets, making it a better choice for pre-trained model selection.

4.2 SOTA Results with Our Selection Method

We also compare ZooD (model ranking and feature selection) with several recent SOTA OoD methods and demonstrate that it achieves substantial performance improvements. We compare previous OoD methods with three versions of our method: 1) **Single**: fine-tune the top-1 model by transferability metric; 2) **Ensemble**: fine-tune an ensemble of the top-K models; 3) **F. Selection**: fine-tune an ensemble of the top-K models with feature selection, which is the expected result using ZooD. By fine-tuning, we mean using ERM with DomainBed settings to fine-tune a top linear classifier for the PTMs. Their predictive performance and **F. Ratio** (the percentage of features used in **F. Selection**) are listed in the last four lines of Table 2.

In all experiment results, except TerraIncognita (discussed in the next paragraph), our method achieves remarkable improvement against ERM and recent SOTA. For **Single**, we list the improvements over the previous SOTA as follows: +14% on Office-Home, +7.9% on PACS, +1.7% on DomainNet, and +0.4% on VLCS. This result also shows that even without aggregation, using proper pre-trained model can improve OoD generalization by a large margin.

The performance of **Single** does not outperform the previous SOTA on TerraIncognita. This is because previous methods fine-tune the whole network. In contrast, we only train a classifier on top of a fixed feature extractor. TerraIncognita is a much more challenging dataset compared with other OoD datasets, as the majority of its images are obscured by the background. Therefore it requires fully fine-tuning. To show the effectiveness of ZooD with fully fine-tuning, we select top-1 ranked model and fine-tune the whole model. Our resulted model achieves a +2.6% improvement compared with the previous SOTA. One limitation of ZooD when aggregating multiple models is that fine-tuning the whole models is difficult due to the limitation of GPU memory. However, for OoD tasks, fine-tuning the whole model may not perform better than fine-tuning the top classifier. For example, the results of fine-tuning the full top-ranked models on PACS, VLCS and Office-Home are 90.6, 79.1 and 83.4, respectively. Empirically, we find if a PTM is suitable for a given OoD task, fine-tuning the top classifier has better OoD generalization than fine-tuning the full model.

To efficiently utilize multiple models, we propose to select informative features in Section 3.2. Here, we compare the performance improvement by **F. Selection** with **Single** and **Ensemble**. ZooD significantly outperforms both candidates while only using a small portion of aggregated features from top-K models. Even on the most sophisticated DomainNet, ZooD can improve predictive performance by +2.4% compared with **Single** and +0.1% compared with **Ensemble**.

To find the appropriate number K for the model ensemble, we performed an ablation study. We varied the number of K, e.g. $K \in \{3, 5, 7\}$. The performance changes are plotted in Figure 4. We found the performance by aggregating top-3 models strikes the right balance between performance and computational complexity. Hence, $K = 3$ is set to the default value.

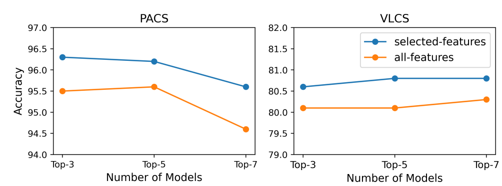

Figure 4: Comparison of selected-feature ensemble vs. all-feature ensemble for varying number of top models in the ensemble.

In summary, our ranking metric in ZooD is good enough to select a model that can outperform the previous SOTA methods without adding any bells and whistles. Furthermore, feature selection in ZooD can efficiently utilize informative features from top-K models to further improve the OoD generalization. Based on extensive experimental results on various OoD datasets, we conclude ZooD makes it easy and efficient to exploit a large set of PTMs for OoD generalization.

4.3 Computational Efficiency of ZooD

In the previous sections, we show its performance on several small and large-scale OoD datasets. Here, we illustrate the precision and computational efficiency of ZooD by comparing it with brute-force fine-tuning in terms of GPU hours. The results are shown in Table 1c. ZooD provides a minimum of 1516× speed-up for DomainNet and a maximum of 9859× speed-up for PACS. Cumulatively, our method took a total of 13 GPU hours to evaluate all the PTMs on all the datasets compared with 35140 GPU hours (equivalent to 4 GPU years) for brute-force fine-tuning. Therefore, ZooD is a scalable and practical method for OoD generalization.

5 Conclusion

Machine learning models rely on IID assumption, which is often violated due to constant distribution shifts in the real-world applications. In this work, we argue for leveraging a large set of PTMs to improve OoD generalization and propose ZooD, a paradigm for efficient PTMs ranking and aggregation. Our paradigm avoids the computationally-prohibitive fine-tuning by ranking PTMs based on quantifying their inter-class discriminability and inter-domain stability, and selecting the most informative features from top-ranked PTMs ensemble. Extensive experiments show ZooD is superior in ranking correlation with the ground-truth performance and achieves SOTA results on various OoD benchmarks.

References

[1] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

[2] Isabela Albuquerque, Nikhil Naik, Junnan Li, Nitish Shirish Keskar, and Richard Socher. Improving out-of-distribution generalization via multi-task self-supervised pretraining. *ArXiv*, abs/2003.13525, 2020.

[3] Anonymous. An empirical study of pre-trained models on out-of-distribution generalization. In *Submitted to The Tenth International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=2RYOwBOFesi`. under review.

[4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[5] Haoyue Bai, Rui Sun, Lanqing Hong, Fengwei Zhou, Nanyang Ye, Han-Jia Ye, S-H Gary Chan, and Zhenguo Li. Decaug: Out-of-distribution generalization via decomposed feature representation and semantic augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6705–6713, 2021.

[6] Haoyue Bai, Fengwei Zhou, Lanqing Hong, Nanyang Ye, S-H Gary Chan, and Zhenguo Li. Nas-ood: Neural architecture search for out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8320–8329, 2021.

[7] Yajie Bao, Yongni Li, Shao-Lun Huang, Lin Zhang, Lizhong Zheng, Amir Roshan Zamir, and Leonidas J. Guibas. An information-theoretic approach to transferability in task transfer learning. *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2309–2313, 2019.

[8] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in Neural Information Processing Systems*, 32:9453–9463, 2019.

[9] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *ECCV*, 2018.

[10] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[11] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019.

[12] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *arXiv preprint arXiv:2102.08604*, 2021.

[13] Joel Dapello, Tiago Marques, Martin Schrimpf, Franziska Geiger, David D Cox, and James J DiCarlo. Simulating a primary visual cortex at the front of cnns improves robustness to image perturbations. *BioRxiv*, 2020.

[14] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[15] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *Advances in Neural Information Processing Systems*, 32: 6450–6461, 2019.

[16] Chen Fang, Ye Xu, and Daniel N. Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. *2013 IEEE International Conference on Computer Vision*, pages 1657–1664, 2013.

[17] Subhashis Ghosal, Jüri Lember, and Aad Van Der Vaart. Nonparametric bayesian model selection and averaging. *Electronic Journal of Statistics*, 2:63–89, 2008.

[18] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

[19] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, page 107383, 2020.

[20] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[21] Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Xiaodong Song. Pretrained transformers improve out-of-distribution robustness. In *ACL*, 2020.

[22] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021.

[23] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14(5), 2013.

[24] Hemant Ishwaran and J Sunil Rao. Spike and slab variable selection: frequentist and bayesian strategies. *The Annals of Statistics*, 33(2):730–773, 2005.

[25] Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[26] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pages 5637–5664. PMLR, 2021.

[27] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *International Conference on Machine Learning*, pages 5815–5826. PMLR, 2021.

[28] Kun Kuang, Peng Cui, Susan Athey, Ruoxuan Xiong, and Bo Li. Stable prediction across unknown environments. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1617–1626, 2018.

[29] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems*, 30, 2017.

[30] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5543–5551, 2017.

[31] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[32] Fan Li and Nancy R Zhang. Bayesian variable selection in structured high-dimensional covariate spaces with applications in genomics. *Journal of the American statistical association*, 105(491):1202–1214, 2010.

[33] Yichen Li and Xingchao Peng. Network architecture search for domain adaptation. *arXiv preprint arXiv:2008.05706*, 2020.

[34] Iou-Jen Liu, Jian Peng, and Alexander G Schwing. Knowledge flow: Improve upon your teachers. *arXiv preprint arXiv:1904.05878*, 2019.

[35] David J. C. MacKay. Choice of basis for laplace approximation. *Mach. Learn.*, 33(1):77–86, 1998.

[36] Theo HE Meuwissen and Mike E Goddard. Mapping multiple qtl using linkage disequilibrium and linkage analysis information and multitrait data. *Genetics Selection Evolution*, 36(3):261–279, 2004.

[37] Cuong V Nguyen, Tal Hassner, C. Archambeau, and Matthias W. Seeger. Leep: A new measure to evaluate transferability of learned representations. In *ICML*, 2020.

[38] Robert B O'Hara and Mikko J Sillanpää. A review of bayesian variable selection methods: what, how and which. *Bayesian analysis*, 4(1):85–117, 2009.

[39] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29:4026–4034, 2016.

[40] Nick Pawlowski, Miguel Jaques, and Ben Glocker. Efficient variational bayesian neural network ensembles for outlier detection. *arXiv preprint arXiv:1703.06749*, 2017.

[41] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1406–1415, 2019.

[42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[43] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[44] Veronika Ročková and Edward I George. Emvs: The em approach to bayesian variable selection. *Journal of the American Statistical Association*, 109(506):828–846, 2014.

[45] Zheyan Shen, Peng Cui, Tong Zhang, and Kun Kunag. Stable learning via sample reweighting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5692–5699, 2020.

[46] Yang Shu, Zhi Kou, Zhangjie Cao, Jianmin Wang, and Mingsheng Long. Zoo-tuning: Adaptive transfer from a zoo of models. In *International Conference on Machine Learning*, pages 9626–9637. PMLR, 2021.

[47] A. Tran, Cuong V Nguyen, and Tal Hassner. Transferability and hardness of supervised classification tasks. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1395–1405, 2019.

[48] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5385–5394, 2017.

[49] Sebastiano Vigna. A weighted correlation index for rankings with ties. In *Proceedings of the 24th international conference on World Wide Web*, pages 1166–1176, 2015.

[50] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[51] Yun Yang and Debdeep Pati. Bayesian model selection consistency and oracle inequality with intractable marginal likelihood. *arXiv preprint arXiv:1701.00311*, 2017.

[52] Haotian Ye, Chuanlong Xie, Tianle Cai, Ruichen Li, Zhenguo Li, and Liwei Wang. Towards a theoretical framework of out-of-distribution generalization. *arXiv preprint arXiv:2106.04496*, 2021.

[53] Mingyang Yi, Lu Hou, Jiacheng Sun, Lifeng Shang, Xin Jiang, Qun Liu, and Zhi-Ming Ma. Improved ood generalization via adversarial training and pre-training. In *ICML*, 2021.

[54] Kaichao You, Yong Liu, Jianmin Wang, Michael I Jordan, and Mingsheng Long. Ranking and tuning pre-trained models: A new paradigm of exploiting model hubs. *arXiv preprint arXiv:2110.10545*, 2021.

[55] Kaichao You, Yong Liu, Jianmin Wang, and Mingsheng Long. Logme: Practical assessment of pre-trained models for transfer learning. In *International Conference on Machine Learning*, pages 12133–12143. PMLR, 2021.

[56] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. *ArXiv*, abs/2104.02008, 2021.

Checklist

1. For all authors...
    (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
    (b) Did you describe the limitations of your work? [Yes] See Section 4.2
    (c) Did you discuss any potential negative societal impacts of your work? [N/A]
    (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
    (a) Did you state the full set of assumptions of all theoretical results? [Yes]
    (b) Did you include complete proofs of all theoretical results? [Yes] Mainly in the Appendix.
3. If you ran experiments...
    (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Will be released upon publication.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Setup Details in Section 4 and Appendix A.1.
(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]