Analyzing Deep Neural Network's Transferability via Fréchet Distance

Yifan Ding Liqiang Wang
Department of Computer Science, University of Central Florida
yf.ding@knights.ucf.edu, lwang@cs.ucf.edu

Boqing Gong
Google
bgong@google.com

Abstract

Transfer learning has become the de facto practice to reuse a deep neural network (DNN) that is pre-trained with abundant training data in a source task to improve the model training on target tasks with smaller-scale training data. In this paper, we first investigate the correlation between the DNN's pre-training performance in the source task and their transfer results in the downstream tasks. We find that high performance of a pre-trained model does not necessarily imply high transferability. We then propose a metric, named Fréchet Pre-train Distance, to estimate the transferability of a deep neural network. By applying the proposed Fréchet Pre-train Distance, we are able to identify the optimal pre-trained checkpoint, and then achieve high transferability on downstream tasks. Finally, we investigate several factors impacting DNN's transferability including normalization, different networks and learning rates. The results consistently support our conclusions.

1. Introduction

This paper is concerned with the transferability of deep neural networks (DNNs), which are pre-trained in a source task with abundant training data, to downstream tasks whose training sets are small-scale or medium-sized. The transferability of DNNs has been studied from various perspectives. Since DNNs have hierarchical architectures, the layers represent different feature granularities and result in distinct transferabilities [28, 15, 4]. The pre-training methods also play a key role in DNNs' transferabilities [25, 9].

All the studies mentioned above fine-tune the DNNs that have converged on the source task. However, we find that the converged models do not always lead to better transfer results to the downstream tasks than those stopped early. Moreover, if we pick up a checkpoint from the early pre-training stage, we could possibly get a transfer result worse than train-from-scratch. *Transfer results* is referred to the highest-performing model after fine-tuning over the target task.

Hence, a new question is raised about DNN's transferability. Suppose the pre-training method and the source task do not change, and that we save multiple checkpoints during the course of the pre-training (e.g., after each learning rate decay). How can we identify the best checkpoint for a given target task? Here, the best checkpoint is referred to the one that yields the best transfer results.

To address the above question, we propose to a specific metric to measure transferability, named Fréchet Pre-train Distance (FPD), derived from Fréchet Distance [6], a widely-used metric to measure the distance between two distributions. We compute Fréchet Pre-train Distance between the source and target datasets through all the pre-trained checkpoints. Our extensive experiments demonstrate that Fréchet Pre-train Distance is well correlated with the checkpoints' transferability for target tasks under different experimental settings.

Equipped with the Fréchet Pre-train Distance, we extend our study to investigate multiple impact factors in transfer learning, including fine-tuneing learning rates, DNNs' depths, and Spectral Normalization [16] to DNNs' weights. An interesting finding is that the over-parameterized fully-connected layer hurts the transferability of AlexNet [14], and yet the Spectral Normalization [16] can alleviate it.

To conclude, our work makes three major contributions:

- We investigate how the transfer performance varies along with the pre-training process.We find that pre-training would not necessarily improve transfer performance, but, on the contrary, sometimes the transfer performance decreases when the pre-training performance increase.
- We propose to use Fréchet Pre-train Distance to estimate the transferability of a pre-trained network between source and target datasets. Our extensive experiments show that Fréchet Pre-train Distance is well correlated with the checkpoints' transferability to target tasks. With the proposed Fréchet Pre-train Distance, we are able to pick up an optimal pre-trained checkpoint for given target tasks without actually conducting transfer learning experiments.
- We further investigate multiple impact factors on the

transfer performance of neural networks. We find that over-parameterization hurts deep neural networks' transferability in the early training stage and the Spectral Normalization helps recover it. Our experiments with different learning rates and networks also support our previous claims that networks' transferability and Fréchet Pre-train Distance are consistent correlated across different settings.

The rest of this paper is organized as follows. Section 2 discusses related areas to our method. In Section 3, we discuss the pre-train/transfer performance correlation and propose Fréchet Pre-train Distance to quantify the transferability of pre-trained networks. Section 4 shows our extensive experimental results. Finally, we conclude our paper in Section 5.

2. Related Work

Transfer learning [17, 5] is a widely used technique in visual perception algorithms, where a deep neural network is first trained on the source dataset then fine-tuned on another downstream dataset. Through transfer learning, knowledge learned from the source task is transferred to the target task. However, a questions is raised here: how to efficiently transfer the knowledge learned from the source dataset and avoid the negative impact from the discrepancy between source and target domains? Previous studies propose different approaches to answer the question.

A straightforward idea is to visualize and understand the knowledge learned by neural networks. Along with this direction, a few approaches were proposed to interpret neural networks through visualization techniques. Yosinski *et al*. [29] propose two tools to visualize live activations and features. Simonyan *et al*. propose an approach to visualize the notion of classes and saliency maps [19]. More recent studies include interpreting through explanatory graph [31] and decision trees [32]. Moreover, to understand neural networks more precisely, quantitative methods are also proposed for interpretation purposes. Bau *et al*. align hidden units with human-interpretable concepts to interpret deep visual representations and quantify their interpretability [4]. Achille *et al*. introduce the notion of "Information in the Weights" to measure generalizability of DNNs [2]. Yosinski *et al*. study the layer-wise transferability in transfer learning tasks by freezing a different number of pre-trained layers and observing the change of transfer performance [28], which experimentally quantifies the generality versus specificity in deep neural networks.

Besides knowledge or information measurement, learning process is also investigated. Achille *et al*. [1] measure Fisher Information of weights in each training phase and conclude that there are "memorization phase" and "forgetting phase" in the learning process. Kirkpatrick *et al*. investigate how to avoid information forgetting in transfer learning [13]. Moreover, the "break-even" point is proposed on the optimization trajectory of learning, and the curvature of the loss surface and noise in the gradient are implicitly regularized by SGD [12]. A similar work studying dynamic stability of learning process is proposed by [24].

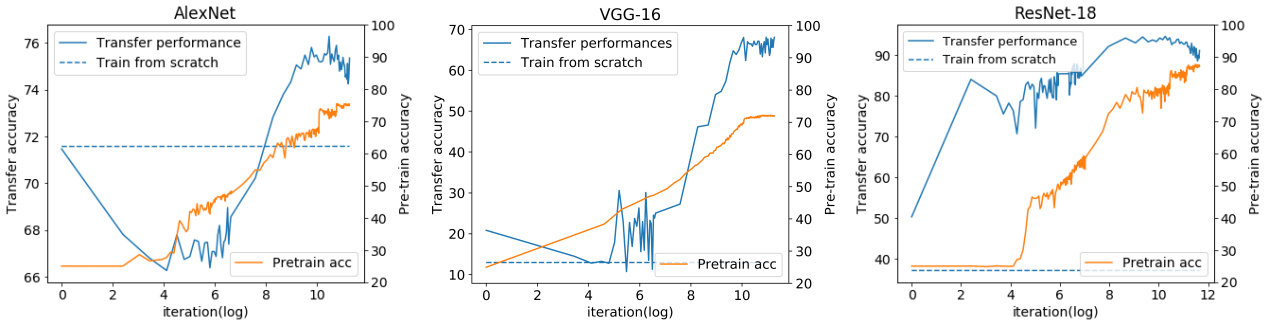
Initialization affects the transferability in many ways. Ash *et al*. [3] compares the performance between warm-starting and fresh random initialization. Regularization may also lead to a better initialization and sometimes help improve the transferability [27, 26]. Miyato *et al*. use spectral norm to evaluate the generalizability in Generative Adversarial Networks [8] and propose the Spectral Normalization to improve the performance of neural networks [16]. Li *et al*. propose L_2-SP penalty with the pre-trained model being referred as the baseline of penalty for transfer learning tasks [26]. As a domain adaptation approach, a parameter regularization scheme is introduced to encourage the representation similarity between the source and target domains [18].

3. Transferability of Neural Networks

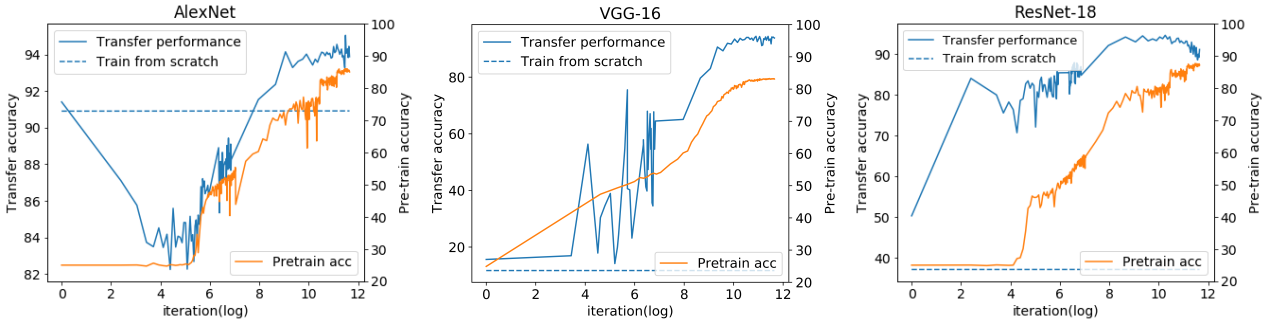
3.1. Pre-training performance VS transfer performance

Transfer learning is a research problem that focuses on storing knowledge gained in solving the source problem and applying it to a different but related target downstream problem [23]. The ultimate goal is to improve the performance on the target problem. To achieve that, most previous work starts from the checkpoint which gains the best performance on source task [9, 25, 30]. While, a problem remaining less explored: does a better model on the source problem necessarily imply a better initialization for the target problem? We conduct an experiment in which we first learn a base network on the source task [7] and keep all checkpoints during the learning. Then we initialize the target network with these pre-trained checkpoints and conduct the same transfer experiments one by one. Finally, best test performance achieved during the fine-tuning is recorded for each pre-trained checkpoint.

We conduct transfer learning experiments on AlexNet [14], VGG-16 [20] and ResNet-18 [10]. We use CIFAR-100 and SVHN as the source datasets and CIFAR-10 and MNIST as the target datasets, respectively. Since both CIFAR-10 and MNIST are simple datasets, to obtain recognizable differences, we choose 10% of CIFAR10 and 1% of MNIST training data for the target tasks' training. But the testing data in CIFAR10 and MNIST remain the same. We follow the experiment settings in [7] to conduct both pre-training and transfer learning. For both source and target tasks, SGD is used for training with batch size 128, momentum 0.9 and weight decay $5e-4$.



(a) CIFAR100-CIFAR10



(b) SVHN-MNIST

Figure 1: Transfer performance vs pre-training performance.

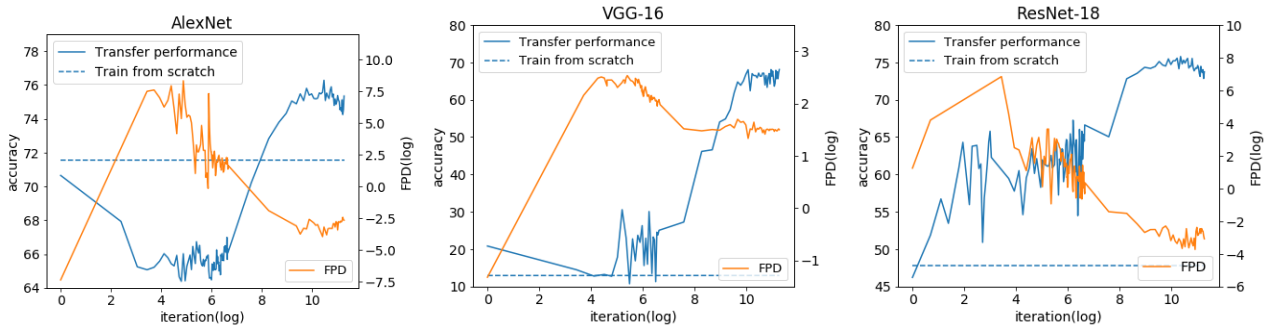
For the source task, we use an initial learning rate of 0.1 for AlexNet and ResNet-18, 0.01 for VGG-16 and the learning rate is dropped by a factor of 5 after epochs 60, 120, and 160 for a total of 200 epochs. In the target task, 100 epochs are trained with the learning rate dropped after 30,60 and 80 epochs by the same factor of 5. The initial learning rate of the pre-trained layers is divided by a factor of 2 compared with the pre-training. Since the results change rapidly in the early stage of the pre-training and become more consistent in the finishing stage, we test the pre-trained checkpoints every 30 iterations in the first 2 epochs, and then every 5 epochs for rest epochs. We also use a log scale for the x-axis in Figure 1 to better visualize the results. All transfer performance refer to the best test performance achieved in the 100 training epochs.

Not surprisingly, Figure 1 shows that better pre-training accuracy (orange lines) does not necessarily lead to better transfer performance (blue lines). On the contrary, in most cases, while the pre-training accuracy is still increasing in the later epochs, the transfer performance starts decreasing. The transfer performance could even drop as many as 5 points on AlexNet using CIFAR100-CIFAR10 setting. Besides, the transfer performance variation is more dramatic in the earlier pre-training stages compared with the later stages, which is also reasonable since a large learning rate is used at the beginning of the pre-training.

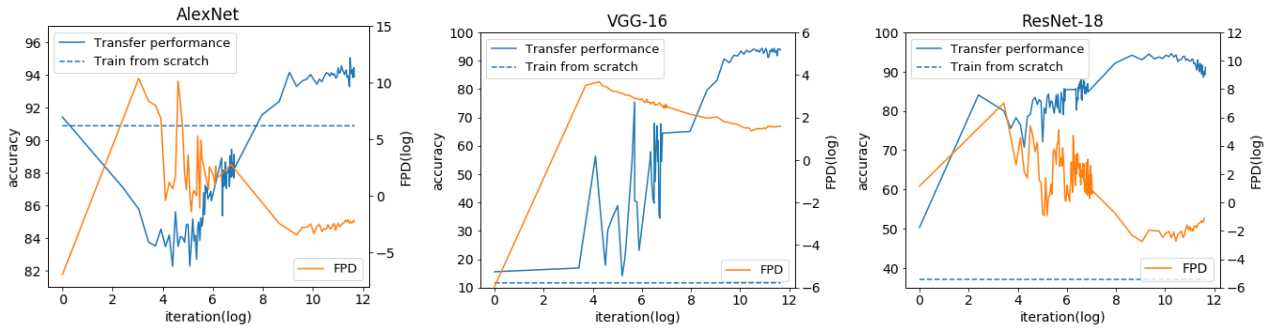
We also notice that for AlexNet, some pre-trained checkpoints from the early training stage would result in a worse transfer learning performance compared with train-from-scratch (blue dotted line), which indicates that picking a wrong pre-trained checkpoint is likely to deviate us from the best performance on the target task, or even leads us to a wrong direction. Compared with AlexNet, VGG and ResNet perform more consistently, as almost all pre-trained checkpoints lead to a better accuracy compared with training from scratch. The only difference is that the transferability decreasing is less severe for VGG in the late stage compared with ResNet.

3.2. Measuring Transferability with Fréchet Pre-train Distance

Fréchet distance Fréchet distance [6] is a measure of similarity between distributions. Specifically, Fréchet distance d between a Gaussian distribution with mean and co-variance (m_1, C_1) and another Gaussian distribution with mean and co-variance (m_2, C_2) is known as Wasserstein-2 distance [22], which is defined as:



(a) CIFAR100-CIFAR10



(b) SVHN-MNIST

Figure 2: Transfer Performance vs Fréchet Pre-train Distance.

$$d^2\Big((m_1,C_1),(m_2,C_2)\Big) = ||m_1 - m_2||_2^2 + Tr(C_1 + C_2 - 2\Big(C_1C_2\Big)^{\frac{1}{2}}\Big) \tag{1}$$

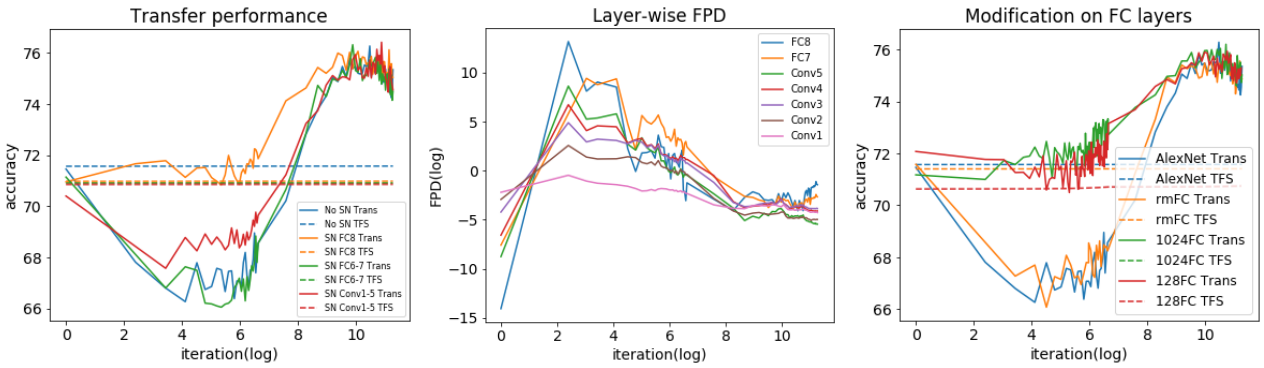
Where Tr denotes the Trace of matrix.

Fréchet Inception Distance (FID) Since Fréchet Distance could be used to measure the similarity of two Gaussian distributions. Heusel *et al*. propose to measure the similarity between GAN [8] generated images and real ones with Fréchet Inception Distance (FID) [11]. The FID is measured between the real images, the GAN generated images, and the ImageNet [14] pre-trained Inception network [21]. In which two Gaussians are fitted on the Inception outputs while the real images and GAN generated images are the inputs, respectively. The value of FID is used to identify if the GAN generated images are as real and diverse as real ones.

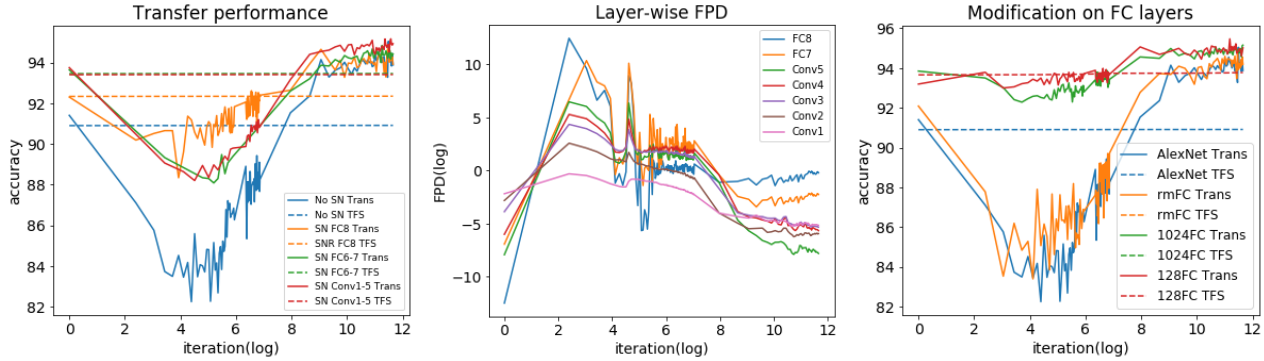
Fréchet Pre-train Distance (FPD) Inspired by Fréchet Inception Distance, we find Fréchet distance can also be used to measure the distances between the source dataset and the target dataset on a given pre-trained network in transfer learning. More specifically, in FID, when the Inception network is fixed, the variation of the Fréchet distance represents the similarity of two group of images. On the contrary, when we fix the two groups of images but change the pre-trained networks, would Fréchet distance also be able to represent the transferability of the pre-trained neural network which connects the source and target task? It provides a potential solution to our previously raised question *i.e*., how to select a best pre-trained checkpoint based on the target performance. Usually the pre-trained network with the best performance on the source dataset and task would be used. But we propose to measure the transferability of a given pre-trained network with Fréchet distance, which proves a quantitative metric to help select a checkpoint which might don't have the best performance on the source task but the best performance on the target task. The Fréchet Pre-train Distance is defined as follows.

$$\begin{aligned} FPD &= \sqrt{d^2\Big((m_s,C_s),(m_t,C_t)\Big)} \\ m_s &= \mathbb{E}(f_\theta(X_s)), m_t = \mathbb{E}(f_\theta(X_t)) \\ C_s &= \mathrm{Cov}(f_\theta(X_s)), C_t = \mathrm{Cov}(f_\theta(X_t)) \end{aligned} \tag{2}$$

where f_θ denotes the source network, X_s and X_t denote the source and target datasets, respectively. m_s and m_t are the sample means and C_s and C_t are the sample covariances for



(a) CIFAR100-CIFAR10



(b) SVHN-MNIST

Figure 3: Transfer Performance Recovered by Spectral Normalization (SN: Spectral Normalization, Trans: transfer performance, TFS: train-from-scratch, rmFC: remove 2 inner FC layers, 1024/128FC: change the size of 2 inner FC layers).

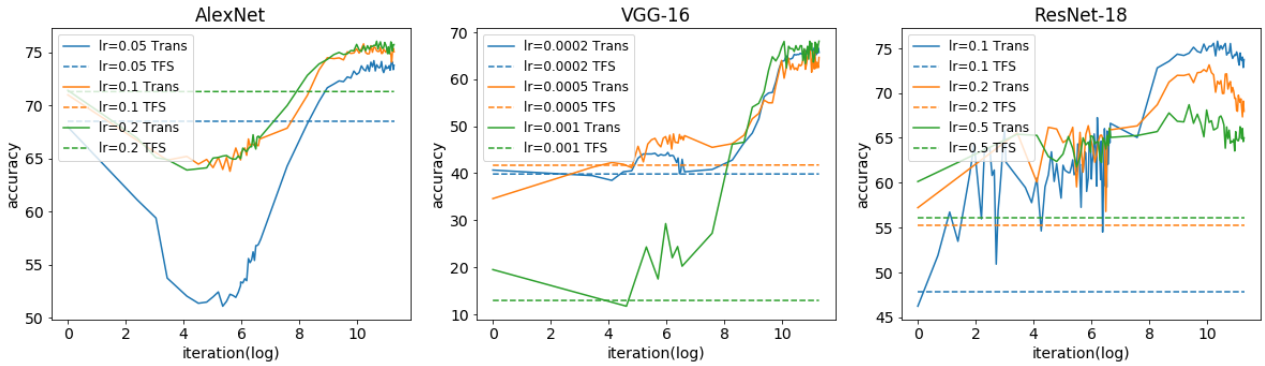
Algorithm 1 Find the optimal pre-training checkpoint with Fréchet Pre-train Distance

Input: Source network f_s, source training dataset X_s, target network f_t, target training dataset X_t
Output: Optimal transfer performance on target task $\hat{A}^t$

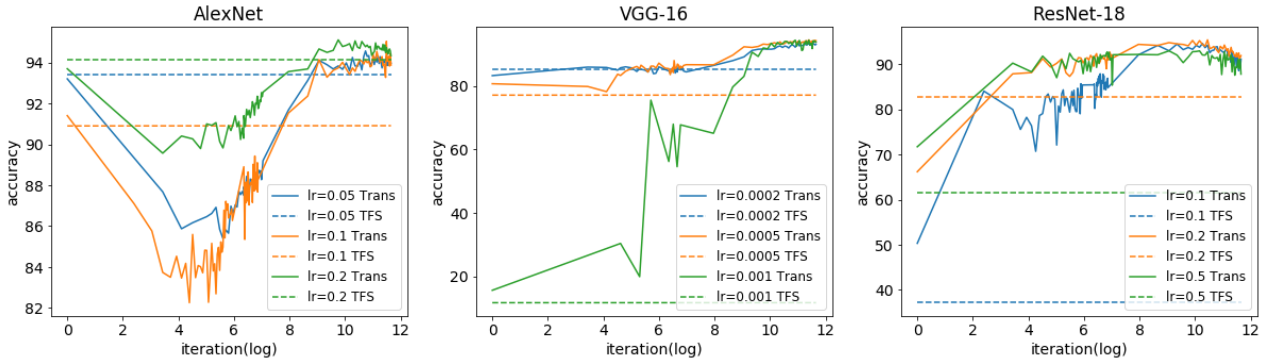
Initialize the source network f_s^0=f_s, initialize the minimal FPD $\hat{F}$
for every epoch **do**
  $f_s^i \Leftarrow$ Training f_s^{i-1} with X_s
  Evaluate Fréchet Pre-train Distance of the current ith pre-training epoch

$$F_i = \sqrt{||m_s^i - m_t^i||_2^2 + Tr(C_s^i + C_t^i - 2(C_s^i C_t^i)^{\frac{1}{2}})}$$
$$m_s^i = \mathbb{E}(f_s^i(X_s)), m_t^i = \mathbb{E}(f_s^i(X_t))$$
$$C_s^i = \mathrm{Cov}(f_s^i(X_s)), C_t^i = \mathrm{Cov}(f_s^i(X_t))$$

  if $F_i < \hat{F}$ **then**
    Record the best epoch $i_{best} = i$
    Record the best FPD $\hat{F} = F_i$
  end if
end for
Initialize the target network f_t with the weights of source network $f_s^{i_{best}}$ from the i_{best} epoch.
Fine-tunning the target network f_t with X_t and return the best transfer accuracy on target task $\hat{A}^t$.



(a) CIFAR100-CIFAR10



(b) SVHN-MNIST

Figure 4: Transfer Performance vs Different Learning Rate (Trans: transfer performance, TFS: train-from-scratch).

the source domain and target domain. In our experiments, we evaluate Fréchet Pre-train Distance between the source and target test datasets on each pre-trained checkpoint. Figure 2 shows the correlation between the value of FPD and the best test performance achieved on target task.

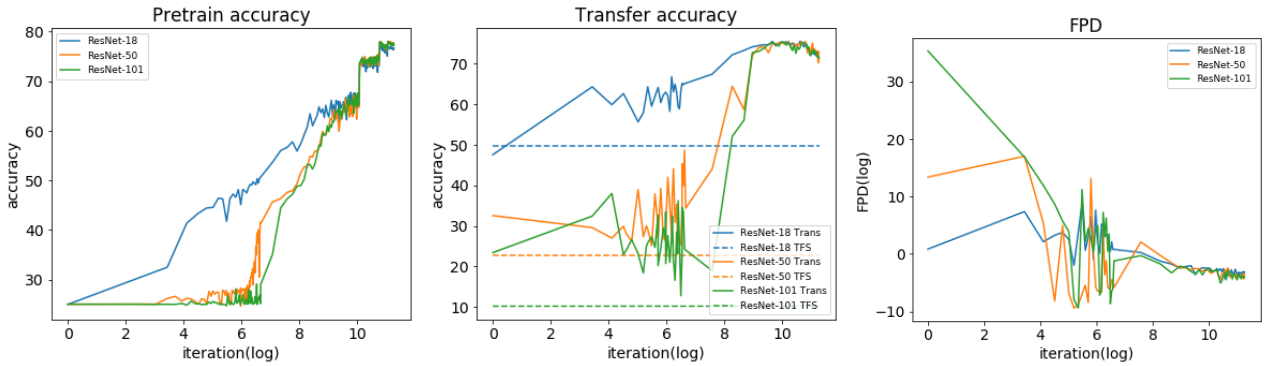
In Figure 2, the left y labels show the best performance achieved in the target task (among 100 epochs) while the bottom layers of the network are initialized with the weights learned on the $i-th$ iteration of the pretraining. The right y labels show the value of FPD. Since the transfer performance changes more dramatically in the early pre-training stages than in the finishing stages. We plot the iterations as the x-axis in a log scale.

Surprisingly, the value of Fréchet Pre-train Distance evidently negatively correlates with the variation of the transfer performance, both in the early training stage and late training stage. It shows a similar trend for all three representative networks (Alexnet, VGG-16, and Resnet-18) and two dataset settings (CIFAR100-CIFAR10, SVHN-MNIST). At the turning point when Fréchet Pre-train Distance begins to decrease, the transfer performance begins to increase. At the late stages, the transfer performance decreases on both AlexNet and ResNet18 while the Fréchet Pre-train Distance also increase slightly. On VGG, both the transfer performance and Fréchet Pre-train Distance are more even in the late stages. The experiments verify that Fréchet Pre-train Distance measures the transferability of the tested networks well, which are nowadays very popular in most of machine learning tasks. Specifically, for the most widely used ResNet, we also test how Fréchet Pre-train Distance works on ResNet-50 and ResNet-101. The comparison is shown in Section 4.3.

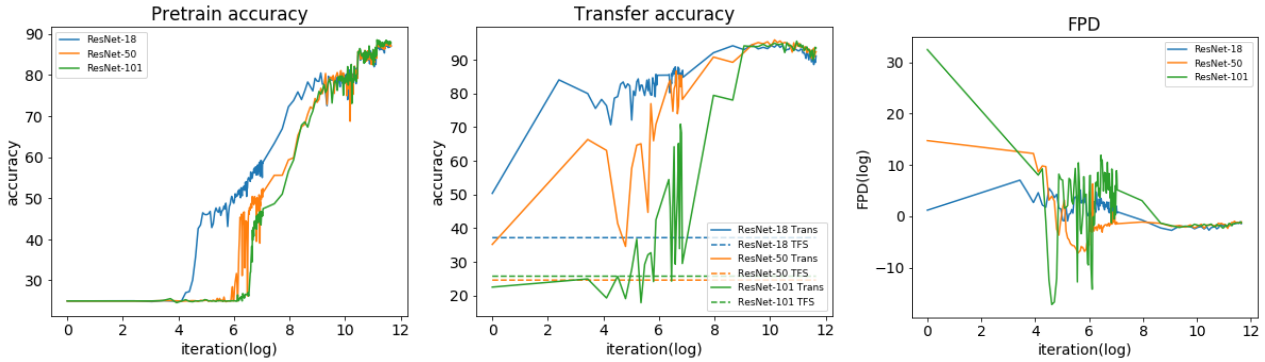
With the help of Fréchet Pre-train Distance, we are able to find an optimal pre-trained checkpoint in the pre-training stage before conducting actual fine-tuning experiments. Algorithm 1 shows how the process works.

4. Investigating Factors Affecting Neural Networks' Transferability

Equipped with Fréchet Pre-train Distance, we investigate more factors that affect the transferability of neural networks. Specifically, we investigate the over-parameterization problem that might cause the degradation of transferability, and propose to recover it with Spectral Normalization. We also explore the influence of learning



(a) CIFAR100-CIFAR10



(b) SVHN-MNIST

Figure 5: Transfer performance vs Different ResNets (Trans: transfer performance, TFS: train-from-scratch).

rates and neural network depth on the transfer results.

4.1. Improving the Transferability of AlexNet with Spectral Normalization

Spectral Norm Section 3.1 shows that in the early pre-training stage, for AlexNet only, pre-trained checkpoints bring a negative effect to the target task. In other words, the pre-trained weights become worse initialization than random for the target network and fine-tuning cannot recover from it. To prevent the degraded weights, inspired by [16], we propose to use Spectral Normalization (SN), which was introduced to improve the generalization of neural networks by reducing the sensitivity to test data perturbation ξ. Specifically, the spectral norm of weight matrix $A \in R^{m \times n}$ is defined as

$$\sigma(A) = \max_{\xi \in \mathbb{R}^n, \xi \neq 0} \frac{||A\xi||_2}{||\xi||_2} \tag{3}$$

which corresponds to the largest singular value of A. It has been proven [27] that for each weight matrix W^l of layer l in f_θ, in order to bound the spectral norm of $W_{\theta,x}$, it suffices to bound the spectral norm of W^l for each $l \in \{1, ...L\}$.

$$\begin{aligned} \sigma(W_{\theta,x}) \leq \sigma(D^L_{\theta,x})\sigma(W^L)\sigma(D^{L-1}_{\theta,x})\sigma(W^{L-1}) \cdots \\ \sigma(D^1_{\theta,x})\sigma(W^1) \leq \prod_{l=1}^{L} \sigma(W^L) \end{aligned} \tag{4}$$

where D denotes the activation function and $\sigma(D^l_{\theta,x}) \leq 1$ for every $l \in \{1, ...L\}$. This suggests to use the spectral norm as a regularizer to improve the generalizability of deep neural networks [27].

Spectral Normalization While Spectral Norm Regularization bounds the spectral norm of the entire neural network. Miyato *et al*. propose to apply Spectral Normalization to each specific layer which requires the spectral norm of each layer satisfies the Lipschitz constraint $\sigma(W) = 1$ [16], which is defined as:

$$\hat{W}_{SN}(W)) := W/\sigma(W) \tag{5}$$

Where W is the weight matrix and $\hat{W}_{SN}$ denotes the normalized W.

In Spectral Normalization, when each layer is normalized, the Lipschitz of the entire network $||f||_{Lip}$ is bounded from above by 1 (check [16] for more details). While in our experiments, we find when applying Spectral Normalization on different layers in AlexNet, the transfer performance varies on both CIFAR100-CIFAR10 (sub-figure 3a) and SVHN-MNIST (Sub-figure 3b) settings. The left column of Figure 3 shows how the transfer performance varies when we apply Spectral Normalization on the five convolution layers, the two 4096×4096 middle fully-connected layers, and the last classification layer. When the classification layer is normalized by Spectral Norm, the transferability is better recovered. To investigate the reason behind, we test the value of Fréchet Pre-train Distance before each layer. If we look into the finishing stages in the layer-wide FPDs (sub-figures in the middle column of 3), we find that only the fully-connected layers show the same trend as the entire network. We wonder whether it is the two over-parameterized 4096×4096 layers that cause the transferability degradation since it is easier to over-fit a layer with more parameters. We then modify the number of channels in the two fully-connected layers and show the results in the right column of 3. It turns out when we choose smaller channel sizes (1024×1024 or 128×128), the transferability of the network is improved immediately. That supports our claim that over-parameterization hurts transferability.

4.2. The Effect of Learning Rates

For the experiments in Section 3, we use a consistent hyper parameters settings such as learning rate for both pre-training and transfer learning experiments. We follow mostly setting from the previous study [7]. However, we wonder whether the same conclusion can be achieved under different learning rates. In this section, we modify the learning rate in our experiments on AlexNet, VGG-16 and ResNet-18 on both CIFAR100-CIFAR10 and SVHN-MNIST transfer learning. To verify our assumption, three different initial learning rates are tested (with same decay strategy) and the correlated results are shown in Figure 4.

From Figure 4 we find different networks and datasets perform slightly differently. For AlexNet and ResNet-18, the transfer performances in different pre-training stages are consistent when different learning rates are used. But for VGG-16, when learning rate equals to 0.5, the transfer performance in the starting stage is much lower than other learning rates but the performance in the finishing stage is better than others. For ResNet-18, the transfer performance changes less when we modify the learning rates. Additionally, the results are more noisy when a large learning rate is used. Besides, for three networks and two dataset settings, the change trending of the transfer performance is consistent across different networks and datasets but the degree varies. This is also reasonable since learning rates affect the speed of over-fitting, and the over-fitting speed further affects the transferability.

4.3. The Effect of ResNet Depths

ResNet [10] is one of the most popular neural networks that has been widely used nowadays among many different tasks. In this section, we would like to investigate how pre-training, transfer learning, and Fréchet Pre-train Distance change across different ResNets. We choose ResNet-18, ResNet-50, and ResNet-101 in our experiments. The experiments are also conducted on both CIFAR100-CIFAR10 and SVHN-MNIST datasets. Figure 5 shows the results. The left sub-figures show the comparison of pre-training accuracy, the middle sub-figures show transfer accuracy, and the right sub-figures show the results on Fréchet Pre-train Distance.

It is interesting that all ResNets reach the similar pre-training performance, transfer performance and Fréchet Pre-train Distance in the finishing training stages. But in the early stages, the pre-trainings of ResNet50 and ResNet101 are slower and nosier. It shows that ResNet-18 is parameterized enough for both CIFAR100-CIFAR10 and SVHN-MNIST. Therefore, increasing the depth of ResNet would not learn extra knowledge. On the contrary, when the network is over-parameterized compared to the complexity of the task. Not only the neural networks converge slower, but also the training process becomes noisier. The is compatible to the conclusion we achieve in Section 4.1, when AlexNet experiences a transfer performance degradation in the starting pre-training stage, which is also caused by the over-parameterization of the fully-connected layers in AlexNet.

5. Conclusion

In this paper, we explore the transferability of deep neural networks. We find that a pre-trained checkpoint that achieves the best performance on a source task would not always lead to a better transfer performance on target tasks, sometimes even cause a transfer degradation where pre-training would be worse than train-from-scratch. It shows the transferability of a pre-trained checkpoint is affected by the pre-training on both the beneficial and harmful sides for the downstream tasks. We propose a metric named Fréchet Pre-train Distance to evaluate the transferability of a pre-trained checkpoint by measuring the Fréchet distance of feature distributions between the source and target datasets. With the help of Fréchet Pre-train Distance, we would be able to identify a proper pre-trained checkpoint as the initialization for the target tasks before conducting transfer learning. Moreover, we investigate other factors that affect transfer learning and discuss the causes of the transferability degradation. In particular, we notice that over-fitting and over-parameterization hurt the transferability.

References

[1] Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep networks. 2018.

[2] Alessandro Achille and Stefano Soatto. Where is the information in a deep neural network? *arXiv preprint arXiv:1905.12213*, 2019.

[3] Jordan T Ash and Ryan P Adams. On the difficulty of warm-starting neural network training. *arXiv preprint arXiv:1910.08475*, 2019.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017.

[5] Yoshua Bengio. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning*, pages 17–36, 2012.

[6] Maurice Fréchet. Sur la distance de deux lois de probabilité. *COMPTES RENDUS HEBDOMADAIRES DES SEANCES DE L ACADEMIE DES SCIENCES*, 244(6):689–692, 1957.

[7] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[12] Stanislaw Jastrzebski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The break-even point on optimization trajectories of deep neural networks. *arXiv preprint arXiv:2002.09572*, 2020.

[13] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[15] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*, 2015.

[16] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[17] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[18] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights for deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 41(4):801–814, 2018.

[19] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[22] Leonid N Wasserstein. Markov processes over denumerable products of spaces describing large systems of automata. *Problems of Information Transmission*, 5(3):47–52, 1969.

[23] Jeremy West, Dan Ventura, and Sean Warnick. Spring research presentation: A theoretical foundation for inductive transfer. *Brigham Young University, College of Physical and Mathematical Sciences*, 1(08), 2007.

[24] Lei Wu, Chao Ma, and E Weinan. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. In *Advances in Neural Information Processing Systems*, pages 8279–8288, 2018.

[25] 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.

[26] LI Xuhong, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International Conference on Machine Learning*, pages 2825–2834, 2018.

[27] Yuichi Yoshida and Takeru Miyato. Spectral norm regularization for improving the generalizability of deep learning. *arXiv preprint arXiv:1705.10941*, 2017.

[28] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[29] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[30] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2720–2729, 2019.

[31] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[32] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. Interpreting cnns via decision trees. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6261–6270, 2019.