Deep metric learning for few-shot X-ray image classification

Jakub Prokop[1,2 * ¶], Javier Montalt Tordera[1 ¶], Joanna Jaworek-Korjakowska[2], and Sadegh Mohammadi[1]

[1]Bayer AG, Germany
[2]AGH University of Krakow, Poland

[*]Corresponding author
E-mail: jakub.prokop@bayer.com (JP)

[¶] These authors contributed equally to this work

Abstract

Deep learning models have proven the potential to aid professionals with medical image analysis, including many image classification tasks. However, the scarcity of data in medical imaging poses a significant challenge, as the limited availability of diverse and comprehensive datasets hinders the development and evaluation of accurate and robust imaging algorithms and models. Few-shot learning approaches have emerged as a potential solution to address this issue. In this research, we propose to deploy the Generalized Metric Learning Model for Few-Shot X-ray Image Classification. The model comprises a feature extractor to embed images into a lower-dimensional space and a distance-based classifier for label assignment based on the relative distance of these embeddings. We extensively evaluate the model using various pre-trained convolutional neural networks (CNNs) and vision transformers (ViTs) as feature extractors. We also assess the performance of the commonly used distance-based classifiers in several few-shot settings. Finally, we analyze the potential to adapt the feature encoders to the medical domain with both supervised and self-supervised frameworks. Our model achieves 0.689 AUROC in 2-way 5-shot COVID-19 recognition task when combined with REMEDIS (Robust and Efficient Medical Imaging with Self-supervision) domain-adapted model as feature extractor, and 0.802 AUROC in 2-way 5-shot tuberculosis recognition task with domain-adapted DenseNet-121 model. Moreover, the simplicity and flexibility of our approach allows for easy improvement in the feature, either by incorporating other few-shot methods or new, powerful architectures into the pipeline.

Keywords— few-shot learning, metric learning, domain adaptation, feature extraction, radiology, medical image classification

1 Introduction

Deep Learning (DL) has shown immense potential in revolutionizing medical image analysis. With access to sufficient data, DL models can achieve human-level performance in a wide range of tasks – from accurate diagnostics comparable to physicians to medical scene perception [1]. However, the main drawback of traditional DL models lies in their heavy reliance on extensive labeled data for effective training on specific tasks. Acquiring and annotating such large datasets can be expensive, especially in the medical domain.

In recent years, few-shot learning (FSL) has emerged as a promising solution to address the limitations of traditional DL models.FSL represents a diverse set of technologies aimed at enabling models to learn and generalize from a limited amount of labeled data, even with novel and unseen tasks. Implementing these technologies holds promising prospects for substantially decreasing the investment necessary for developing novel DL applications through the mitigation of data collection and annotation requirements and the reduction of computational resource demands. Furthermore, FSL techniques can facilitate DL in domains where substantial data availability is lacking, broadening the applicability of DL to various fields.

Radiology is one area with vast potential to benefit from FSL techniques. A frequent use of DL applications in radiology is image classification, closely related to the common radiological interpretative task of providing a diagnosis. Examples include tuberculosis recognition [2], mammographic tumor classification [3], and bone age assessment [4], among many others [5]. These applications can assist radiologists by providing a ”second opinion,” speeding up triage, reducing miss rates, or allowing them to divert attention to more complex cases or tasks. When confronted with the classification of rare or emerging diseases, such as COVID-19 diagnosis [6, 7, 8] and diverse lung pathologies recognition [9, 10], the utilization of the FSL techniques proves invaluable.

Among the FSL approaches, metric learning has emerged as a simple yet highly effective method. Although the idea is long-established, recent results suggest that the metric-based approach remains one of the most powerful FSL methods, outperforming other more sophisticated state-of-the-art algorithms [11]. Metric learning (embedding learning) measures samples' similarity with defined metrics. It embeds the data samples into a lower-dimensional latent space to reduce the spacing between similar samples and increase the distance between dissimilar ones. Various metric learning approaches have been proposed in the scientific literature to address the problem at hand. Examples of such approaches include *Matching Nets* [12] and its modifications [13, 14, 15, 16], which aim to learn embedding functions for data samples. Additionally, *Prototypical Networks* [17, 18, 19, 20] have been introduced, leveraging the concept of developing class prototypes to enhance metric learning capabilities. Metric learning methods were successfully applied in radiology in tasks such as brain tumor classification [21] and recognition of various chest X-ray pathologies [22].

The recent advance of self-supervised learning [23] further boost the potential of metric learning in medical imaging. Self-supervision can be viewed form of unsupervised learning solving typically supervised tasks. A self-supervised model is either learning to recover missing data parts, e.g. by predicting the masked fragments of an image (generative approach), or learning to predict the similarity of two fragments of the same data sample, e.g. two augmented versions of one image (contrastive approach) [24]. As the self-supervision does not require labelled data, it correlates particularly well with the few-shot paradigm, and perfectly fits the realities of low availability of labelled medical data. Meanwhile, self-supervised models has already proved to be highly effective in many medical tasks [25, 26]. With the flexibility of metric learning approach in terms of model architecture, highly efficient self-supervised models can be easily incorporated into few-shot metric-based pipeline.

A systematic evaluation of different few-shot metric learning models, both supervised and self-supervised, still needs improvement. Metric-based approaches, such as *Prototypical Networks*, allow many variations of model architecture, training strategy and classifier. Yet, to the best of our knowledge, there is no such comparison available for medical imaging. Therefore, we evaluated the method described in [17] on various X-ray classification tasks, implementing several base model variants using

more modern architectures and pre-training frameworks.

The main contributions of this paper are as follows:

- We analysed the effectiveness of the few-shot metric learning approach inspired by ProtoNet [17] in radiology, namely in COVID-19 and tuberculosis X-ray classification tasks.
- We benchmarked three convolutional neural network (CNN) and three vision transformer (ViT) architectures as medical feature extractors, as well as three commonly used distance-based classifiers in several few-shot settings, under different data imbalance conditions.
- We compared 6 architectures trained on natural images with 5 domain-adapted ones, including CNN models from TorchXrayVision [27] and RadImageNet [28], as well as with the self-supervised in-domain REMEDIS [29] model based on ResNet-50.
- We fine-tuned DINO-ViT, DINO-ResNet-50 [30] and ViT-MAE models [31] within their respective self-supervised frameworks, using CheXpert [32] X-ray image dataset. We evaluated the effectiveness of this domain adaptation on 11 in-distribution (ID) and 9 out-of-distribution (OOD) disease classification tasks.
- In COVID-19 and tuberculosis classification tasks our model achieved 0.689 and 0.802 AUROC in 2-way 5-shot setting, as well as 0.782 and 0.903 AUROC in 2-way 50-shot scenario.

2 Methods

A generalized metric learning model consists of a feature extractor, to embed an image into a lower-dimensional space, and a distance-based classifier, to assign labels to the test samples basing on the relative distance of these embeddings. We followed this schema, and implemented our model as shown in Fig **??**. We incorporated several neural network architectures as task-agnostic feature extractors, and described them in detail in Section 2.3. Our selection of classifiers is described in Section 2.4. To evaluate our model in varied few-shot settings, we generated a number of episodes, in which a small, task-specific support set was sampled to fit the classifier (see Section 2.2 for details). Then, the evaluation was performed on the test sets referring to different target tasks, which are given a comprehensive overview in Section 2.6 and Section 2.5.

2.1 Metric learning

The goal of metric learning is to embed each sample $\boldsymbol{x_i} \in X \in \mathbb{R}^n$ to a lower-dimensional $\boldsymbol{z_i} \in Z \in \mathbb{R}^m$, $m < n$, such that similar samples are close

together and can be easily grouped by a distance-based classifier. In the simplest case, given image-label pairs $(\boldsymbol{x_1}, y_1)$, $(\boldsymbol{x_2}, y_2)$, unlabelled image $\hat{\boldsymbol{x}}$ and distance function d, the metric learning model would compute $\boldsymbol{z_1}$, $\boldsymbol{z_2}$, $\hat{\boldsymbol{z}}$, and assign the label y_1 to $\hat{\boldsymbol{x}}$ if $d(\boldsymbol{z_1}, \hat{\boldsymbol{z}}) < d(\boldsymbol{z_2}, \hat{\boldsymbol{z}})$, and y_2 to $\hat{\boldsymbol{x}}$ otherwise.

For the purpose of encoding x_i into z_i we utilized a neural network feature extractor, which is trained using task-agnostic supervised or self-supervised learning. To find a correct label y_i, instead of directly measuring the distance d to closest labelled sample as described above, we used a distance-based classifier c such as k-NN (see Section 2.4). To fit the classifier, we followed [12] and defined a small, task-specific *support set* $S = \{(\boldsymbol{x_i}, y_i)\}_{i=1}^{M}$, where M refers to its size. Then, given a task-specific classifier c_S and a sample from a test set $\hat{\boldsymbol{x}}$, we embedded every $\boldsymbol{x_i}$ and $\hat{\boldsymbol{x}}$ into $\boldsymbol{z_i}$, $\hat{\boldsymbol{z}}$ and predicted $\hat{y} = c_S(\hat{\boldsymbol{z}}, d)$. In our work the distance function d is defined as the Euclidean distance, as done in [17].

2.2 Few-shot scenarios

The few-shot classification problem is usually referred to as k-shot N-way classification task [12], where k denotes the number of labeled samples for each category in the training set, and N refers to the overall number of classes. We implemented four different few-shot settings with $k \in \{5, 10, 25, 50\}$.

Our model was assessed using various unseen target tasks, which are described in Section 2.6. Several variations of the model were built with a selection of feature extractors and classification heads. Therefore, we made an evaluation with every combination of k-shot setting, target task, feature extractor, and classifier. For each of these combinations, 200 few-shot scenarios (episodes) $(s_1, s_2, ..., s_{200})$ were randomly generated to reduce random effects and assess the statistical significance of obtained results. For every scenario s_i, a vector $\boldsymbol{k_i} = (k_{i,1}, k_{i,2}, ..., k_{i,N}) \in \mathbb{N}^N$ was randomly selected, with the value $k_{i,j}$ denoting number of examples for class N_j in the support set S_i, under the following constraints:

1. The number of examples for any class in the support set was at least 20% of k:

$$k_{i,j} \in [0.2k; 1.8k] \tag{1}$$

2. The total size of the support set was equal to k times N:

$$\sum_{j=1}^{N} k_{i,j} = M = kN \tag{2}$$

For example, for a 50-shot *2*-way task, there were at least 10 examples for every class and 100 examples in total. This sampling strategy allowed us to study classifier

performance under different imbalance ratios. In our experiments we considered only binary classification tasks, giving $N = 2$ and $\boldsymbol{k_i} = (k_{i,1}, k_{i,2})$.

2.3 Feature extractors

2.3.1 Off-the-shelf models

Our study evaluated the performance of widely used and publicly available neural networks employed as feature extractors. Since these networks were initially pre-trained on natural images, we designated them as "general-purpose" feature extractors. The analysis included the following models:

- ResNet-50. ResNet [33] is a deep CNN that uses residual connections to alleviate the vanishing gradient problem. Its 50-layer variant is the most used by far and was also used in this work to facilitate comparison with related literature.
- DenseNet-121. DenseNet [34] is a CNN that incorporates feed-forward connections from each layer to every other layer, improving feature propagation and parameter efficiency. Its 121-layer variant was used in this work as it has become particularly popular within the research community.
- ConvNeXT-XL. ConvNeXT [35] is a more modern convolutional architecture, in which the authors systematically studied many design decisions, inspired by recent advances in vision transformers. The extra-large (XL) variant was used in this work because it offers the best performance on ImageNet.
- DINO-ViT-B/8. DINO-ViT [30] is a ViT trained with a self-supervised method called DINO (self-distillation with no labels). Apart of excellent classification performance, the model performs particularly well when combined with a basic nearest neighbors classifier (k-NN), which resonates with our own selection of classifier heads. The B/8 variant (patch size 8) was used due to its classification accuracy on ImageNet.
- DINOv2-ViT-B/14. DINOv2 [36] is the second release of the DINO framework, which produced models with even higher ability to extract high-performance visual features. The B/14 distilled variant was used.
- ViT-MAE-B. ViT-MAE [31] is a ViT trained using masked autoencoding (MAE), a simple self-supervision method that involves masking and reconstructing a large proportion of the image. The base variant with patch size 16 was used in this work.

For CNNs, the feature vector passed as input to the classification head was the output of the global average pooling layer. For ViTs, the class token was passed.

Besides the general feature extractors, we added five ”domain-specific” models to the pool. ResNet-50 and DenseNet-121 have publicly available weights derived from medical datasets published by RadImageNet (RIN) [28] and TorchXRayVision (XRV) [27]. RIN weights are derived from several medical imaging modalities but do not include X-rays. XRV weights are derived primarily from X-ray images. In total, four models were included: ResNet-50-RIN, ResNet-50-XRV, DenseNet-121-RIN and DenseNet-121-XRV.

Lastly, we incorporate REMEDIS-CXR-50-M model. The ”CXR-50” variant of REMEDIS (”Robust and Efficient Medical Imaging with Self-supervision”) [29] is a ResNet-50 architecture initialized with BiT-M [37] weights from natural domain. It was trained within SimCLR [38] contrastive self-supervised framework using CheXpert [32] and MIMIC-IV-CXR [39] large-scale chest X-ray datasets.

2.3.2 Self-supervised domain adaptation

Several sources ([40, 41, 42, 29, 43]) suggest that self-supervised medical domain-adaptation may result in better performance than the supervised approach. Zhou et al.[42] verifies this hypothesis for the ViT-MAE model, while Matsoukas et al. [41] do the same for DINO. In both cases, the obtained improvements seem promising, though in some cases, marginal. Finally, Azizi et al. [29] report outstanding performance of self-supervised domain-adapted models across many different medical image classification tasks, which further boosts our motivation.

We fine-tuned DINO-ViT-B/8 and ViT-MAE-B within their respective self-supervised frameworks to evaluate this approach. We did the same with ResNet-50 using DINO, as this architecture is often used as a backbone for self-supervised models [44, 30, 38, 29]. The pre-trained models and training code are publicly available for both frameworks[1][2]. For training, we utilized the CheXpert dataset [32], containing 224,316 chest radiographs of 65,240 patients, from which 223,648 images were used as our training set (we followed the default split). In both cases (DINO and MAE), we trained the model for 50 epochs, with five warm-up epochs, on 8 A100 40GB GPU units. The batch size was set to 8 per processing unit, and the rest of the hyperparameters were left at their default values.

2.4 Classifiers

The following classification algorithms were included in the analysis:

- k-nearest neighbors (k-NN) assigns each observation to the class most common among its k nearest neighbors. k-NN is the most widespread non-parametric

[1] https://github.com/facebookresearch/dino
[2] https://github.com/facebookresearch/mae

classifier. The number of neighbors participating in the vote was set to the expected number of examples per class [45]. Votes were weighted by the inverse of the distance to the observation, as done in [30].

- Nearest centroid (NC) assigns each observation to the class of the training samples whose mean (centroid) is closest to the observation. NC is a simple classifier that has been successful in few-shot settings [17].

- Neighborhood Components Analysis (NCA) learns an optimal distance metric for the nearest neighbors (k-NN) classifier.

All distances were measured using the Euclidean distance, as done in [17] and [45]. The performance of each classifier was evaluated according to the area under the receiver-operating characteristic curve (AUROC) and balanced accuracy – the arithmetic mean of sensitivity and specificity.

2.5 Datasets

We used four chest X-ray datasets in our work. We show samples with exemplary metadata from these datasets on Fig **??**. The COVID-19 image data collection project [46] collects images from several sources. The dataset has independent labels for several conditions and is suitable for multi-label classification. To prepare the dataset for binary classification, the label vectors containing the COVID-19 label were set to positive, and those without it were set to negative, resulting in 584 positive and 1841 negative cases. The dataset was split in proportions 80%/20% for train and test subsets.

Montgomery and Shenzhen datasets were obtained from medical centers in Montgomery County, MD, USA, and Shenzhen, China, and released by the US National Library of Medicine [47]. We combine them and obtain a dataset suitable for binary classification, with images labeled normal (healthy) or abnormal (tuberculosis). There are 406 cases with "healthy" labels and 394 cases with "tuberculosis detected" labels. Again, we randomly split the set to obtain the train/test set in proportions 80%/20%

The CheXpert dataset [32] is a large-scale chest X-ray dataset containing 224,316 images from 65,240 patients. We selected 11 pathologies with the largest number of positive cases available, resulting in 223,413 images in the training set. To every one of these cases, we assigned a positive label if the pathology was detected and a negative label otherwise. The corresponding test set is much smaller, with the exact numbers shown in Table 1.

This NIH Chest X-ray dataset [48] contains 112,120 chest X-ray images from 30,805 patients. We sampled nine pathologies with the largest number of positive cases available, resulting in 86,524 training images. Every case with confirmed pathology

Table 1: Number of examples available for selected pathologies in the test split of CheXpert dataset.

Pathology	# positive cases	# negative cases
Enlarged Cardiomediastinum	298	370
Cardiomegaly	175	493
Lung Opacity	310	358
Lung Lesion	14	654
Edema	85	583
Consolidation	35	633
Pneumonia	14	654
Atelectasis	178	490
Pneumothorax	10	658
Pleural Effusion	120	548
Fracture	6	662

was assigned a positive label, and a negative label was set for all negative or uncertain cases. We sampled the same pathologies from the test set, resulting in numbers in Table 2.

2.6 Classification tasks for the generalized metric learning model

We chose two binary classification target tasks to evaluate our baseline selection of off-the-shelf feature encoders and adapted classifiers: COVID-19 and tuberculosis diagnosis. COVID-19 recognition task was composed from COVID-19 image data collection. Data for the tuberculosis recognition task was provided by Montgomery and Shenzhen datasets. In both cases the support set was randomly sampled from the train split, while the whole test set was used for evaluation.

For the evaluation of the model with domain-adapted feature extractors, we used a much larger set of target tasks to improve our ability to detect small effects of self-supervised fine-tuning. At first, we utilized binary classification tasks drawn from the CheXpert train dataset to see if the fine-tuning resulted in any few-shot classification improvement within the source dataset. We follow [29] and call this setting *in-distribution* (ID) evaluation. Next, we composed the *out-of-distribution* (OOD) set of binary classification tasks from NIH Chest X-ray dataset. This allowed us to assess the model generalization ability and transferability of knowledge learned on ID dataset.

In both cases (ID and OOD evaluations) we measured a relative mean change of AUROC achieved with feature extractors before and after domain adaptation. The statistical significance of the results was assessed using a paired two-tailed t-test with

Table 2: Number of examples available for selected pathologies in the test split of NIH Chest X-ray dataset.

Pathology	# positive cases	# negative cases
Atelectasis	3219	33788
Cardiomegaly	1069	35998
Consolidation	1825	35252
Effusion	4658	32409
Infiltration	6112	30955
Mass	1748	35319
Nodule	1623	35444
Pleural Thickening	1143	35924
Pneumothorax	2665	34402

the null hypothesis that domain adaptation has no impact on performance. p-values less than 0.05 were considered statistically significant.

3 Results

The results of our experiments include the performance assessment of several variants of the model, in different few-shot settings. We present the comparison of model efficacy with the incorporation of general and domain-specific feature extractors, including our adapted versions of DINO-ViT-B/8, ViT-MAE-B and ResNet-50. We examine the performance of three described classifiers with the respect to the support data imbalance ratio. Lastly, we give a detailed analysis of the effectiveness of our attempt of self-supervised domain adaptation of DINO-ViT-B/8, ViT-MAE-B and ResNet-50 models.

3.1 General feature extractors

The results of the evaluation of general feature extractors for COVID-19 recognition are described in Table 3. ViTs outperformed CNNs in almost every scenario, with DINO-ViT-B/8 being the most effective. DINOv2 fell behind DINO-ViT and did not outperform ViT-MAE in most cases. Of the CNNs, ResNet-50 proved to be the most reliable, with ConvNeXt falling behind slightly and DenseNet-121 trailing them both significantly. Table 4 shows a similar comparison for the tuberculosis classification, but this time the results are very similar to the ones observed for COVID-19 task.

Table 3: Evaluation of general feature extractors for COVID-19 recognition task. Top 3 results measured by mean AUROC for each number of examples per class are marked in bold.

Encoder	Classifier	Examples per class			
		5	10	25	50
ResNet-50	k-NN	0.578	0.615	0.654	0.681
	NC	0.586	0.607	0.652	0.681
	NCA+k-NN	0.565	0.606	0.652	0.686
DenseNet-121	k-NN	0.519	0.528	0.545	0.561
	NC	0.525	0.533	0.556	0.568
	NCA+k-NN	0.511	0.527	0.540	0.555
ConvNeXT-XL	k-NN	0.566	0.587	0.612	0.627
	NC	0.579	0.609	0.645	0.667
	NCA+k-NN	0.565	0.588	0.631	0.656
DINO-ViT-B/8	k-NN	0.591	0.617	0.667	0.706
	NC	**0.614**	**0.658**	**0.715**	**0.758**
	NCA+k-NN	0.592	0.618	0.669	0.706
DINOv2-ViT-B/14	k-NN	0.581	0.610	0.665	0.693
	NC	**0.594**	0.613	**0.684**	0.709
	NCA+k-NN	0.585	0.602	0.663	0.695
ViT-MAE-B	k-NN	0.591	**0.623**	0.670	**0.712**
	NC	**0.599**	**0.656**	**0.700**	0.737
	NCA+k-NN	0.581	0.603	0.671	0.711

Table 4: Evaluation of general feature extractors for tuberculosis recognition task. Top 3 results measured by mean AUROC for each number of examples per class are marked in bold.

Encoder	Classifier	Examples per class			
		5	10	25	50
ResNet-50	k-NN	0.630	0.672	0.719	0.752
	NC	0.638	0.685	0.734	0.767
	NCA+k-NN	0.626	0.685	0.748	0.786
DenseNet-121	k-NN	0.575	0.600	0.639	0.657
	NC	0.584	0.612	0.650	0.669
	NCA+k-NN	0.544	0.579	0.623	0.659
ConvNeXT-XL	k-NN	0.639	0.674	0.709	0.730
	NC	0.650	0.695	0.738	0.758
	NCA+k-NN	0.629	0.684	0.734	0.766
DINO-ViT-B/8	k-NN	0.739	0.776	0.807	**0.834**
	NC	**0.757**	**0.789**	**0.825**	**0.841**
	NCA+k-NN	**0.739**	**0.782**	**0.817**	0.831
DINOv2-ViT-B/14	k-NN	0.729	0.778	0.808	0.819
	NC	**0.752**	**0.791**	**0.815**	0.822
	NCA+k-NN	0.733	0.771	0.805	0.823
ViT-MAE-B	k-NN	0.719	0.762	0.797	0.815
	NC	0.718	0.768	0.812	0.827
	NCA+k-NN	0.679	0.751	0.810	**0.842**

3.2 Domain-specific feature extractors

The Table 5 shows the comparison of performance achieved with domain-specific feature extractors. The REMEDIS model stands out in this task, outperforming our adapted DINO-ViT-Xray model by a large margin, with every combination of classifier and few-shot setting. ViT-MAE-Xray model performed similarly to DINO-ViT-Xray. Domain-specific CNNs performed relatively poorly compared to ViTs. Among these architectures, ResNet-50-RIN performed marginally better than others.

Different observations were made in the tuberculosis recognition task (Table 6), with supervised CNNs, ResNet-50-XRV and DenseNet-121-XRV, noting the highest AUROC across all domain-specific feature extractors. CNNs with RIN weights, however, performed worse with NC classifier, and the only improvement over their general versions is noted with k-NN and NCA+k-NN classifiaction heads. Our adapted DINO-ResNet-Xray performs very similar to the ResNet-50-RIN model, however the efficacy rises slightly with the NC classifier. The model also outperforms the adapted ViT-MAE-Xray. REMEDIS extractor performs much better than the RIN version of ResNet, however still stays far behind ResNet-50-XRV. DINO-ViT notes similar, yet slightly better efficacy.

An additional comparison of general and domain-specific versions of DenseNet-121 and ResNet-50 is shown on Fig **??**. The observations further emphasise relatively low performance of the base DenseNet-121, and superiority of REMEDIS in COVID-19 task. It is also clearly seen that XRV models are notably more effective than both RIN and REMEDIS models in tuberculosis classification task.

We measured the relative change of mean AUROC after the domain adaptation of feature extractors. Results of ID evaluation are presented in Table 7. For ViT-MAE-B and DINO-ViT-B/8 we compared our self-supervised domain-adapted models with their base versions. DINO-ResNet-50 was juxtaposed with ResNet-50-XRV instead, to compare the efficacy of supervised and self-supervised domain-adaptation methods.

Fine-tuning ViT-MAE-B improved the model performance (by mean AUROC) in 36% of tasks. Fine-tuning DINO-ViT-B/8 resulted in improvement in 75% of tasks, and fine-tuned DINO-ResNet-50 notes performance better than ResNet-50-XRV in 45% of cases. Averaging AUROC change across all tasks, DiNO-ViT performance increased by 0.0115, ViT-MAE performance decreased by 0.0146 and ResNet performance decreased by 0.0059.

Results of OOD evaluation are presented in Table 8. To sum it up, fine-tuning ViT-MAE-B improved the model performance in 22% of tasks, fine-tuning DINO-ViT-B/8 resulted in improvement in 6% of tasks, and fine-tuned DINO-ResNet-50 notes no performance improvement in any case. On average, DiNO-ViT performance decresed by 0.0195, ViT-MAE performance decreased by 0.0097 and ResNet performance decreased by 0.1195.

Table 5: Evaluation of domain-specific feature extractors for COVID-19 recognition task. Models adapted by ourselves are underlined. Top 3 results measured by mean AUROC for each number of examples per class are marked in bold.

Encoder	Classifier	Examples per class			
		5	10	25	50
ResNet-50-RIN	k-NN	0.577	0.611	0.653	0.687
	NC	0.595	0.633	0.658	0.678
	NCA+k-NN	0.570	0.610	0.651	0.689
ResNet-50-XRV	k-NN	0.566	0.571	0.605	0.650
	NC	0.574	0.588	0.622	0.672
	NCA+k-NN	0.552	0.596	0.650	0.708
DenseNet-121-RIN	k-NN	0.560	0.591	0.628	0.652
	NC	0.564	0.577	0.600	0.620
	NCA+k-NN	0.570	0.598	0.640	0.666
DenseNet-121-XRV	k-NN	0.559	0.589	0.625	0.658
	NC	0.561	0.581	0.608	0.640
	NCA+k-NN	0.550	0.585	0.622	0.669
REMEDIS-CXR-50-M	k-NN	**0.680**	**0.714**	**0.760**	**0.780**
	NC	**0.689**	**0.721**	**0.760**	**0.770**
	NCA+k-NN	**0.679**	**0.709**	**0.757**	**0.782**
DINO-ViT-xray	k-NN	0.602	0.641	0.694	0.728
	NC	0.625	0.659	0.708	0.724
	NCA+k-NN	0.613	0.646	0.699	0.735
ViT-MAE-xray	k-NN	0.582	0.626	0.663	0.704
	NC	0.601	0.628	0.680	0.700
	NCA+k-NN	0.577	0.613	0.659	0.701
DINO-ResNet-xray	k-NN	0.559	0.587	0.608	0.634
	NC	0.570	0.593	0.613	0.622
	NCA+k-NN	0.555	0.577	0.610	0.651

Table 6: Evaluation of domain-specific feature extractors for tuberculosis recognition task. Models adapted by ourselves are underlined. Top 3 results measured by mean AUROC for each number of examples per class are marked in bold.

Encoder	Classifier	Examples per class			
		5	10	25	50
ResNet-50-RIN	k-NN	0.682	0.711	0.751	0.776
	NC	0.659	0.681	0.715	0.740
	NCA+k-NN	0.668	0.720	0.767	0.788
ResNet-50-XRV	k-NN	0.778	0.814	0.847	0.861
	NC	**0.785**	**0.831**	0.870	0.880
	NCA+k-NN	0.768	0.830	**0.886**	**0.903**
DenseNet-121-RIN	k-NN	0.590	0.631	0.682	0.717
	NC	0.575	0.597	0.634	0.653
	NCA+k-NN	0.612	0.647	0.705	0.742
DenseNet-121-XRV	k-NN	**0.784**	**0.832**	**0.872**	0.883
	NC	**0.802**	**0.851**	**0.880**	**0.888**
	NCA+k-NN	0.750	0.813	0.866	**0.885**
REMEDIS-CXR-50-M	k-NN	0.705	0.752	0.810	0.837
	NC	0.687	0.739	0.791	0.815
	NCA+k-NN	0.698	0.753	0.811	0.837
DINO-ViT-xray	k-NN	0.719	0.777	0.816	0.831
	NC	0.711	0.751	0.804	0.834
	NCA+k-NN	0.724	0.770	0.821	0.837
ViT-MAE-xray	k-NN	0.652	0.656	0.721	0.736
	NC	0.650	0.698	0.736	0.747
	NCA+k-NN	0.647	0.687	0.721	0.740
DINO-ResNet-xray	k-NN	0.680	0.724	0.766	0.781
	NC	0.684	0.726	0.758	0.763
	NCA+k-NN	0.681	0.733	0.780	0.802

Table 7: Relative mean AUROC change in ID tasks after domain adaptation of the encoder. In case of ResNet-50, the comparison was made with the relation to XRV model instead. Evaluation was performed on ID tasks sampled from CheXpert dataset. Positive changes are marked green. Significant differences were marked with stars.

Encoder	Recognition task	Examples per class			
		5	10	25	50
ViT-MAE-B	Atelectasis	0.005	0.014*	0.005	0.010
	Cardiomegaly	-0.012	-0.025*	-0.032*	-0.042*
	Consolidation	-0.033*	-0.023*	-0.049*	-0.066*
	Edema	-0.027*	-0.031*	-0.029*	-0.038*
	ECM[a]	0.004	0.014	0.015	0.010
	Fracture	-0.015	-0.031*	-0.064*	-0.064*
	Lung Lesion	0.015	0.011	0.013*	0.010
	Lung Opacity	-0.004	-0.017*	-0.008	-0.001
	Pleural Effusion	-0.026*	-0.028*	-0.031*	-0.041*
	Pneumonia	0.001	0.009	0.003	-0.009
	Pneumothorax	-0.009	-0.009	0.003	-0.024*
	ALL TASKS	**-0.009***	**-0.011***	**-0.002***	**-0.023***
DINO-ViT-B/8	Atelectasis	-0.017	-0.012	-0.048*	-0.048*
	Cardiomegaly	0.027*	0.012	0.005	0.007
	Consolidation	0.017	0.030*	0.035*	0.025*
	Edema	-0.003	0.023*	0.027*	0.032*
	ECM[a]	0.008	0.015	0.008	0.010
	Fracture	0.008	0.010	0.007	0.013
	Lung Lesion	0.014	0.028*	0.031*	0.045*
	Lung Opacity	0.020	0.011	0.026*	0.035*
	Pleural Effusion	-0.006	-0.011	-0.006	-0.026*
	Pneumonia	0.023*	0.019*	0.040*	0.040*
	Pneumothorax	-0.002	0.017*	0.015	-0.002
	ALL TASKS	**0.008***	**0.013***	**0.013***	**0.012***
DINO-ResNet-50	Atelectasis	0.044*	0.058*	0.103*	0.120*
	Cardiomegaly	0.017	0.009	0.028*	0.032*
	Consolidation	-0.037*	-0.056*	-0.071*	-0.081*
	Edema	0.114*	0.108*	0.105*	0.081*
	ECM[a]	-0.019	-0.023	-0.055*	-0.053*
	Fracture	-0.042*	-0.067*	-0.101*	-0.127*
	Lung Lesion	0.071*	0.090*	0.096*	0.124*
	Lung Opacity	0.063*	0.049*	0.073*	0.096*
	Pleural Effusion	-0.013	-0.017*	-0.035*	-0.041*
	Pneumonia	-0.054*	-0.087*	-0.129*	-0.168*
	Pneumothorax	-0.073*	-0.103*	-0.132*	-0.156*
	ALL TASKS	**0.006***	**-0.004**	**-0.011**	**-0.016***

* $p < 0.05$

[a] Enlarged Cardiomediastinum

Table 8: Relative mean AUROC change in OOD tasks after domain adaptation of the encoder. In case of ResNet-50, the comparison was made with the relation to XRV model instead. Evaluation was performed on OOD tasks sampled from NIH Chext xray dataset. Positive changes are marked green. Significant differences were marked with stars.

Encoder	Recognition task	Examples per class			
		5	10	25	50
ViT-MAE-B	Atelectasis	-0.014*	-0.018*	-0.017*	-0.017*
	Cardiomegaly	0.008	0.011*	0.001	0.010*
	Consolidation	-0.026*	-0.030*	-0.027*	-0.022*
	Effusion	-0.011*	-0.006	-0.012*	-0.014*
	Infiltration	0.008	-0.001	0.005	-0.004
	Mass	0.004	-0.010	-0.009	-0.013*
	Nodule	-0.009	0.006	-0.004	-0.009
	Pleural Thickening	-0.003	-0.003	-0.005	-0.005
	Pneumothorax	-0.013*	-0.026*	-0.031*	-0.046*
	ALL TASKS	**-0.006***	**-0.086***	**-0.011***	**-0.013***
DINO-ViT-B/8	Atelectasis	-0.018*	-0.031*	-0.027*	-0.029*
	Cardiomegaly	-0.019*	-0.008	-0.033*	-0.012*
	Consolidation	-0.013	-0.003	-0.001	-0.001
	Effusion	-0.045*	-0.048*	-0.051*	-0.046*
	Infiltration	-0.007	-0.004	-0.007	-0.025*
	Mass	0.001	-0.004	-0.014*	-0.013*
	Nodule	-0.005	-0.033*	-0.030*	-0.039*
	Pleural Thickening	-0.006	0.003	-0.001	-0.003
	Pneumothorax	-0.021*	-0.030*	-0.048*	-0.033*
	ALL TASKS	**-0.015***	**-0.018***	**-0.024***	**-0.022***
DINO-ResNet-50	Atelectasis	-0.047*	-0.062*	-0.075*	-0.080*
	Cardiomegaly	-0.136*	-0.162*	-0.191*	-0.206*
	Consolidation	-0.055*	-0.071*	-0.068*	-0.063*
	Effusion	-0.133*	-0.134*	-0.137*	-0.135*
	Infiltration	-0.045*	-0.047*	-0.081*	-0.092*
	Mass	-0.144*	-0.172*	-0.200*	-0.221*
	Nodule	-0.059*	-0.083*	-0.114*	-0.123*
	Pleural Thickening	-0.076*	-0.085*	-0.105*	-0.098*
	Pneumothorax	-0.162*	-0.206*	-0.216*	-0.217*
	ALL TASKS	**-0.095***	**-0.114***	**-0.132***	**-0.137***

* $p < 0.05$

3.3 Classifiers overview

The examination of model performance with respect to different support set imbalance ratios in COVID-19 and tuberculosis recognition tasks are shown on Figs **??** and **??**. The results indicate that NC classifier is not only highly robust to the imbalance ratio, but in many cases achieves the best overall performance even when data is balanced, in every k-shot setting.

4 Conclusion and discussions

Our work provides a systematic evaluation of the metric learning approach in several few-shot medical image classification tasks. Our experiments have shown that the nearest centroid algorithm is a much more reliable choice as a classification head than k-NN, outperforming the latter in almost every few-shot scenario. We further conclude that both DINO and MAE vision transformers may be a good selection as feature extractors in metric learning models, outperforming general-purpose CNNs by a large margin. The recently updated second version of DINO did not perform better than its baseline in our setting.

The performance of medical-trained CNNs varies, although the best results are seen when the model training domain is closer to the target tasks (XRV models generally outperform RIN). Domain-specific REMEDIS model achieves remarkable results in COVID-19 detection task and is comparable to the XRV and RIN models in tuberculosis recognition. Other self-supervised models, however, noted varying changes in performance after fine-tuning them on the source dataset. This relative performance became straightly negative when the model was applied to out-of-distribution tasks. This leads us to a counter-intuitive conclusion that some self-supervised models may have a greater ability to extract important features from medical data if trained in a natural domain, as we observed with DINO and MAE.

While in our experiments we did not come close to the state-of-the-art classification performance noted by non-few-shot learning models, the simplicity of the presented approach easily allows for further improvement of the pipeline through mixing in other few-shot techniques, such as meta-learning or ensemble methods.

4.1 Different model variants comparison

The superiority of REMEDIS model in COVID-19 classification task (Table 5) is not surprising, as this framework is a recent state-of-the-art solution carved purposely to solve medical classification problems, and the CXR-50 version of this model was trained on chest X-rays. Its high performance suggests that relatively simple and

long-established CNN architectures can achieve outstanding performance in medical imaging and the training strategy is the key factor in that matter.

Aside REMEDIS, ViT models proved to be the best choice for feature extractors in this task, outperforming even domain-specific RIN and XRV models in every case. Surprisingly, the DINOv2-ViT model did not outperform the base DINO-ViT as might have been expected [36]. ViT-MAE model noted slightly lower performance than DINO, which goes on par with conclusions from [29, 24] stating that contrastive self-supervised approaches work better than image-reconstruction-based ones. The CNNs' performance was observed to be comparable, except the base DenseNet-121 which fell behind noticeably (this is also seen clearly on Figs **??** and **??**).

In the case of tuberculosis classification, the best performance was obtained through the incorporation of XRV models, with ViTs being close behind and outperforming RIN models. Again, DINOv2 performed slightly worse than the base DINO. This time REMEDIS model achieved performance similar to ViT models. What is also interesting is that in the tuberculosis task the application of DenseNet-121-RIN results in worse performance than the base DenseNet-121. This suggests that in some cases models trained on natural images may have a greater ability to extract important features from X-ray data than the ones trained on medical, but not X-ray images.

Figure **??**, shows that in a difficult task, such as COVID-19 recognition, the performance of CNN model can be improved by training it on medical data in a supervised way, no matter if it is in-domain (XRV models) or out-of-domain (RIN models). For the easier task, however, the advantages of in-domain training start to become clear, with XRV models greatly outperforming RIN ones.

To compare our results to the state-of-the-art, we note that Zhang et al. [49] reports the performance of the proposed COVID-19 screening method as 0.952 AUROC. Next, Tartaglione et al. [50] report the COVID-19 recognition performance as high as 1.0 AUROC, however, the specificity of the proposed solution is only 0.20. Interestingly, in a work of Shorfuzzaman and Hossain [51] the authors report 0.975 AUROC for 3-way 10-shot classification of COVID-19 and pneumonia with the use of contrastive learning and Siamese network. This highlights the potential of self-supervised learning in medical few-shot imaging. It also indicates that our approach still needs refinement to compete with the most effective methods established for this task. Our model with REMEDIS-CXR-50-M feature extractor achieved 0.721 AUROC in the 2-way 10-shot COVID-19 classification scenario and 0.782 AUROC in 2-way 50-shot setting.

In the work of Saif et al. [52] the AUROC in tuberculosis classification is set at 0.997 for the Montgomery dataset and 0.981 for the Shenzhen dataset. This was achieved through ensemble voting of different handcrafted and deep-learned features with data augmentation. Cahndra et al.[53] achieved 0.95 AUROC on Montgomery and 0.99 AUROC for Shenzhen set with the application of hierarchical feature extraction. Lastly, Rajamaran et al. [54] reports 0.954 AUROC on Shenzhen and 0.964 AUROC on Montgomery datasets. These results were achieved through the bone

suppression technique, and the baseline performance without it is stated to be 0.899 and 0.857 AUROC respectively. This goes on par with our results, as we achieved 0.903 AUROC on these datasets combined in the more difficult 50-shot setting. Some of the works mentioned above mark the possible directions to improve our work in the future, either through introducing ensemble voting or incorporating bone suppression. As our approach is simple in conception, both of these methods would be relatively straightforward to implement within our framework.

4.2 Self-supervised domain adaptation

The results of self-supervised domain adaptation experiments suggest that domain-specific feature extractors are sometimes less effective than general-purpose ones. While the REMEDIS model shows great performance in COVID-19 and tuberculosis recognition tasks, our own attempts to adapt DINO and MAE frameworks similarly were not that successful. It can be said that while their average performance on ID tasks improved (Table 7), this is not the whole picture. In cases such as pneumothorax or edema recognition, there were many scenarios where the performance did not change significantly or even straightly decreased. In the worst case, in the pleural effusion detection task, the measured AUROC dropped in every scenario across all adapted feature extractors. On the other hand, there is an example of a lung lesion recognition task where the performance improved across all scenarios. This inconsistency of results is most notably seen after ResNet fine-tuning when the difference in the performance of the adapted model varied from almost -0.17 to 0.12 AUROC. It shows that it is difficult to indicate which feature extractor is the best for domain adaptation based only on mean performance change, and that the stability of results across many tasks must also be included in the analysis.

The results of the out-of-distribution evaluation (Table 8) are even more discouraging, as there is only one case (ViT-MAE cardiomegaly recognition) where the performance consistently improved. It suggests that the knowledge learned during the self-supervised training sometimes helps with classification on the source dataset, but is not always transferable to OOD tasks. While the reports from [40], [41] and [29] confirm the effectiveness of self-supervised domain adaptation, which we confirm for REMEDIS model [29], our own experiments indicate that this rule should not be uniformly applied to every self-supervised framework. Next, [41] reports minor improvement in the performance after the adaptation of DINO ResNet-50 and DINO DeiT-S transformer [55], yet the gains were mostly within the range of measured standard deviation. While this still marks the domain adaptation as a valid direction for improving a model's efficacy, our experiments show that the gains from self-supervised pre-training should not be taken for granted.

References

[1] Esteva A, Chou K, Yeung S, Naik N, Madani A, Mottaghi A, et al. Deep learning-enabled medical computer vision. NPJ digital medicine. 2021;4(1):5.

[2] Lakhani P, Sundaram B. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. Radiology. 2017;284(2):574-82.

[3] Huynh BQ, Li H, Giger ML. Digital mammographic tumor classification using transfer learning from deep convolutional neural networks. Journal of Medical Imaging. 2016;3(3):034501-1.

[4] Lee H, Tajmir S, Lee J, Zissen M, Yeshiwas BA, Alkasab TK, et al. Fully automated deep learning system for bone age assessment. Journal of digital imaging. 2017;30:427-41.

[5] McBee MP, Awan OA, Colucci AT, Ghobadi CW, Kadom N, Kansagra AP, et al. Deep learning in radiology. Academic radiology. 2018;25(11):1472-80.

[6] Chen X, Yao L, Zhou T, Dong J, Zhang Y. Momentum contrastive learning for few-shot COVID-19 diagnosis from chest CT images. Pattern recognition. 2021;113:107826.

[7] Jiang Y, Chen H, Ko H, Han DK. Few-shot learning for ct scan based covid-19 diagnosis. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE; 2021. p. 1045-9.

[8] Cores D, Vila-Blanco N, Pérez-Alarcón M, Martínez-de Alegría A, Mucientes M, Carreira MJ. A few-shot approach for COVID-19 screening in standard and portable chest X-ray images. Scientific Reports. 2022;12(1):21511.

[9] Reddy Bhimireddy A, Burns JL, Purkayastha S, Wawira Gichoya J. Few-Shot Transfer Learning to improve Chest X-Ray pathology detection using limited triplets. arXiv e-prints. 2022:arXiv-2204.

[10] Paul A, Tang YX, Shen TC, Summers RM. Discriminative ensemble learning for few-shot chest x-ray diagnosis. Medical image analysis. 2021;68:101911.

[11] Hu SX, Li D, Stühmer J, Kim M, Hospedales TM. Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022. p. 9068-77.

[12] Vinyals O, Blundell C, Lillicrap T, Wierstra D, et al. Matching networks for one shot learning. Advances in neural information processing systems. 2016;29.

[13] Cai Q, Pan Y, Yao T, Yan C, Mei T. Memory matching networks for one-shot image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2018. p. 4080-8.

[14] Li Y, Gu C, Dullien T, Vinyals O, Kohli P. Graph matching networks for learning the similarity of graph structured objects. In: International conference on machine learning. PMLR; 2019. p. 3835-45.

[15] Altae-Tran H, Ramsundar B, Pappu AS, Pande V. Low data drug discovery with one-shot learning. ACS central science. 2017;3(4):283-93.

[16] Bachman P, Sordoni A, Trischler A. Learning algorithms for active learning. In: international conference on machine learning. PMLR; 2017. p. 301-10.

[17] Snell J, Swersky K, Zemel R. Prototypical networks for few-shot learning. Advances in neural information processing systems. 2017;30.

[18] Oreshkin B, Rodriguez Lopez P, Lacoste A. Tadam: Task dependent adaptive metric for improved few-shot learning. Advances in neural information processing systems. 2018;31.

[19] Ren M, Triantafillou E, Ravi S, Snell J, Swersky K, Tenenbaum JB, et al. Meta-learning for semi-supervised few-shot classification. arXiv preprint arXiv:180300676. 2018.

[20] Wang YX, Girshick R, Hebert M, Hariharan B. Low-shot learning from imaginary data. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2018. p. 7278-86.

[21] Cai A, Hu W, Zheng J. Few-shot learning for medical image classification. In: Artificial Neural Networks and Machine Learning–ICANN 2020: 29th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 15–18, 2020, Proceedings, Part I 29. Springer; 2020. p. 441-52.

[22] Jin Y, Lu H, Zhu W, Yan K, Gao Z, Li Z. CTFC: A Convolution and Visual Transformer Based Classifier for Few-Shot Chest X-ray Images. In: 2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE). IEEE; 2021. p. 616-22.

[23] Ohri K, Kumar M. Review on self-supervised image recognition using deep neural networks. Knowledge-Based Systems. 2021;224:107090.

[24] Liu X, Zhang F, Hou Z, Mian L, Wang Z, Zhang J, et al. Self-supervised learning: Generative or contrastive. IEEE transactions on knowledge and data engineering. 2021;35(1):857-76.

[25] Krishnan R, Rajpurkar P, Topol EJ. Self-supervised learning in medicine and healthcare. Nature Biomedical Engineering. 2022;6(12):1346-52.

[26] Shurrab S, Duwairi R. Self-supervised learning methods and applications in medical imaging analysis: A survey. PeerJ Computer Science. 2022;8:e1045.

[27] Cohen JP, Viviano JD, Bertin P, Morrison P, Torabian P, Guarrera M, et al. TorchXRayVision: A library of chest X-ray datasets and models. In: Medical Imaging with Deep Learning; 2022. Available from: https://github.com/mlmed/torchxrayvision.

[28] Mei X, Liu Z, Robson PM, Marinelli B, Huang M, Doshi A, et al. RadImageNet: An Open Radiologic Deep Learning Research Dataset for Effective Transfer Learning. Radiology: Artificial Intelligence. 0;0(ja):e210315. Available from: https://doi.org/10.1148/ryai.210315.

[29] Azizi S, Culp L, Freyberg J, Mustafa B, Baur S, Kornblith S, et al. Robust and data-efficient generalization of self-supervised machine learning for diagnostic imaging. Nature Biomedical Engineering. 2023:1-24.

[30] Caron M, Touvron H, Misra I, Jégou H, Mairal J, Bojanowski P, et al. Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision; 2021. p. 9650-60.

[31] He K, Chen X, Xie S, Li Y, Dollár P, Girshick R. Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022. p. 16000-9.

[32] Irvin J, Rajpurkar P, Ko M, Yu Y, Ciurea-Ilcus S, Chute C, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33; 2019. p. 590-7.

[33] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770-8.

[34] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 4700-8.

[35] Liu Z, Mao H, Wu CY, Feichtenhofer C, Darrell T, Xie S. A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2022. p. 11976-86.

[36] Oquab M, Darcet T, Moutakanni T, Vo H, Szafraniec M, Khalidov V, et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:230407193. 2023.

[37] Kolesnikov A, Beyer L, Zhai X, Puigcerver J, Yung J, Gelly S, et al. Big transfer (bit): General visual representation learning. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16. Springer; 2020. p. 491-507.

[38] Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: International conference on machine learning. PMLR; 2020. p. 1597-607.

[39] Johnson AE, Pollard TJ, Berkowitz SJ, Greenbaum NR, Lungren MP, Deng Cy, et al. MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports. Scientific data. 2019;6(1):317.

[40] Hosseinzadeh Taher MR, Haghighi F, Feng R, Gotway MB, Liang J. A systematic benchmarking analysis of transfer learning for medical image analysis. In: Domain Adaptation and Representation Transfer, and Affordable Healthcare and AI for Resource Diverse Global Health: Third MICCAI Workshop, DART 2021, and First MICCAI Workshop, FAIR 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27 and October 1, 2021, Proceedings 3. Springer; 2021. p. 3-13.

[41] Matsoukas C, Haslum JF, Söderberg M, Smith K. Is it time to replace cnns with transformers for medical images? arXiv preprint arXiv:210809038. 2021.

[42] Zhou L, Liu H, Bae J, He J, Samaras D, Prasanna P. Self pre-training with masked autoencoders for medical image analysis. arXiv preprint arXiv:220305573. 2022.

[43] Truong T, Mohammadi S, Lenga M. How transferable are self-supervised features in medical image classification tasks? In: Machine Learning for Health. PMLR; 2021. p. 54-74.

[44] Caron M, Misra I, Mairal J, Goyal P, Bojanowski P, Joulin A. Unsupervised learning of visual features by contrasting cluster assignments. Advances in neural information processing systems. 2020;33:9912-24.

[45] Laenen S, Bertinetto L. On episodes, prototypical networks, and few-shot learning. Advances in Neural Information Processing Systems. 2021;34:24581-92.

[46] Cohen JP, Morrison P, Dao L. COVID-19 image data collection. arXiv 200311597. 2020. Available from: `https://github.com/ieee8023/covid-chestxray-dataset`.

[47] Jaeger S, Candemir S, Antani S, Wáng YXJ, Lu PX, Thoma G. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases. Quantitative imaging in medicine and surgery. 2014;4(6):475.

[48] Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM. ChestX-ray8: Hospital-Scale Chest X-Ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2017. .

[49] Zhang J, Xie Y, Li Y, Shen C, Xia Y. Covid-19 screening on chest x-ray images using deep learning based anomaly detection. arXiv preprint arXiv:200312338. 2020;27(10.48550).

[50] Tartaglione E, Barbano CA, Berzovini C, Calandri M, Grangetto M. Unveiling covid-19 from chest x-ray with deep learning: a hurdles race with small data. International Journal of Environmental Research and Public Health. 2020;17(18):6933.

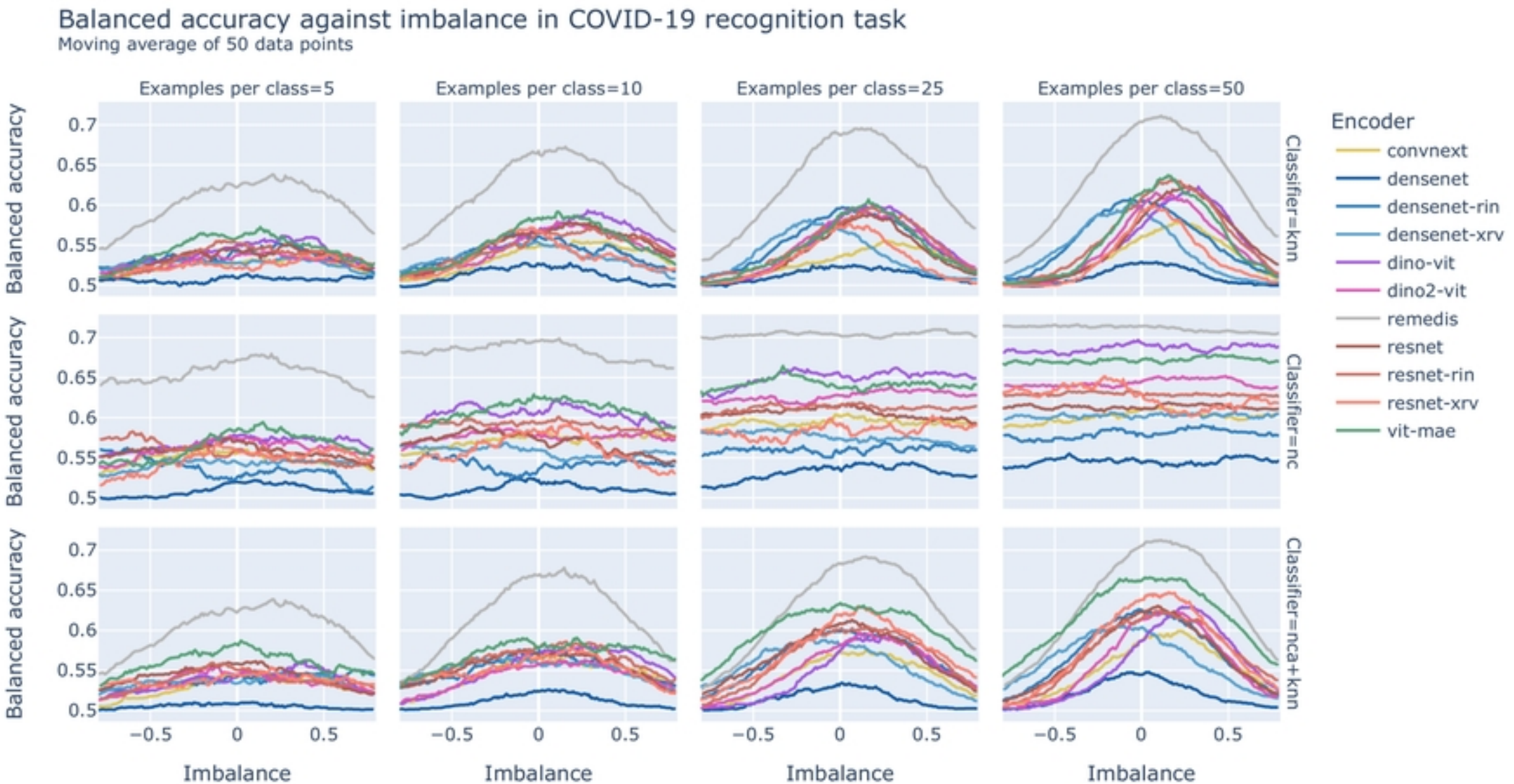
[51] Shorfuzzaman M, Hossain MS. MetaCOVID: A Siamese neural network framework with contrastive loss for n-shot diagnosis of COVID-19 patients. Pattern recognition. 2021;113:107700.

[52] Saif A, Imtiaz T, Shahnaz C, Zhu WP, Ahmad MO. Exploiting cascaded ensemble of features for the detection of tuberculosis using chest radiographs. IEEE Access. 2021;9:112388-99.

[53] Chandra TB, Verma K, Singh BK, Jain D, Netam SS. Automatic detection of tuberculosis related abnormalities in Chest X-ray images using hierarchical feature extraction scheme. Expert Systems with Applications. 2020;158:113514.

[54] Rajaraman S, Zamzmi G, Folio L, Alderson P, Antani S. Chest x-ray bone suppression for improving classification of tuberculosis-consistent findings. Diagnostics. 2021;11(5):840.

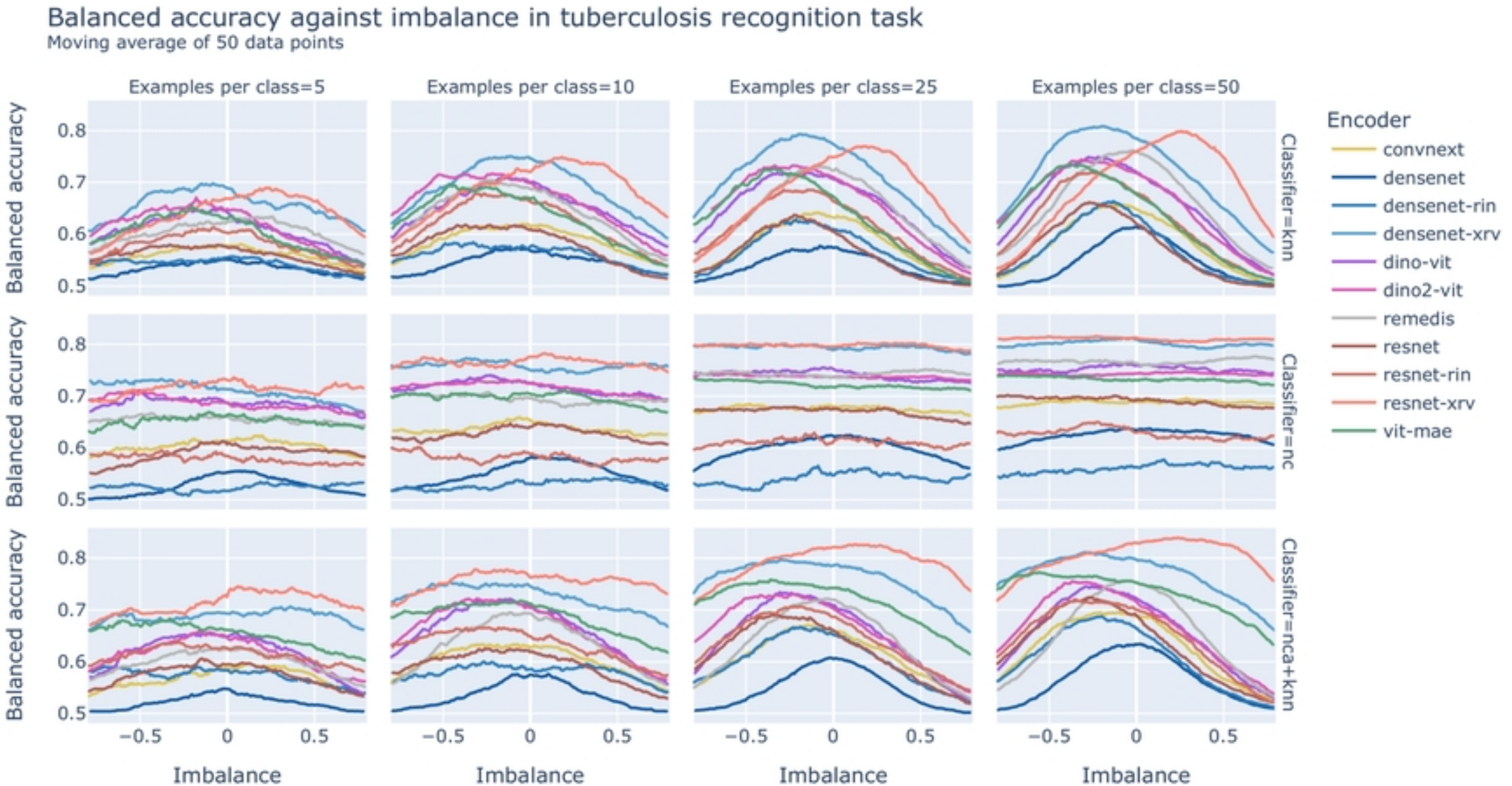
[55] Touvron H, Cord M, Douze M, Massa F, Sablayrolles A, Jégou H. Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. PMLR; 2021. p. 10347-57.


Balanced accuracy against imbalance in COVID-19 recognition task
Moving average of 50 data points
Examples per class=5
Examples per class=10
Examples per class=25
Examples per class=50
Balanced accuracy
Classifier=knn
Classifier=nc
Classifier=nca+knn
Imbalance
Encoder
convnext
densenet
densenet-rin
densenet-xrv
dino-vit
dino2-vit
remedis
resnet
resnet-rin
resnet-xrv
vit-mae

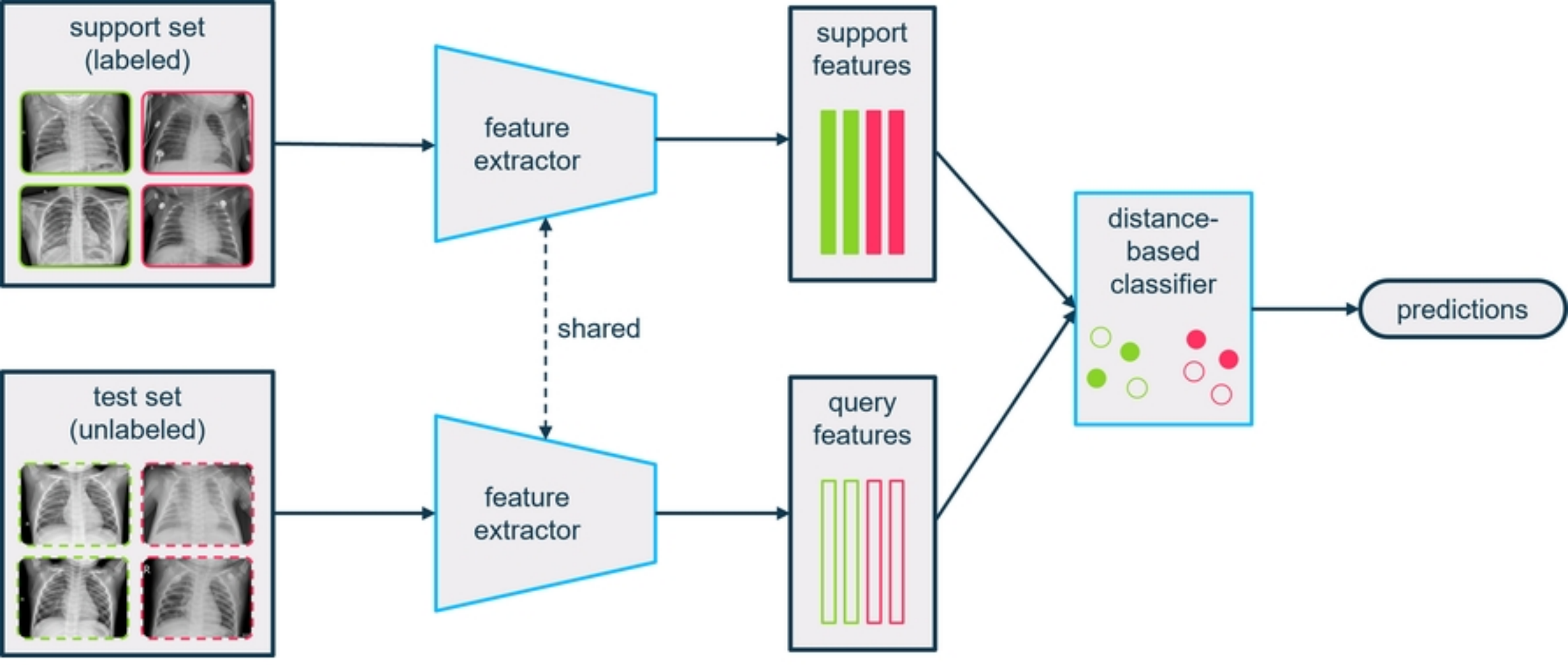
Figure

AUROC by dataset, encoder, classifier and number of examples
Dataset=covid-chest-xray
Dataset=nlm-tuberculosis-chest-xray
AUROC
0.8
0.6
0.4
Classifier=knn
Classifier=nc
Classifier=nca+knn
5
10
25
50
Examples per class
Encoder
densenet
densenet-rin
densenet-xrv
resnet
resnet-rin
resnet-xrv
remedis (resnet-50-cxr)

Figure


Balanced accuracy against imbalance in tuberculosis recognition task
Moving average of 50 data points
Examples per class=5
Examples per class=10
Examples per class=25
Examples per class=50
Balanced accuracy
Classifier=knn
Classifier=nc
Classifier=nca+knn
Imbalance
Encoder
convnext
densenet
densenet-rin
densenet-xrv
dino-vit
dino2-vit
remedis
resnet
resnet-rin
resnet-xrv
vit-mae

Figure

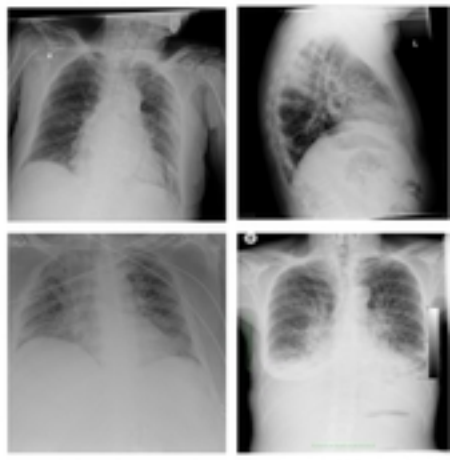

support set
(labeled)
feature
extractor
support
features
distance-
based
classifier
predictions
shared
test set
(unlabeled)
feature
extractor
query
features

Figure

COVID-19 Image Data Collection

Findings: **Pneumonia/Viral/Influenza**

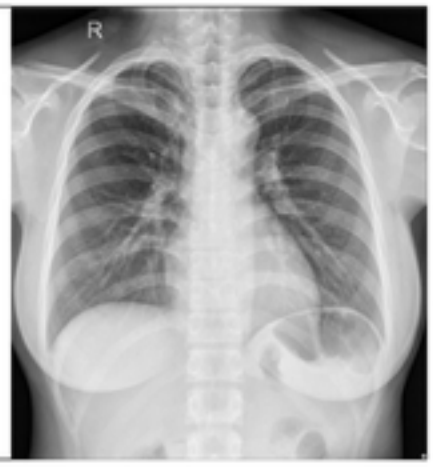
Report: A 52-year-old man, active smoker with chronic obstructive lung disease, was admitted with pneumonia and influenza. Treated in intensive care unit with mechanical ventilation for nine days. Routine chest radiograph taken 7 days after extubation.



Shenzhen dataset

female 28yrs

bilateral PTB



Montgomery dataset

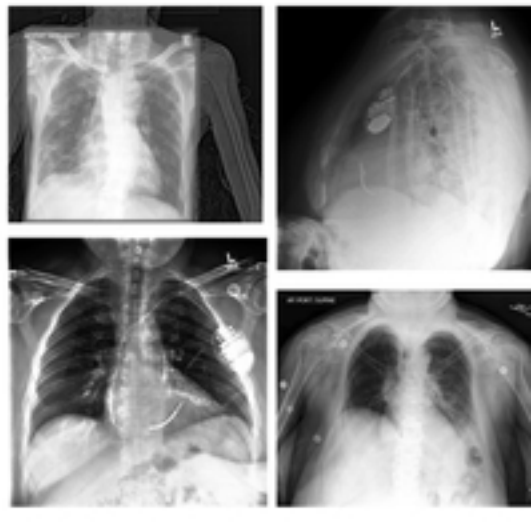
Patient's Sex: F
Patient's Age: 011Y

normal

CheXpert dataset

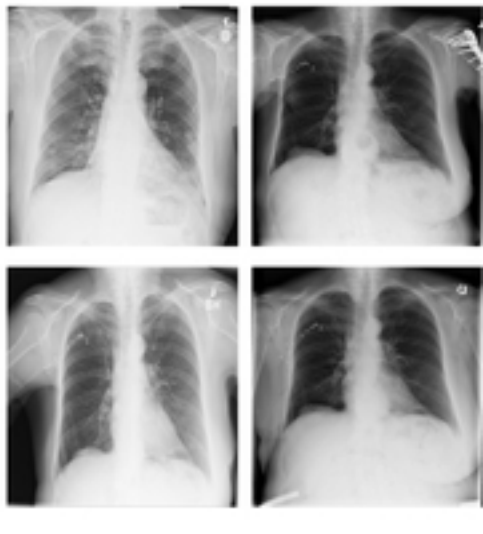
Labels:

No Finding	0.0	**Enlarged Cardiomediastinum :**	**1.0**
Cardiomegaly	**1.0**	**Lung Opacity :**	**1.0**
Lung Lesion	0.0	Edema :	0.0
Consolidation	0.0	Pneumonia :	0.0
Atelectasis	0.0	Pneumothorax :	0.0
Pleural Effusion	0.0	Pleural Other :	0.0
Fracture	0.0	**Support Devices :**	**1.0**



NIH Chest X-Ray dataset

Finding Labels :	**Cardiomegaly**
Follow-up:	0
Patient ID:	1
Patient Age:	58
Patient Gender:	M
View Position:	PA
Width:	2566
Height:	2681
Pixel spacing:	0.143



Figure