Effect of pre-training scale on intra- and inter-domain transfer for natural and X-Ray chest images

Mehdi Cherti, Jenia Jitsev
Juelich Supercomputing Center, Research Center Juelich
Helmholtz AI
Wilhelm-Johnen-Str, 52425 Juelich, Germany
{m.cherti, j.jitsev}@fz-juelich.de

Abstract

Recent line of work indicated strong improvement for transfer learning and model generalization when increasing model, data and compute budget scale in the pre-training. To compare effect of scale both in intra- and inter-domain full and few-shot transfer, in this study we combine for the first time large openly available medical X-Ray chest imaging datasets to reach a dataset scale comparable to ImageNet-1k. We then conduct pre-training and transfer to different natural or medical targets while varying network size and source data scale and domain, being either large natural (ImageNet-1k/21k) or large medical chest X-Ray datasets[1]. We observe strong improvement due to larger pre-training scale for intra-domain natural-natural and medical-medical transfer. For inter-domain natural-medical transfer, we find improvements due to larger pre-training scale on larger X-Ray targets in full shot regime, while for smaller targets and for few-shot regime the improvement is not visible. Remarkably, large networks pre-trained on very large natural ImageNet-21k are as good or better than networks pre-trained on largest available medical X-Ray data when performing transfer to large X-Ray targets. We conclude that high quality models for inter-domain transfer can be also obtained by substantially increasing scale of model and generic natural source data, removing necessity for large domain-specific medical source data in the pre-training.

1 Introduction

Re-using models obtained by pre-training on available source datasets to improve learning performance on upcoming target datasets is core idea behind transfer learning [1, 2], which was employed already at the very early rise of deep neural networks in the vision domain [3, 4]. Recent line of work on scaling in language modeling [5, 6] and vision [7, 8, 9] demonstrated strong improvement for model's ability to generalize or transfer on unseen target data when increasing model, data, and compute budget scale during the training.

The majority of the studies looking at the effect of pre-training scale on transfer deal with the intra-domain scenario scenario, where source and target data are close to each other, often originating from the same domain. This raises the question whether the observed positive effect of larger scale will also uphold in the inter-domain transfer scenario when using different types of source and target data, for instance natural and medical images [10, 11], that are not so closely related.

To address this, we conduct a series of large-scale pre-training and transfer experiments where we vary not only ResNet model [12, 7] and dataset size during pre-training, but also the domain of the source and the target datasets, being either natural or medical X-Ray chest images. This

[1] Code is available at: https://github.com/SLAMPAI/large-scale-pretraining-transfer

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

allows us to study effect of scale on both intra- and inter-domain transfer, also looking at both full and few-shot regime when using only few examples per class. To vary pre-training data scale for X-Ray domain, we combine here, for the first time, large openly available medical X-Ray chest imaging datasets (MIMIC-CXR [13], CheXpert [14], PadChest [15], NIH Chest X-ray14 [16]) into supersets. This provides scale comparable with ImageNet-1k [17], which we use alongside with much larger ImageNet-21k for natural domain pre-training. The pre-trained models of different scale are fine-tuned on various natural or X-Ray image target datasets. As large-scale pre-training requires heavy computational resources, we make use of a supercomputer tailored for distributed training to conduct our experiments (JUWELS Booster [18], see Supplementary for further details).

2 Methods

Large-scale pre-training. For pre-training, we largely followed the training procedure of [7]. We used both ResNet-50x1 and ResNet-152x4 (in following R50x1 and R152x4) from [7] on different natural image and medical datasets. Smaller R50x1 has 26M weight parameters, while larger R152x4 has 928M parameters. This substantial difference in size allows us to compare the effect of model scaling in the pre-training on subsequent transfer. For natural images, we took ImageNet-1k (≈ 1.4 Millions images) and the much larger full ImageNet-21k (≈ 14 Millions images), using a standard supervised classification setup with softmax as an output activation and cross entropy as a loss. We followed the training hyper-parameters of [7]. For optimization, we use stochastic gradient descent (SGD) with adaptive gradient clipping (AGC) from [19], as we found that it helps both pre-training and transfer.

For medical X-Ray chest data, we created supersets that may contain any set of available large multi-label X-Ray datasets: MIMIC-CXR, CheXpert, PadChest, NIH X-ray14 (112k, 160k, 224k and 377k samples each; also Suppl. Tab. 3 for more details on datasets). We refer to those as X-Ray supersets in following. The datasets are combined by finding intersecting disease labels and performing classification on those. We start with single available X-Ray datasets and progressively add other datasets into X-Ray supersets of successively growing size, which provides us with X-Ray source datasets spanning scales from small ($\approx$ 100-200k samples) to large ($\approx$ 873k samples) for pre-training. To process the datasets and extract the labels from raw data, we used TorchXRayVision [20] from the work of [21]. For pre-training, we followed [21], using a multi-label setup where we have independent binary tasks, one for each label (disease), and we used sigmoid as an output activation function and binary cross entropy as a loss for each label.

In order to speedup training, we used data parallel training with Horovod [22], using 256 A100 GPUs for R152x4 and 128 A100 GPUs for R50x1 models on natural images, while for X-Ray data 64 GPUs were taken. A pre-training on ImageNet-21k with large R152x4 takes about 81 hours using 256 GPUs, while with small R50x1 it needs about 13.5 hours to finish using 128 GPUs on JUWELS Booster supercomputer [18].

Fine-tuning and transfer evaluation. For fine-tuning, we follow [7]. We employ BiT-HyperRule - a heuristic that selects fine-tuning hyper-parameters (learning rate schedule, resolution, usage of MixUp, and total number of steps) based on training set size and image resolution. We used a batch size of 128, and an initial learning rate of 0.001 on all experiments, optimization procedure following otherwise that of pre-training. For each experiment, the classification head of the pre-trained model was replaced with a new classification head for the fine-tuning task, fine-tuning all the layers of the network. We perform 5 independent runs with different seeds to have an estimate of the variance of the performance.

For transfer, few-shot setup (we used 1, 5, 10, 100 or 500 examples per class) and full shot fine-tuning on the full training set are employed. We used CIFAR-10, CIFAR-100 [23], Flowers-102 [24], and Oxford-IIIT Pet [25] for natural image fine-tuning. For medical image fine-tuning, we used single-label Tuberculosis [26] and COVIDx [27] as small X-Ray targets ($\approx$ 800 and 16k samples each), and multi-label CheXpert, MIMIC-CXR, NIH or PadChest as larger X-Ray targets (magnitude order of 100k-300k samples). When performing medical-medical transfer experiments, the given X-Ray target dataset is excluded from the X-Ray supersets used for pre-training the models (see also Suppl. Tabs. 6, 7, 8, 9 for details on superset composition for each given target). In addition, to perform few-shot experiments similar to natural domain, we employ PadChest-cl, single-label dataset derived from PadChest, where we keep only images with exactly one label. For Flowers-102 and

Target (natural, medical)	ResNet-50x1 S-MED	L-MED	1K-NAT	21K-NAT	ResNet-152x4 S-MED	L-MED	1K-NAT	21K-NAT
CIFAR-10(1)	—	—	94.26 ± 0.05	**95.78 ± 0.09**	—	—	96.93 ± 0.05	**97.82 ± 0.07**
CIFAR-100(1)	—	—	75.90 ± 0.05	**82.47 ± 0.21**	—	—	83.90 ± 0.09	**88.54 ± 0.14**
Flowers-102(1)	—	—	74.94 ± 0.99	**98.21 ± 0.22**	—	—	89.41 ± 0.25	**99.49 ± 0.08**
Pets(1)	—	—	85.21 ± 0.58	**87.23 ± 0.18**	—	—	*93.32 ± 0.30*	*93.21 ± 0.14*
COVIDx(1)	68.50 ± 0.18	76.05 ± 0.21	*76.30 ± 1.30*	*78.35 ± 1.63*	78.65 ± 0.84	**83.00 ± 1.16**	78.10 ± 0.95	78.90 ± 0.49
Tuberculosis(1)	79.83 ± 0.45	81.65 ± 0.91	79.83 ± 1.50	**83.47 ± 0.83**	79.01 ± 0.45	**90.91 ± 0.83**	81.49 ± 2.23	80.83 ± 2.51
MIMIC CXR(2)	84.17 ± 0.03	86.38 ± 0.03	85.41 ± 0.10	**86.82 ± 0.10**	87.63 ± 0.04	**88.00 ± 0.03**	86.85 ± 0.06	87.79 ± 0.13
CheXpert(2)	82.10 ± 0.07	*86.66 ± 0.05*	84.83 ± 0.14	*86.60 ± 0.14*	84.92 ± 0.07	*87.82 ± 0.03*	86.82 ± 0.06	*87.77 ± 0.07*
PadChest(2)	68.06 ± 0.24	68.14 ± 0.21	76.72 ± 0.27	**80.99 ± 0.22**	75.91 ± 0.12	75.23 ± 0.17	79.59 ± 0.17	**83.94 ± 0.19**
PadChest-Cl(2)	73.01 ± 0.13	78.33 ± 0.08	80.17 ± 0.17	**82.03 ± 0.17**	81.79 ± 0.07	82.68 ± 0.05	82.55 ± 0.05	**84.02 ± 0.24**
NIH (2)	70.11 ± 0.15	74.21 ± 0.57	75.53 ± 0.47	**81.02 ± 0.57**	77.95 ± 0.13	78.95 ± 0.13	79.82 ± 0.38	**82.80 ± 0.41**

Table 1: **Varying model and data pre-training scale for intra- and inter-domain transfer.** Pre-training is performed with either natural or medical source data (ordered by increasing scale) being one of large X-Ray datasets (S-MED), compositional X-Ray superset (L-MED), ImageNet-1k (1K-NAT), ImageNet-21k (21K-NAT), using either small ResNet-50x1 or large ResNet-152x4. (1) - Top-1 Acc [%] metric; (2) - mean AUC metric. Bold indicates best transfer performance for a fixed network size and pinpoints the effect of data scale on transfer. Italics indicates transfer performance with no significant difference between data scale. Red indicates best overall performance for a given target.

COVIDx, since the datasets are strongly imbalanced, we used oversampling. We measure either final accuracy or mean AUC on the test sets.

3 Results.

Effect of scale on intra-domain transfer. Results we obtain either for natural-natural or medical-medical full shot transfer deliver a clear picture showing transfer improvement across target datasets when increasing pre-training model and data scale, the improvement due to increase of network size being most consistent (as indicated by outcomes in red in (Tab. 1).

For few-shot transfer, we observe a differentiated picture. In line with previous work, for natural-natural transfer we obtain strong improvement due to larger scale in the very low data regime of 1- or 5-shot transfer (eg. for CIFAR-100, Fig. 1a). In contrast, for medical-medical scenario, there is no consistent few-shot transfer improvement due to larger scale (Fig. 1b; Suppl. Fig. 6b, 6d, 7b). Increasing number of shots and approaching full shot regime, the improvement due to scale becomes more and more visible.

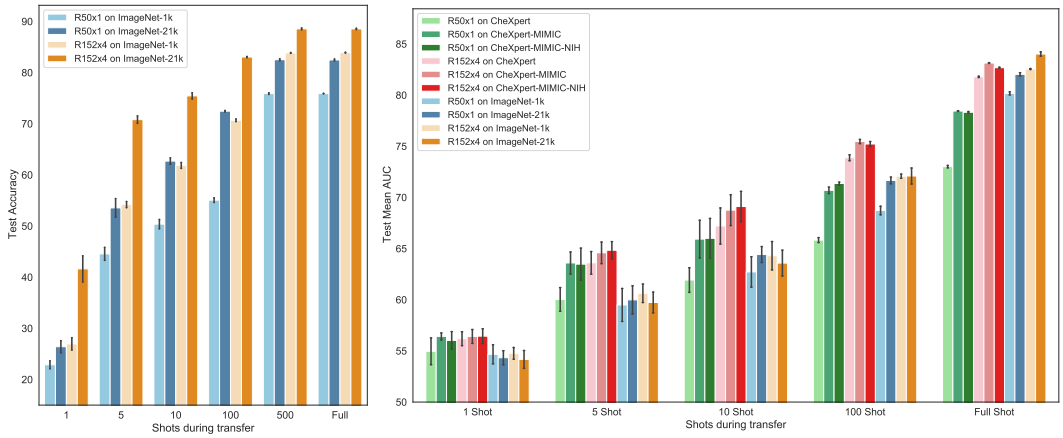
Effect of scale on inter-domain transfer. Here we transfer on either small or large medical X-Ray chest imaging targets after pre-training on ImageNet-1k (1.4M samples) or much larger ImageNet-21k (14M samples). For all large X-Ray targets (MIMIC-CXR, CheXpert, PadChest, PadChest-cl or NIH) we observe clear full-shot transfer improvement due to larger pre-training scale (Tab. 1). The effect is consistent for both model and data scale across large X-Ray targets.

For small X-Ray targets (Tuberculosis and COVIDx), we do not observe such consistent improvement due to larger scale. For instance, while we see improvement due to larger data scale for small ResNet-50x1 on both small targets, the improvement is not there when increasing network size. There is also no evidence for positive effect of larger scale on few-shot transfer, neither for large nor for small X-Ray targets (Fig. 1b; Suppl. Fig. 6a, 6c, 7a).

Remarkably, when further comparing intra- and inter-domain transfer performance, we observe that large ResNet-152x4 pre-trained on very large generic natural ImageNet-21k are as good or better than networks pre-trained on largest available medical domain specific X-Ray superset data when performing full shot transfer to large X-Ray targets (Tab. 1, Fig. 1b). This fits into overall picture of larger model and data pre-training scale improving transfer on larger targets observed here, as ImageNet-21k has order of magnitude larger scale than the largest X-Ray superset constructed for this study.

4 Conclusion & Outlook

We present here evidence that substantially increasing model and data scale in the pre-training benefits both intra- and inter-domain transfer across various target datasets from natural and medical X-Ray image domain. The effect of pre-training scale on transfer performance depends on transfer scenario.



(a) CIFAR-100 (b) PadChest-Cl

Figure 1: Few- and full shot transfer performance on a natural and a medical X-Ray target when varying model and data scale in pre-training. Each color represents a combination of model scale and data scale (and domain, in **(b)**) during pre-training.

Transfer improvement due to larger pre-training scale is found to be substantial in natural-natural or medical-medical, intra-domain transfer scenarios where source and target datasets are closely related, being especially strongly pronounced in the few-shot transfer regime for natural-natural case and concentrated in full-shot scenario in medical-medical case. For natural-medical inter-domain transfer, clear positive effect of larger pre-training scale is evident for full shot transfer on large X-Ray targets. On small X-Ray targets and for few-shot transfer regime, no clear inter-domain transfer improvements are observed. Remarkably, the largest ResNet-152x4 network pre-trained on very large generic natural ImageNet-21k matches or even outperformes networks pre-trained on largest medical domain-specific X-Ray superset data combined for this study when performing full shot transfer to large X-Ray targets. This is relevant for the practice, where inter-domain transfer is often the only viable option, as large volumes of medical domain-specific data may be not available for pre-training. Here we show that high quality models for large X-Ray targets can also be obtained via inter-domain transfer when substantially increasing pre-training model and generic natural image source data scale, instead of relying on large domain-specific X-Ray chest imaging data.

The study offers different follow-up directions, like experimenting with larger scale both for network and data size, mixing natural and medical source data for pre-training or using different network architectures like transformers. Following these directions would pave the path towards scaling laws and enabling systematic prediction of transfer performance and improvement due to increase of pre-training scale in the important inter-domain setting, where source and target are further apart.

Broader and Social Impact

Our work aims on advancing transfer learning, which can make learning algorithms perform better and more efficient by re-using models already pre-trained on various tasks and therefore requiring less compute and data to learn solutions for other relevant tasks. The approach to improve transfer learning by increasing scale of the pre-training is generic and has impact far beyond vision domain, for instance in language modeling, and is not bound to any specific application. As any generic method, it can be therefore applied to enhance technologies for sensitive applications, for instance in health domain or in public surveillance, that may have both strong positive and negative social impact, depending on policies introduced on their usage. Special care should be taken about applications in clinical domain where further development of diagnostic tools based on data driven machine learning should be accompanied by a broad panel of experts from corresponding domains. The method depends on computationally heavy large-scale pre-training that is energy demanding on the one hand. On the other hand, it contains a promise to pay off the energy budget put into training by obtaining generic models that can be very efficiently adapted to a large range of problems via transfer, saving computational and energy costs that would otherwise incur for their solution from scratch.

Acknowledgments and Disclosure of Funding

We would like to express gratitude to all the people who are working on making code, models and data publicly available, advancing community based research and making research more reproducible. Special thanks go to creators and maintainers of open available X-Ray medical imaging datasets that also enabled our research, some of those gathered under difficult circumstances of the COVID-19 pandemics. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this work by providing computing time through the John von Neumann Institute for Computing (NIC) on the GCS Supercomputers JUWELS, JUWELS Booster at Jülich Supercomputing Centre (JSC). We also acknowledge computing resources from the Helmholtz Data Federation and further computing time provided on supercomputer JUSUF in frame of offer for epidemiology research on COVID-19 by JSC.

References

[1] Lorien Y Pratt, Jack Mostow, Candace A Kamm, and Ace A Kamm. Direct transfer of learned information among neural networks. In *AAAI*, volume 91, pages 584–589, 1991.

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

[3] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, pages 512–519, June 2014.

[4] Hossein Azizpour, Ali Sharif Razavian, Josephine Sullivan, Atsuto Maki, and Stefan Carlsson. Factors of transferability for a generic convnet representation. *IEEE transactions on pattern analysis and machine intelligence*, 38:1790–1802, September 2016.

[5] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[7] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 491–507, Cham, 2020. Springer International Publishing.

[8] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. *arXiv preprint arXiv:2104.10972*, 2021.

[9] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. *arXiv preprint arXiv:2106.04560*, 2021.

[10] Maithra Raghu, Chiyuan Zhang, Jon Kleinberg, and Samy Bengio. Transfusion: Understanding transfer learning for medical imaging. In *Advances in Neural Information Processing Systems 32*, pages 3347–3357. 2019.

[11] Basil Mustafa, Aaron Loh, Jan Freyberg, Patricia MacWilliams, Alan Karthikesalingam, Neil Houlsby, and Vivek Natarajan. Supervised transfer learning at scale for medical imaging. *arXiv preprint arXiv:2101.05913*, 2021.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016.

[13] Alistair E. W. Johnson, Tom J. Pollard, Seth J. Berkowitz, Nathaniel R. Greenbaum, Matthew P. Lungren, Chih-Ying Deng, Roger G. Mark, and Steven Horng. Mimic-cxr, a de-identified

publicly available database of chest radiographs with free-text reports. *Scientific data*, 6:317, December 2019.

[14] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 590–597, 2019.

[15] Aurelia Bustos, Antonio Pertusa, Jose-Maria Salinas, and Maria de la Iglesia-Vayá. Padchest: A large chest x-ray image dataset with multi-label annotated reports. *Medical image analysis*, 66:101797, December 2020.

[16] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pages 3462–3471, 2017.

[17] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 248–255, June 2009.

[18] Juelich Supercomputing Center. JUWELS Booster Supercomputer, 2020. `https://apps.fz-juelich.de/jsc/hps/juwels/configuration.html#hardware-configuration-of-the-system-name-booster-module`.

[19] Andrew Brock, Soham De, Samuel L Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. *arXiv preprint arXiv:2102.06171*, 2021.

[20] Joseph Paul Cohen, Joseph Viviano, Paul Morrison, Rupert Brooks, Mohammad Hashir, and Hadrien Bertrand. TorchXRayVision: A library of chest X-ray datasets and models. *https://github.com/mlmed/torchxrayvision*, 2020.

[21] Joseph Paul Cohen, Mohammad Hashir, Rupert Brooks, and Hadrien Bertrand. On the limits of cross-domain generalization in automated x-ray prediction. In *Medical Imaging with Deep Learning*, pages 136–155. PMLR, 2020.

[22] Alexander Sergeev and Mike Del Balso. Horovod: fast and easy distributed deep learning in tensorflow. *arXiv preprint arXiv:1802.05799*, 2018.

[23] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[24] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[25] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505, June 2012.

[26] Stefan Jaeger, Sema Candemir, Sameer Antani, Yì-Xiáng J. Wáng, Pu-Xuan Lu, and George Thoma. Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quantitative imaging in medicine and surgery*, 4:475–477, December 2014.

[27] Linda Wang, Zhong Qiu Lin, and Alexander Wong. Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. *Scientific reports*, 10:19549, November 2020.

Supplementary: Effect of pre-training scale on intra- and inter-domain transfer for natural and X-Ray chest images

A Distributed Training

A.1 JUWELS Booster Supercomputer

JUWELS Booster [18] features 936 compute nodes that host four NVIDIA A100 GPUs each, providing 3744 GPUs in total. The installed A100 Tensor Core GPUs (40 GB) provide 19.5 TFLOP/s of FP64_{TC} computing performance each. The GPUs are hosted by AMD EPYC 7402 CPUs with 2×24 cores (SMT-2) per node, clocked with 2.8 GHz. Each node is diskless and is equipped with 512 GB of RAM. The network of JUWELS Booster is based on Mellanox HDR200 InfiniBand, with four Mellanox ConnectX 6 devices per node, each providing 200 Gbit/s bandwidth per direction.

The NVIDIA A100 GPUs installed into JUWELS Booster reach peak efficiency of 48.75 GFLOP/(s W) when utilizing the FP64 Tensor Cores. This makes JUWELS Booster rank highest in the Green500 list of November 2020 as the most energy efficient supercomputer among the first 100 machines of the Top500 list with 25 GFLOP/(s W).

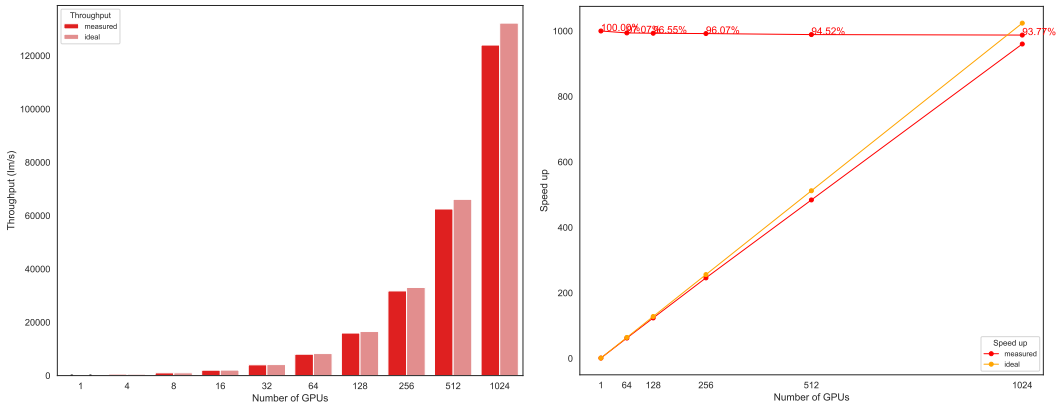
A.2 Scaling and training time

Here, we report scaling behavior during large-scale pre-training for ResNet networks we used in the experiments.

We performed scaling experiments to assess the scalability of data parallel training distributed across many GPUs on multiple nodes using Horovod. The efficiency in Figure 2b (upper part of the figure with percentages) is computed using the following formula: $E(N) = 100 \times \frac{T(N)}{N \times T(1)}$. $T(N)$ is the total measured throughput in Im/s for N GPUs. The best achievable efficiency, when scaling is perfect, is 100%.

We also provide the raw throughput (Im/s) numbers in Figure 2a and Tab. 2. On 1024 GPUs, we achieve an efficiency of $\approx 93.7\%$ with single precision (FP32). To make sure distributed training is stable, we check the end accuracy of full training for each number of GPUs to reassure we reach target accuracy acceptable for standard ImageNet-1k Top-1 and Top-5 results.

Achieved scaling on JUWELS Booster allows to perform full pre-training on ImageNet-21k with large R152x4 in about 81 hours using 256 GPUs. For small R50x1, full training needs about 13.5 hours to finish using 128 GPUs.



(a) Throughput in Im/s (b) Scaling efficiency

Figure 2: Distributed training for R152x4, scaling behavior on JUWELS Booster using A100 GPUs.

Table 2: Scaling behavior in Im/s of ImageNet-1k training using ResNet-152x4 architecture from [7] with batch size 128. For each GPU, one MPI process is assigned. Computations were done on up to 256 nodes on JUWELS Booster. Throughput performance during training is reported for single precision mode (FP32). The corresponding speedup is provided relative to reference training with 1 GPU. Note that the measured Im/s throughput includes I/O.

#GPUs	Im/s	speedup
1	129.14	1.00
4	508.00	3.93
8	1009.30	7.82
16	2023.78	15.67
32	4029.69	31.21
64	8022.31	62.12
128	15959.86	123.59
256	31758.59	245.93
512	62496.35	483.96
1024	124003.59	960.26

B Additional details on experimental results

B.1 Datasets employed in experiments.

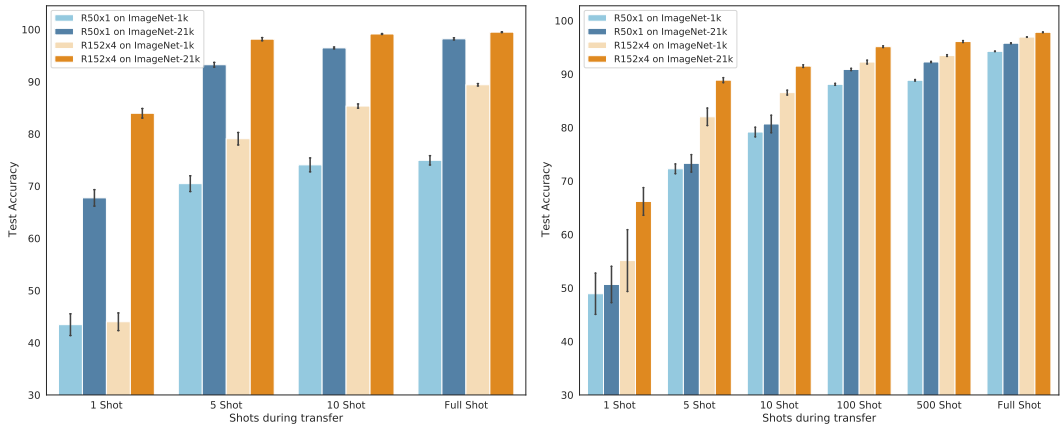
Table 3: Datasets used as source for pre-training and target for transfer. The url of each dataset we will use is provided below.

Dataset	Size
Source pre-training	
Natural Images	
ImageNet-1k [17]	1.4M images, 1000 classes
ImageNet-21k [17]	14M images, 21842 classes
X-Ray Chest Imaging	
CheXpert [14]	224K radiographs of 65K patients, 14 classes
NIH Chest X-ray14 [16]	112K radiographs of 32K patients, 14 classes
PadChest [15]	160K radiographs of 67K patients, 19 classes
MIMIC-CXR [13]	377K radiographs of 65K patients, 14 classes
Total X-Ray images	**873K** chest radiographs, 229K patients
Target transfer	
Natural Images	
CIFAR-10, 100 [23]	60K images, 10,100 classes
Oxford Flowers-102 [24]	8K images, 102 classes
Oxford-IIIT Pet [25]	7.3K images, 37 classes
X-Ray Chest Imaging	
PadChest [15]	160K radiographs of 67K patients, 19 / 27 classes
COVIDx [27]	16K radiographs, 15K patients, 2 / 3 classes
Tuberculosis [26]	800 radiographs, 800 patients, 2 classes

All datasets employed in our experiments are publicly available and can be obtained following links in the Tab. 3

B.2 Further transfer results

Here, we present more detailed results of transfer experiments described in the main document. For medical X-Ray targets, we provide tables reporting transfer performance (Tabs. 5, 6, 7, 8, 9) listing each source X-Ray dataset and supersets used for pre-training, as outlined in the experiments description in the main document.



(a) Flowers-102 (b) CIFAR-10

Figure 3: Few-shot and full shot transfer performance on target datasets when varying model size and dataset size in pre-training. Transfer improvement due to model and source data size is evident, especially strongly pronounced in few-shot regime.

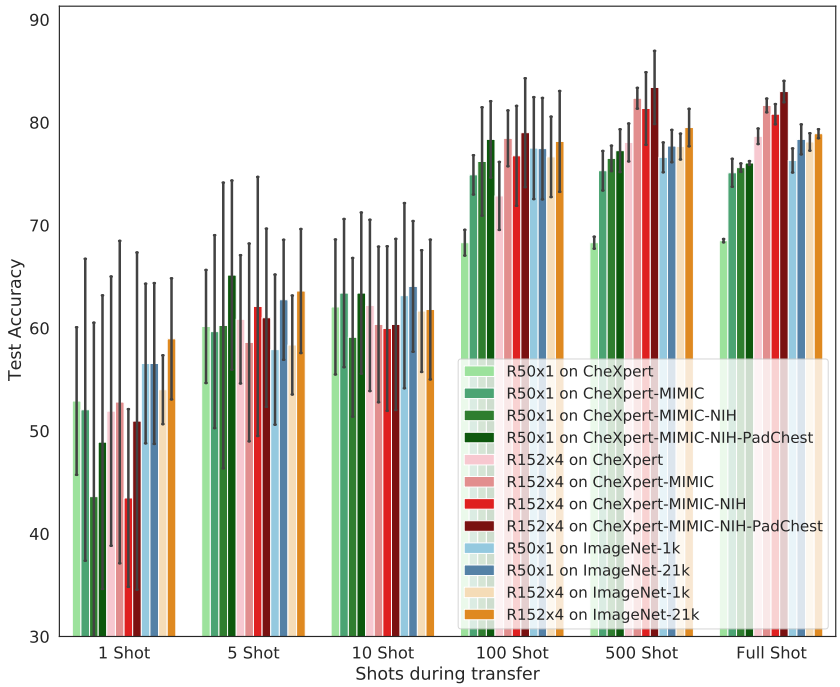


Figure 4: Few-shot and full shot transfer performance on COVIDx dataset when varying model, source data size and domain in pre-training. In the full shot transfer, improvement due to model and data scale is evident when pre-training on X-Ray chest imaging source data. In few-shot regime, no transfer improvement due to larger model or data size is observed.

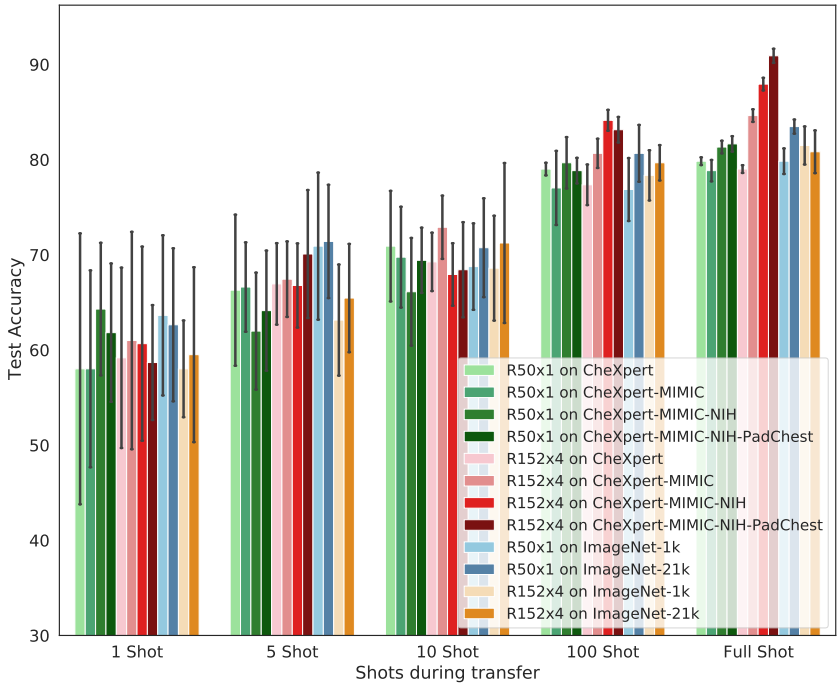
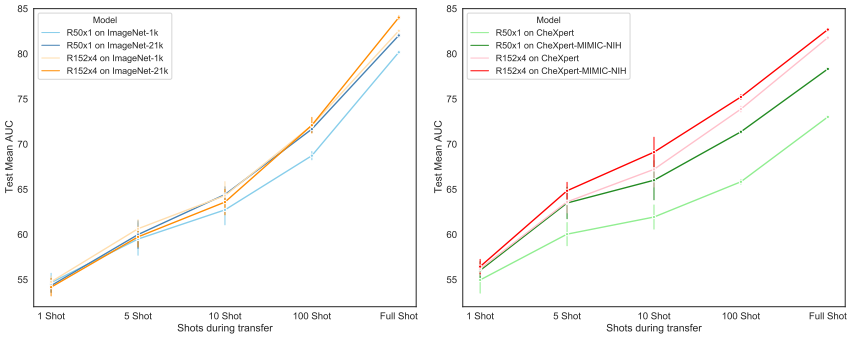
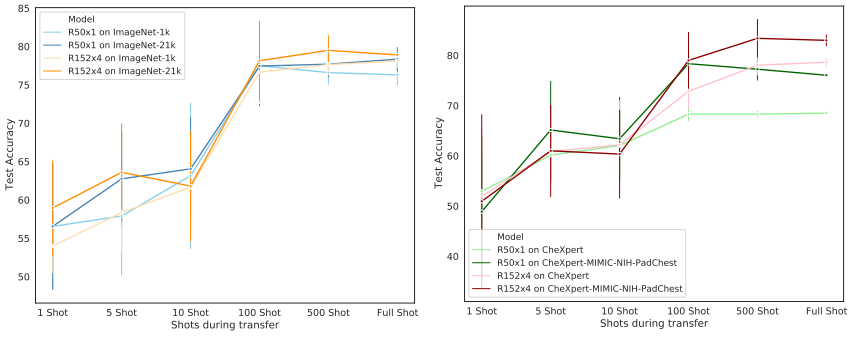


Figure 5: Few-shot and full shot transfer performance on Tuberculosis dataset when varying model, source data size and domain in pre-training. In the full shot transfer, improvement due to model and data scale is evident when pre-training on X-Ray chest imaging source data. In few-shot regime, no transfer improvement due to larger model or data size is observed.



(a) PadChest-Cl, natural source

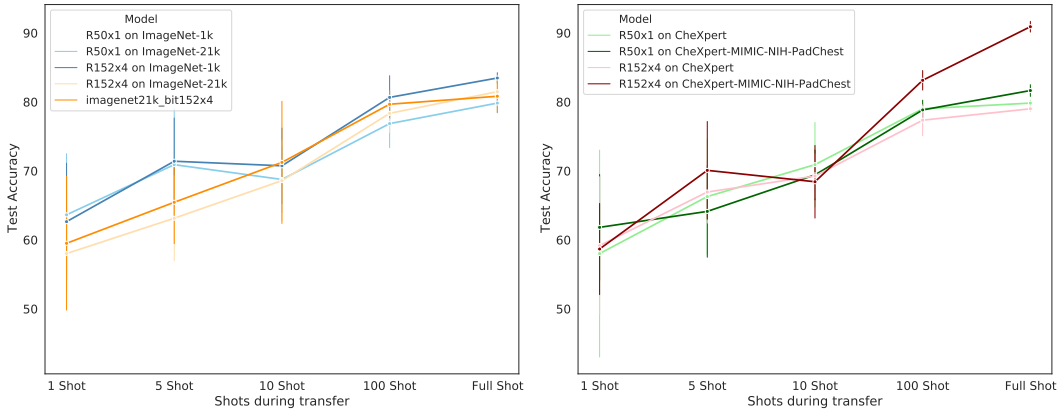
(b) PadChest-Cl, medical source



(c) COVIDx, natural source

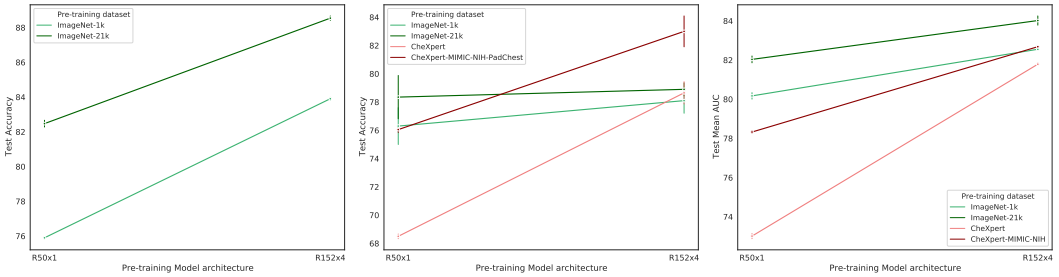
(d) COVIDx, medical source

Figure 6: Few-shot and full shot transfer performance on medical X-Ray targets of different size for intra- and inter-domain scenarios, medical-medical or natural-medical. Each color represents a combination of model and data scale during pre-training.



(a) Tuberculosis, natural sources (b) Tuberculosis, medical sources

Figure 7: Few-shot and full shot transfer performance on Tuberculosis dataset when pre-training with different model sizes on different sources (natural or medical datasets) of various sizes. In natural-medical scenario **(a)**, no transfer improvement due to model or data scale is evident. In medical-medical scenario **(b)**, larger model and data size lead to transfer improvement in full shot regime, without benefits in few-shot mode.



(a) CIFAR-100 (b) COVIDx (c) PadChest-Cl

Figure 8: Full shot transfer performance on target datasets when varying model and source data size, taking the smallest and largest pre-training datasets available for each domain.

Table 4: **Intra- and inter-domain transfer using natural ImageNet-1k and ImageNet-21k for pre-training with different sized ResNets** (1) - Top-1 Acc [%] metric; (2) - mean AUC metric. Bold indicates best transfer performance for a fixed network size and pinpoints the effect of data scale on transfer. Italics indicates transfer performance with no significant difference between data scale. Red indicates best overall performance for a given target. Clear transfer improvement emerges for natural-natural scenario due to both model and data scale. For natural-medical scenario the positive effect of larger scale is consistently given for larger targets, but not for smaller ones. For instance, for very small Tuberculosis target, larger data scale improves transfer for small ResNet-50x1, while larger model scale does not lead to any transfer improvement.

Target Domain	Dataset	ResNet-50x1		ResNet-152x4	
		1K	21K	1K	21K
Natural	CIFAR-10 (1)	94.26 ± 0.05	**95.78 ± 0.09**	96.93 ± 0.05	**97.82 ± 0.07**
	CIFAR-100 (1)	75.90 ± 0.05	**82.47 ± 0.21**	83.90 ± 0.09	**88.54 ± 0.14**
	Flowers-102 (1)	74.94 ± 0.99	**98.21 ± 0.22**	89.41 ± 0.25	**99.49 ± 0.08**
	Pets (1)	85.21 ± 0.58	**87.23 ± 0.18**	*93.32 ± 0.30*	*93.21 ± 0.14*
Medical	Tuberculosis (1)	79.83 ± 1.50	***83.47 ± 0.83***	*81.49 ± 2.23*	*80.83 ± 2.51*
	COVIDx (1)	*76.30 ± 1.30*	*78.35 ± 1.63*	*78.10 ± 0.95*	*78.90 ± 0.49*
	NIH (2)	75.53 ± 0.47	**81.02 ± 0.57**	79.82 ± 0.38	**82.80 ± 0.41**
	PadChest-Cl (2)	80.17 ± 0.17	**82.03 ± 0.17**	82.55 ± 0.05	**84.02 ± 0.24**
	PadChest (2)	76.72 ± 0.27	**80.99 ± 0.22**	79.59 ± 0.17	**83.94 ± 0.19**
	CheXpert (2)	84.83 ± 0.14	**86.60 ± 0.14**	86.82 ± 0.06	**87.77 ± 0.07**
	MIMIC CXR (2)	85.41 ± 0.10	**86.82 ± 0.10**	86.85 ± 0.06	**87.79 ± 0.13**

Table 5: **Intra-domain transfer using different sized medical X-Ray source data for pre-training with different sized ResNets** (1) - Top-1 Acc [%] metric; (2) - mean AUC metric. "+" indicates addition into a successively larger source superset. Clear transfer improvement is evident due to larger model and data scale across different targets.

Target	ResNet-50x1				ResNet-152x4			
	CheXpert	+MIMIC	+ NIH	+PadChest	CheXpert	+MIMIC	+NIH	+PadChest
PadChest-Cl[2]	73.01 ± 0.13	*78.44 ± 0.04*	*78.33 ± 0.08*	—	81.79 ± 0.07	**83.14 ± 0.04**	82.68 ± 0.05	—
COVIDx[1]	68.50 ± 0.18	75.10 ± 1.52	75.60 ± 0.45	**76.05 ± 0.21**	78.65 ± 0.84	81.65 ± 0.74	80.80 ± 1.10	**83.00 ± 1.16**
Tuberculosis[1]	79.83 ± 0.45	78.84 ± 1.25	*81.32 ± 0.74*	*81.65 ± 0.91*	79.01 ± 0.45	84.63 ± 0.74	87.93 ± 0.74	**90.91 ± 0.83**

Table 6: **Intra-domain transfer using different sized medical X-Ray source data for pre-training with different sized ResNets, target MIMIC-CXR** Mean AUC metric. "+" indicates addition into a successively larger source superset. Clear transfer improvement is evident by scaling the model size. Using a superset containing CheXpert and PadChest improves the results, but adding NIH does not or does very little, this could be explained by the fact that NIH is the smallest dataset among the medical pre-training datasets, and a larger increase in the superset would be needed to substantially improve the transfer results, as it has been observed in transfer results that were obtained using models pre-trained on much larger natural data.

Target	ResNet-50x1			ResNet-152x4		
	CheXpert	+PadChest	+ NIH	CheXpert	+PadChest	+NIH
MIMIC CXR	84.17 ± 0.03	86.19 ± 0.03	**86.38 ± 0.03**	87.63 ± 0.04	**88.13 ± 0.03**	88.00 ± 0.03

Table 7: **Intra-domain transfer using different sized medical X-Ray source data for pre-training with different sized ResNets, target CheXpert** Mean AUC metric. "+" indicates addition into a successively larger source superset. Clear transfer improvement is evident by scaling the model size. Using a superset containing PadChest and MIMIC CXR improves the results, adding NIH does not lead to further improvement. This could be explained by the fact that NIH is the smallest dataset among the medical pre-training datasets, and a larger increase in the superset would be needed to substantially improve the transfer results, as it has been observed in transfer results that were obtained using models pre-trained on much larger natural data.

Target	ResNet-50x1			ResNet-152x4		
	PadChest	+MIMIC	+ NIH	PadChest	+MIMIC	+NIH
CheXpert	82.10 ± 0.07	*86.56 ± 0.08*	*86.66 ± 0.05*	84.92 ± 0.07	**88.03 ± 0.03**	87.82 ± 0.03

Table 8: **Intra-domain transfer using different sized medical X-Ray source data for pre-training with different sized ResNets, target PadChest** Mean AUC metric. "+" indicates addition into a successively larger source superset. Clear transfer improvement is evident by scaling the model size. Improvement by increasing data size is not evident and only happens using the small R50x1 model and a superset containing CheXpert and MIMIC, adding NIH (which is smaller compared to CheXpert and MIMIC) the superset does not help further. This indicates that larger increase in the superset may be necessary to further improve the transfer results, as it has been observed when using models pre-trained on much larger natural data.

Target	ResNet-50x1			ResNet-152x4		
	CheXpert	+MIMIC	+ NIH	CheXpert	+MIMIC	+NIH
PadChest	68.06 ± 0.24	**70.07 ± 0.49**	68.14 ± 0.21	*75.91 ± 0.12*	*75.81 ± 0.07*	75.23 ± 0.17

Table 9: **Intra-domain transfer using different sized medical X-Ray source data for pre-training with different sized ResNets, target NIH** (Mean AUC metric). "+" indicates addition into a successively larger source superset. Clear transfer improvement is evident by scaling the model size. We also observe transfer improvement by scaling data size, however the improvement seems to flatten. Since the transfer results using pre-trained models on larger natural data show a better performance, this indicates that a larger superset scale may be necessary to further improve transfer.

Target	ResNet-50x1			ResNet-152x4		
	CheXpert	+PadChest	+ MIMIC	CheXpert	+PadChest	+MIMIC
NIH	70.11 ± 0.15	*73.37 ± 0.38*	*74.21 ± 0.57*	77.95 ± 0.13	78.16 ± 0.13	**78.95 ± 0.13**

C Code and Data availability

Repository containing code used for running experiments and producing figures in this study can be found at `https://github.com/SLAMPAI/large-scale-pretraining-transfer`. All datasets used in the study are openly available and are listed together with references to the original work in the Table 3. Further details on the usage of the datasets for conducting and reproducing experiments in this study are also provided in the linked repository.