In-domain Model Pre-training for Chest X-Rays

Vasudev Awatramani, Akshama, Alexander Miller
{va2134,ax2028,ahm9968}@nyu.edu

Abstract
Pretraining on large datasets has proven to be critical to good performance on many image classification tasks. However, the nature of the pretraining task matters a lot: conventional wisdom says that the closer to the downstream task, the more useful the pretraining will be. We evaluate the effectiveness of using imagenet pretraining in contrast to pretraining on a different in-domain dataset in the setting of pathology detection in chest x-ray images and show that in-domain pretraining slightly helps performance on bigger ResNet models (ResNet50) but slightly hurts performance on smaller DenseNet models (DenseNet121). Not only that, but ImageNet pretraining did the opposite: it helped the smaller model but hurt the bigger model in contrast to even zero pretraining.[1]

1. Introduction

Pre-training on larger datasets has become a staple of the machine learning researcher's pipeline across computer vision [1,2,3,4], NLP [5,6,7], and more. This technique has massively improved performance again and again on downstream tasks. First, the model learns transferable, generic features on a pretraining dataset - often using a loss particular to the pretraining task such as predicting hashtags assigned to an image by Instagram users [4] or Masked Language Modeling [5]. There are several factors of primary importance in pre-training:

1) How much pretraining data do you have? What is the quality of this data?
2) How similar is the pretraining data to the downstream task data?
3) What is the loss function? How is this different from the loss on the downstream task?

The more high quality pretraining data you have, the more similar the data, and the more similar the loss function, the better you can expect to perform on your downstream task [4].

In this work, we focus on radiological data; in particular, we look at chest x-rays. This kind of data represents a massive distribution shift from the main datasets used for pretraining in typical image tasks: instead of pictures of everyday objects or scenery, these images use different color profiles and capture content impossible to find in typical photographed experiences. We hypothesize that this kind of shift renders typical pretraining useless and evaluate the effects of different pretraining regimes on these tasks: random initialization, imagenet pretraining, and pretraining on a larger dataset of chest x-rays with annotation targets.

2. Related Work

Various works have been proposed to automatically classify thoracic diseases from CXRs, thanks to the public release of

[1] Code available at github.com/alexholdenmiller/nyu_medical_cv

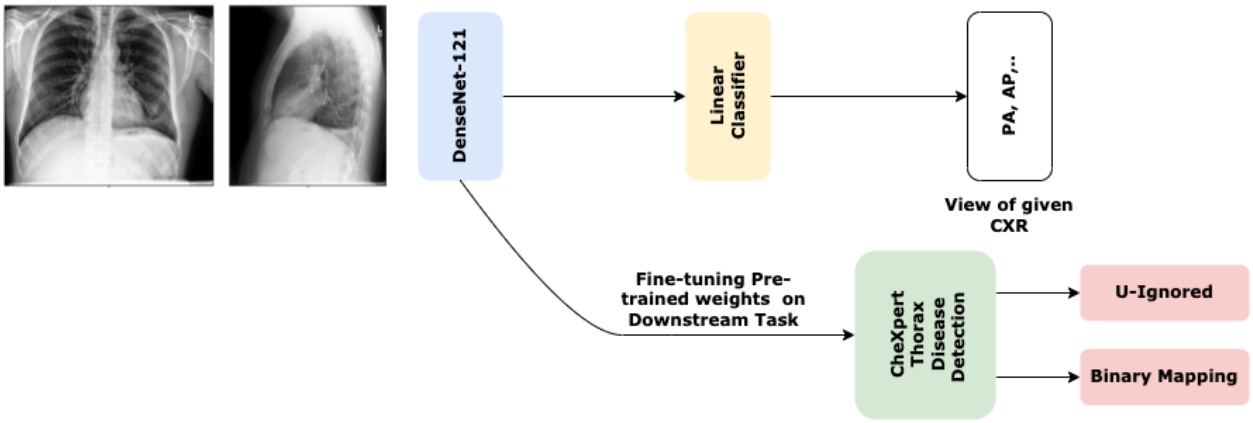

Figure 1: Model architecture. We optionally pretrain the model to predict properties of other chest x-rays before fine-tuning the model to predict pathologies on different images.

datasets like ChestXpert[8], ChestX-ray8[9] and PadChest[]. Convolutional Neural Networks is a common feature in such architectures: Wang et al., 2019, evaluated four classic CNN architectures to classify and localize disease lesion areas in a weakly supervised manner. CheXNet[13] fine-tunes a DenseNet-121 on the global chest X-ray images, with a modified last fully-connected layer.

In CXR analysis, it is common to use multiple views per given patient to improve detection accuracy. Peng et al. 2018[] proposed a two-stage method involving feature extraction, followed by feature fusion based on multiple views to learn multi-view semantic information. Similar approaches have been extended to other tasks such as Breast Cancer Screening using CNN features from multiple views to produce cumulative predictions using mechanisms like Attention[11]. Furthermore, recent self-supervised learning approaches like Multi-Instance Contrastive Learning[12] exploited this characteristic by maximizing agreement in representations of multiple views per medical condition.

3. Dataset Description

CheXpert is a large dataset of chest X-rays for automated chest x-ray interpretation, which features uncertainty labels and radiologist-labeled reference standard evaluation sets. It consists of 224,316 chest radiographs of 65,240 patients, where the chest radiographic examinations and the associated radiology reports were retrospectively collected from Stanford Hospital. Each report was labeled for the presence of 14 observations as positive, negative, or uncertain.

ChestX-ray8 is an open-source medical database from the Department of Radiology and Imaging Sciences, National Library of Medicine, and National Institutes of Health, Bethesda A ELIMINER consisting of 108,948 frontal views of CXR images of 32,717 unique patients, comprising classified images of 8 popular diseases.

4. Approach

We first attempt to the results of the CheXpert [8] paper. We simplify the problem slightly by sticking to the lower-resolution version of the dataset, using simple uncertainty handling techniques, and treating different viewing angles as separate examples instead of taking the max

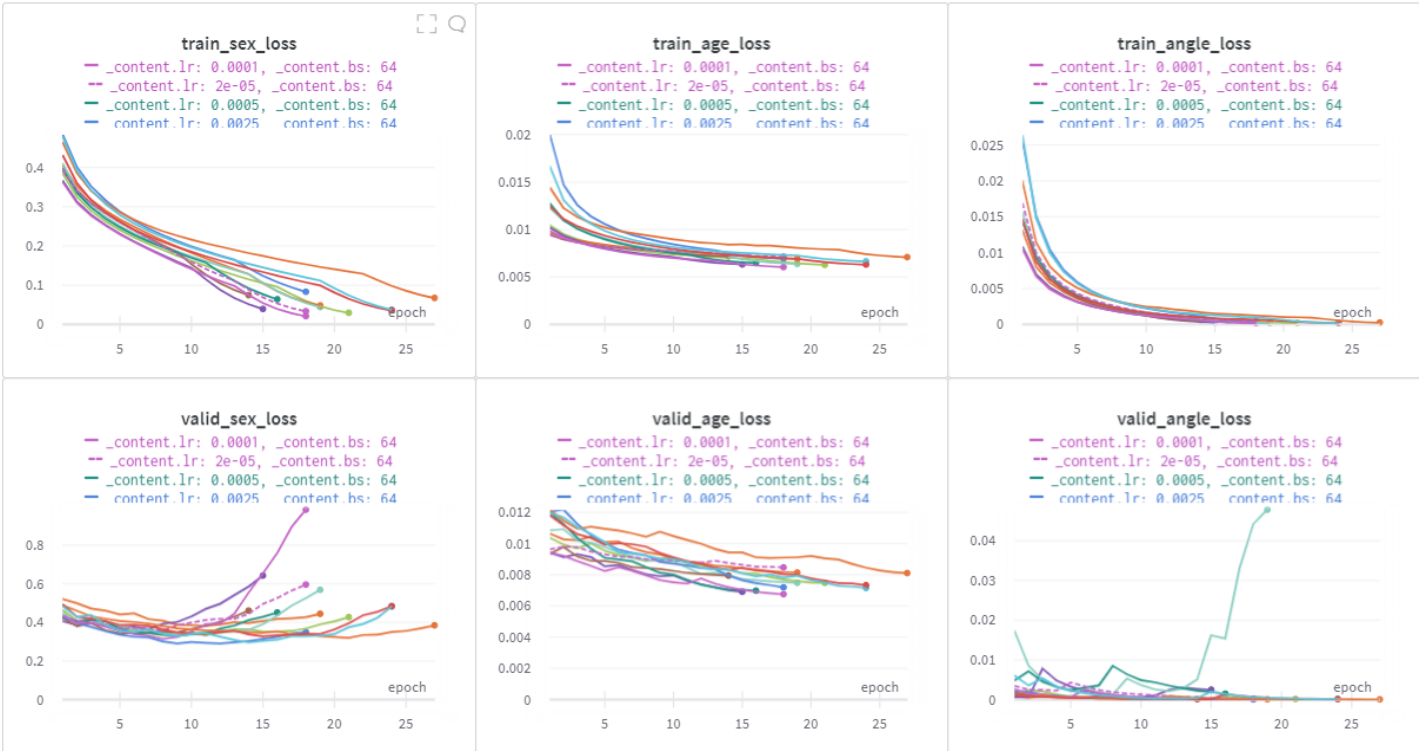

Figure 2: auxiliary task training and validation losses. Each curve represents a different combination of learning rates and batch sizes. We are glad to see the training losses going down, but the increase in validation loss over time clearly indicates overfitting.

prediction across both angles when available.

For uncertainty mapping specifically, we try both the “ignore” method (throw out uncertain labels and only learn to predict presence or absence of a condition) as well as the binary mapping technique of uncertainty handling as described in the paper by replacing all uncertain labels with a positive label.

We experiment with both ResNets and DenseNets, including ResNet50, ResNet152, and DenseNet121. The authors of CheXpert reported their best performance with DenseNet121.

4.1 Predict the easy stuff

First, we try training models to predict “easy” parts of the dataset: predicting the sex and age of the patient as well as predicting the viewing angle of the xray itself. This served to validate our experimental setup.

4.2 Predict pathologies

Next, we predict the actual pathologies present in the dataset and attempt to reach the numbers reported in the paper. We evaluate several attempts for improving this performance, including techniques such as weight decay as well as auxiliary losses.

4.3 Evaluate pre-training methods

Finally, we test several models that have either been randomly initialized, pretrained on imagenet, or pretrained on an alternative chest x-ray dataset called ChestX-ray8.The the dataset comprises 108,948 frontal view X-ray images of 32,717 patients. To account for the multi-view nature of medical images, we follow a predictive pre-training task. We train randomly initialized models on Chest-Xray8 to predict the view of a given CXR and then evaluate the learned representations on CheXpert.

Figure 3: auxiliary task training and validation losses. Weight decay mitigates overfitting when it is strong enough: 1e-6 doesn't have an effect, but 1e-4 works well.

To explore other pre-training alternatives, we also train a DenseNet-121 using the Redundancy Reduction Objective: Barlow Twins. We pre-train for 50 epochs.

For pretraining we also employ common augmentations: Center Crop, Horizontal flip, Color Jitter and Rotation for both pre-training methods as well as an additional Multi-Instance Augmentation [12] for the self-supervised approach.

5. Experiments & Analysis

Here we discuss the results of our approach in detail.

5.1 Predicting Sex, Age, Angle

See Figure 2. We predict meta-data about the patients for each image in the dataset: the sex of the patient and the angle of the xray (front or lateral) using a binary classification loss and the age using a smoothed L1 loss. We out a variety of batch sizes and learning rates and find that we get the best performance with the same learning rate as the CheXpert authors but move forward with double the batch size.

We find here that overfitting is clearly a problem and employ weight decay to good effect: see Figure 3.

5.2 Predicting pathologies

In Figure 4, we show our results predicting pathologies. Here we again observe

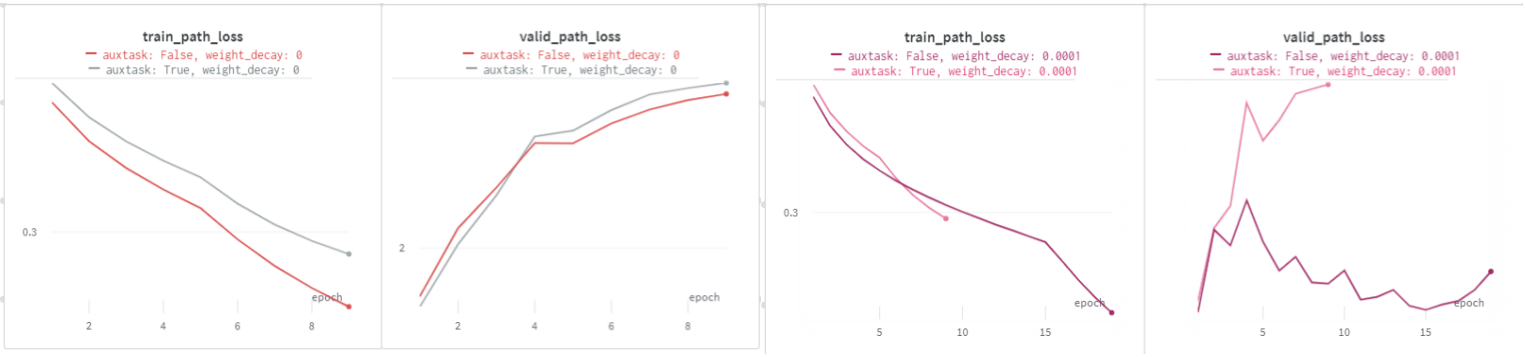

Figure 4: on the left we see the effect of auxiliary losses without weight_decay, i.e. negligible leaning bad (both overfit). On the right we see an even more dramatic effect: with weight_decay enabled, overfitting is reduced only when the auxiliary losses are also disabled. We conclude that the auxiliary losses actually increase overfitting on the training dataset.

overfitting to be a problem which is mitigated by a strong weight decay. We try using the sex, age, and angle losses as an auxiliary task to potentially reduce overfitting, but this actually has the opposite effect: it actually increases overfitting to the point of cancelling out the weight decay. This surprised usas we thought the auxiliary task may actually help to generalize the parameters and prevent overfitting on the target task of pathology prediction.

Next in Figure 5, we show a comparison between ResNet152 and DenseNet121. We find, as did the CheXpert authors, a slight preference for DenseNet.

5.3 Predicting pathologies - binary mapping of uncertainty labels

Here, we follow the same implementation as above, with the difference being in uncertainty label handling. The uncertain labels, instead of being 'ignored' are binary mapped. We used U-Ones technique for this mapping, that is the uncertain labels are mapped to Ones.

5.4 Evaluating pretraining methods

We pretrain a ResNet50 model as well as theDenseNet121 model with two different training setups on the ChestX-ray8 dataset, then use these weights to initialize a model with new output heads which is then fine-tuned on CheXpert. We also try initializing the models with weights based on ImageNet pretraining. Note that the DenseNet121 model is about one-third of the size of ResNet50 in terms of number of parameters.

We find that pretraining on ImageNet helped the smaller DenseNet121, performing better

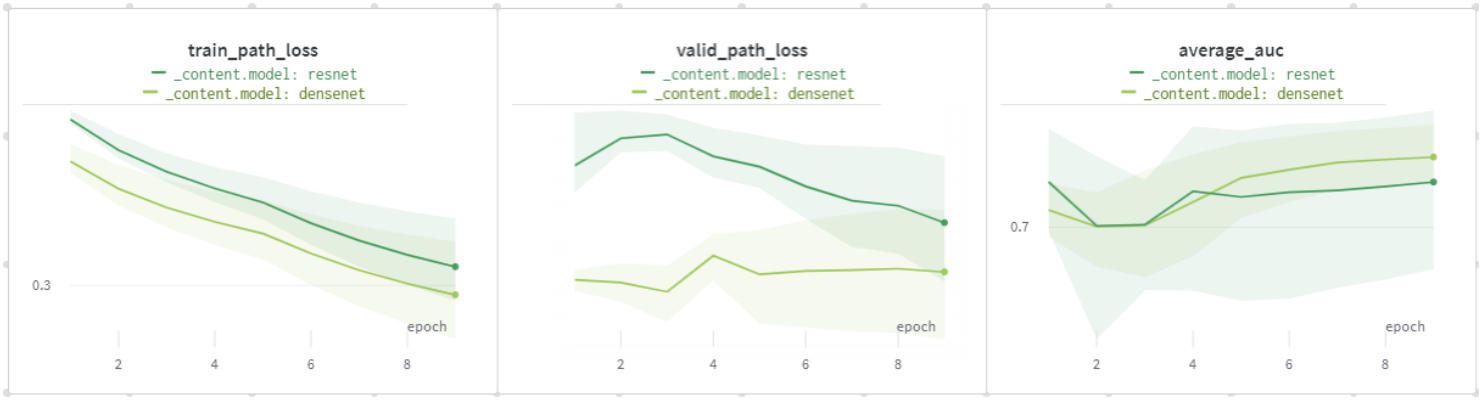

Figure 5: we find that DenseNet121 slightly outperforms ResNet152.

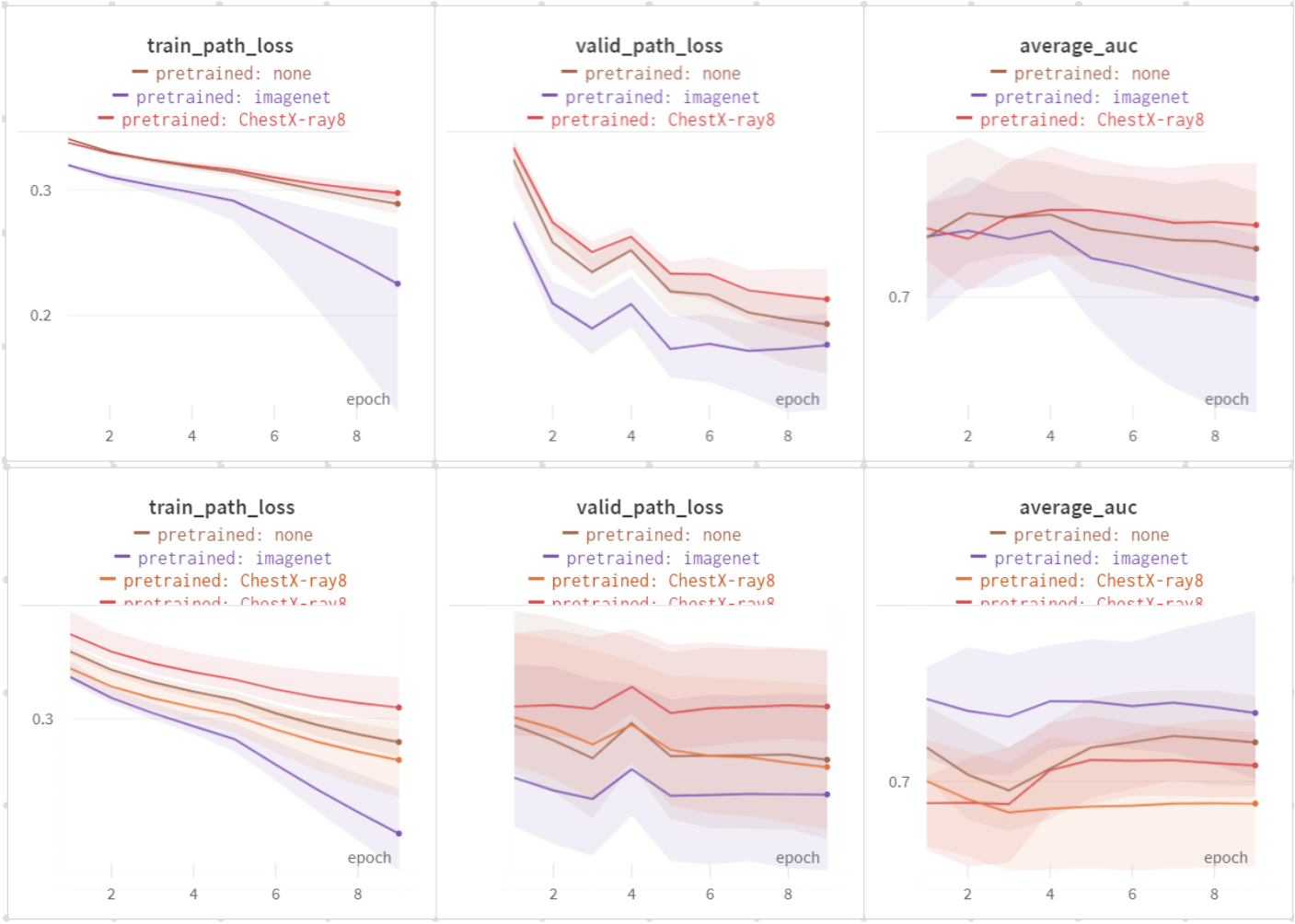

Figure 6: on the top we see that pretraining for ResNet50 on another chest x-ray dataset improves results slightly on the validation set, while imagenet pretraining slightly hurts results. We see on the lower plots the opposite effect for DenseNet121, where performance is slightly hurt by the chest x-ray pretraining and slightly improved by imagenet pretraining. On the lower plots, the two chest x-ray plots represent two slightly different training setups.

than random initialization. However, pre-training on the chest x-ray data hurt the model, performing worse than both ImageNet pretraining and random initialization.

We observe the opposite effect for the larger ResNet: pretraining on ImageNet is worse than random initialization, while pretraining on chest x-rays outperforms both random and ImageNet initialization.

Note that the effect of each of these different settings is relatively small. See figure 6 for more detail.

6. Discussion & Future Work

From our experiments, we find mixed results when applying in-domain pre-training in comparison to Imagenet pre-trained models. [14] also found that imagenet pretraining outperformed in-domain pretraining and stated that a definitive answer to such a hypothesis is difficult to conclude as most of the medical datasets

are generated from the institute's records which may not be standard to all. [15] and [16] highlighted that pre-training on natural images was beneficial to the given medical downstream task.

For the CheXpert dataset, [13] also concluded that ImageNet pretraining yields a statistically significant boost in performance for chest x-ray classification. He et al. [11] found models without pretraining had comparable performance to models pre-trained on ImageNet for object detection and image segmentation of natural images—to the medical imaging setting. Therefore, we need to enhance our in-domain pre-training objective for better representations pertaining to clinical data.

Apart from pre-training, other factors that have contributed to better performance on CheXpert is using large resolution images [8] and Deep AUC Maximization (DAM) [17]. Employing DAM's margin-based min-max surrogate loss and uncertainty mapping, we get comparable AUC scores to [8]. This suggests that for medical images, factors like resolution, uncertainty resolution and metric suited loss functions might be significant to boost performance.

Future directions include enhancing pre-training objectives by incorporating large resolution images and regularized pre-training objectives like [19] that can avoid non-informative representations.

A. References

1. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. (2014)

2. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: DeCAF: A Deep Convolutional

Activation Feature for Generic Visual Recognition.
arXiv:1310.1531 (2013)

3. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional neural networks. In: ECCV. (2014)

4. Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., Van Der Maaten, L.: Exploring the limits of weakly supervised pretraining. In: ECC. (2018)

5. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
arXiv:1810.04805 (2018)

6. Lewis, M., Liu, Y., Goyal, N, Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. arXiv:1910.13461 (2019)

7. Brown, B. et al: Language Models are Few-Shot Learners. arXiv:2005.14165 (2020)

8. Irvin, J. et al: CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison. In: AAAI. (2019)

9. Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, Ronald Summers, ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised

Classification and Localization of Common Thorax Diseases, IEEE CVPR, pp. 3462-3471, 2017

10. Bustos, Aurelia, A. Pertusa, Josee María Salinas and María de la Iglesia-Vayá. “PadChest: A large chest x-ray image dataset with multi-label annotated reports.” Medical image analysis 66 (2020): 101797.

11. Shen, Y., Wu, N., Phang, J., Park, J., Kim, G., Moy, L., Cho, K., & Geras, K. J. (2019). Globally-Aware Multiple Instance Classifier for Breast Cancer Screening. Machine learning in medical imaging. MLMI (Workshop), 11861, 18–26. https://doi.org/10.1007/978-3-030-32692-0_3

12. Azizi, Shekoofeh, Basil Mustafa, Fiona Ryan, Zach Beaver, Jana von Freyberg, Jonathan Deaton, Aaron Loh, Alan Karthikesalingam, Simon Kornblith, Ting Chen, Vivek Natarajan and Mohammad Norouzi. “Big Self-Supervised Models Advance Medical Image Classification.” arXiv: abs/2101.05224 (2021)

13. Rajpurkar, P., Irvin, J.A., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D.Y., Bagul, A., Langlotz, C., Shpanskaya, K.S., Lungren, M.P., & Ng, A. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. arXiv: abs/1711.05225.

14. Cheplygina, Veronika , “Cats or CAT scans: transfer learning from natural or medical image source datasets?,” arXiv:1810.05444 [cs.CV], Jan. 2019.

15. T. Schlegl, J. Ofner, and G. Langs. Unsupervised pre-training across image domains improves lung tissue classification. In Medical Computer Vision:
Algorithms for Big Data (MICCAI MCV), pages 82–93. Springer, 2014.

16. A. Menegola, M. Fornaciali, R. Pires, F. V. Bittencourt, S. Avila, and
E. Valle. Knowledge transfer for melanoma screening with deep learning.
In International Symposium on Biomedical Imaging (ISBI), pages 297–300.
IEEE, 2017.

17. Yuan, Z., Yan, Y., Sonka, M., & Yang, T. (2021). Large-scale Robust Deep AUC Maximization: A New Surrogate Loss and Empirical Studies on Medical Image Classification. Proceedings of the IEEE/CVF International Conference on Computer Vision.

18. Bardes, A., Ponce, J., & LeCun, Y. (2021).VICReg:Variance-Invariance-Covariance Regularization for Self-Supervised Learning. arXiv: abs/2105.04906.