Federated Contrastive Learning for Decentralized Unlabeled Medical Images

Anonymous

Anonymous Organization
@****.***

Abstract. A label-efficient paradigm in computer vision is based on self-supervised contrastive pre-training on unlabeled data followed by fine-tuning with a small number of labels. Making practical use of a federated computing environment in the clinical domain and learning on medical images poses specific challenges. In this work, we propose `FedMoCo`, a robust federated contrastive learning (FCL) framework, which makes efficient use of decentralized unlabeled medical data. `FedMoCo` has two novel modules: *metadata transfer*, an inter-node statistical data augmentation module, and *self-adaptive aggregation*, an aggregation module based on representational similarity analysis. To the best of our knowledge, this is the first FCL work on medical images. Our experiments show that `FedMoCo` can consistently outperform `FedAvg`, a seminal federated learning framework, in extracting meaningful representations for downstream tasks. We further show that `FedMoCo` can substantially reduce the amount of labeled data required in a downstream task, such as COVID-19 detection, to achieve a reasonable performance. It is evident from our results that FCL is a promising direction for digital medicine.

Keywords: Federated Learning · Contrastive Representation Learning.

1 Introduction

Recent studies in self-supervised learning (SSL) [20] have led to a renaissance of research on contrastive representation learning or contrastive learning (CL) [1]. Self-supervised CL or unsupervised CL aims to learn useful and transferable representations from unlabeled data. In a CL framework, a model is first pre-trained on unlabeled data in a self-supervised fashion via a contrastive loss, and then fine-tuned on labeled data. Utilizing the state-of-the-art (SOTA) CL frameworks [7, 15, 3, 23], a model trained with only unlabeled data plus a small amount of labeled data can achieve comparable performance on various downstream tasks with the same model trained with a large amount of labeled data.

As a data-driven approach, deep learning has fueled many breakthroughs in medical image analysis (MIA). Meanwhile, large-scale fully labeled medical datasets require considerable human annotation cost, which makes data scarcity a major bottleneck in practical research and applications. To leverage unlabeled data, there have been CL works [9, 4, 22] demonstrating promising results. However, none of these studies consider federated learning (FL), which has received

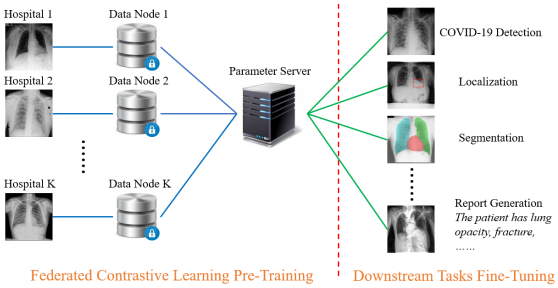


Fig. 1: Illustration of FCL workflow for chest X-ray images (CXRs). Each data node (hospital) works on an independent local dataset. The parameter server works with data nodes for periodic synchronization, metadata transfer (Sec. 2.2), and self-adaptive aggregation (Sec. 2.3). For each downstream task, the pre-trained model weights are fine-tuned with small amount of labeled data in either supervised or semi-supervised fashion. Note, the labeled data of a downstream task can only be accessed *locally* due to data privacy regulation.

increasing attention for handling sensitive private data stored in multiple devices [21, 13, 19]. Due to data privacy regulations, since patient data usually contains confidential information, it is difficult to collect clinical data from different hospitals for standard data-centralized CL. Under FL protocols, medical images (either raw or encoded) are not allowed to be exchanged between data nodes. In this work, we aim to understand how to design FCL framework on decentralized medical data (i.e. medical images stored on multiple devices or at multiple locations) to extract useful representations for MIA.

There are two direct *negative* impacts of FL environment on CL. First, domain shift exists between data nodes (e.g. hospitals), as different nodes may have different imaging protocols [6]. In contrast to CL on centralized data, each node in FCL only has access to its local data, which has a smaller variation in sample distribution. This will impair the CL performance on each single node. Second, FL assumes non-IID data distributions across data nodes. Without the supervision of any ground truth labels, FCL could be easily dominated by a few nodes, which leads to a worse generalization ability of the final learning outcome. How to aggregate CL models across nodes is still an open question.

In this work, we present `FedMoCo`, a robust FCL framework for MIA (see Fig. 1 for illustration). `FedMoCo` uses MoCo [7] as the intra-node CL model. To mitigate the aforementioned methodological gaps, we propose *metadata transfer*, an inter-node augmentation module utilizing Box-Cox power transformation [2] and Gaussian modeling, and *self-adaptive aggregation*, an aggregation module based on representational similarity analysis (RSA) [11]. We empirically evaluate `FedMoCo` under various simulated scenarios. Our experiments show that `FedMoCo` can consistently outperform `FedAvg` [14], a SOTA FL framework and `FedMoCo` can efficiently reduce annotation cost in downstream tasks, such as COVID-19 detection. By pre-training on unlabeled non-COVID datasets, `FedMoCo` requires

only 3% of examples from a labeled COVID-19 dataset to achieve 90% in accuracy. Our contributions are threefold: (1) to the best of our knowledge, this is the first work of FCL on medical images; (2) we propose `FedMoCo` with two novel modules; (3) our results provide insights into future research on FCL.

2 Method

Problem Formulation Assume $K > 1$ is the number of nodes, $\mathcal{D}_k$ denotes an *unlabeled* dataset stored in node k, where $k \in \{1, \cdots, K\}$. For node k, we have $\mathcal{D}_k = \{x_i\}_{i=1}^{n_k}$, where x_i is a medical image and $n_k = |\mathcal{D}_k|$. As a standard practice in FL, we assume that $\mathcal{D}_k \cap \mathcal{D}_l = \emptyset$ wheneve $k \neq l$ and $\{\mathcal{D}_k\}_{k=1}^K$ are non-IID data. In addition to K data nodes, we assume there is another master node, which does not store any clinical data. In this work, the master node is implemented as a *parameter server* (PS) [12].

Assume the model we are interested in is denoted as f_θ and the parameter set θ_0 is randomly initialized in PS. At the beginning of the federated training, K copies of θ_0 are distributed into each node as $\{\theta_k\}_{k=1}^K$, i.e. we fully synchronize the data nodes with PS. An important protocol of FL is data privacy-preserving, i.e. exchanging training data between nodes is strictly prohibited. Instead, node k updates θ_k by training on $\mathcal{D}_k$ independently. After the same number of local epochs, $\{\theta_k\}_{k=1}^K$ are aggregated into θ_0 in PS. Again, we synchronize $\{\theta_k\}_{k=1}^K$ with aggregated θ_0. This process is repeated until certain criteria are met. Note, to enforce data privacy regulation, we are only allowed to exchange model parameters $\{\theta_k\}_{k=0}^K$ and metadata between the data nodes and PS in the whole training process. The learning outcome is θ_0 for relevant downstream tasks.

2.1 Intra-Node Contrastive Learning

We create a positive pair by generating two random views of the same images through data augmentation. A negative pair is formed by taking two random views from two different images. Concretely, given an image x and a family of stochastic image transformations T, we randomly sample two transformations t and t' to have two random views $t(x)$ and $t'(x)$. $t(x)$ and $t'(x)$ form a positive pair. Let z denote the representation of an image extracted by a CNN encoder f_θ. In contrast to previous works [26, 7, 3], we add a ReLU function between the last fully-connected layer and a L2-normalization layer, which will project all extracted features into a non-negative feature space. So, for a query image $x \in \mathcal{D}_k$, we have positive pair z_q and z_0. Assume there are N negative examples $\{x_i\}_{i=1}^N$, we have the contrastive loss InfoNCE [17] defined as

$$\mathcal{L}_q = -\log \frac{\exp(z_q \cdot z_0/\tau)}{\sum_{i=0}^{N} \exp(z_q \cdot z_i/\tau)} \tag{1}$$

where τ is the temperature parameter.

We use the dynamic dictionary with momentum update in MoCo [7] to maintain a large number of negative examples in Eq. 1. In node k, there are a CNN

encoder (θ_k^q) for the query image and another CNN encoder (θ_k^d) for the corresponding positive example, which are defined as

$$\theta_k^q \leftarrow \theta_k,\ \theta_k^d \leftarrow m\theta_k^d + (1-m)\theta_k^q, \tag{2}$$

where $m \in [0, 1)$ is the momentum coefficient. For query image x, we have $z_q = f_{\theta_k^q}(t(x))$ and $z_0 = f_{\theta_k^q}(t'(x))$.

2.2 Metadata Transfer

Apparently, the dynamic dictionary of node k is not allowed to exchange information with other nodes in a federated environment. This limits the sample variation of the dynamic dictionary, which limits CL performance. In addition, a node overfitted to local data may not generalize well to other nodes. To address these two concerns, we propose a simple yet robust solution by utilizing metadata of the encoded feature vectors in each node. For node k, after the full synchronization with PS (i.e. before the start of next round of local updates), we first extract the feature vectors $\{f_{\theta_k}(x_i)\}_{i=1}^{n_k}$ of $\mathcal{D}_k$. We assume the feature distribution is Gaussian. In order to enforce this assumption, we use Box-Cox power transformation (BC) to make the feature distribution more Gaussian-like. BC is a reversible transformation defined as

$$BC(x) = \begin{cases} \frac{x^\lambda - 1}{\lambda}, & \lambda \neq 0 \\ \log(x), & \lambda = 0 \end{cases} \tag{3}$$

where λ controls the skewness of the transformed distribution. Then, we calculate the mean and covariance of the transformed features

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{n_k} y_i}{n_k},\ \boldsymbol{\Sigma}_k = \frac{\sum_{i=1}^{n_k}(y_i - \boldsymbol{\mu}_k)(y_i - \boldsymbol{\mu}_k)^T}{n_k - 1} \tag{4}$$

where $y_i = \mathrm{BC}(f_{\theta_k}(x_i))$. The metadata of the learned representations in each node are collected by PS as $\{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$. $\{(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\}_{j \neq k}$ is sent to node k. We name this operation as *metadata transfer*. Metadata transfer aims to improve CL performance of node k by augmenting the dynamic dictionary of node k statistically with metadata collected from node $j \neq k$.

For each pair of $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, there is a corresponding Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. In the next round of local updates, we increase the sample variation in each node by sampling new examples from Gaussian distributions when minimizing Eq. 1. Specifically, for node k, we increase the number of negative examples in Eq. 1 from N to $\lfloor N(1+\eta) \rfloor$, where $\eta \geq 0$ is a hyper-parameter to control the level of interaction between the node k and the other nodes (e.g. if $\eta = 0$, there is no interaction). We sample $\left\lfloor \frac{\eta N}{K-1} \right\rfloor$ examples from each $\mathcal{N}(\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)\ \forall\ l \neq k$. Let $\tilde{y} \sim \mathcal{N}(\boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)\ \forall\ l \neq k$, we have $\tilde{z} = BC^{-1}(\tilde{y})$ and the new contrastive loss is

$$\mathcal{L}_q = -\log \frac{\exp(z_q \cdot z_0/\tau)}{\sum_{i=0}^{N} \exp(z_q \cdot z_i/\tau) + \sum_{j=1}^{(K-1)\lfloor \frac{\eta N}{K-1} \rfloor} \exp(z_q \cdot \tilde{z_j}/\tau)} \tag{5}$$

Algorithm 1 `FedMoCo`. The training in each data node is warmed up for t_w rounds before metadata transfer. A local round could be a few local epochs.

Initialize θ_0^0
for $t = 1, 2, \cdots, T$ **do**
  for $k = 1, 2, \cdots, K$ **do**
    $\theta_k^t \leftarrow \theta_0^{t-1}$ ▷ Synchronize with PS
    if $t \leq t_w$ **then** ▷ Warm up
      $\theta_k^t \leftarrow \textit{local\_update}(\theta_k^t)$ ▷ Eq. 1
    else ▷ Metadata transfer
      Download $\{(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\}_{j \neq k}$ from PS
      $\theta_k^t \leftarrow \textit{local\_update}(\theta_k^t, \{(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)\}_{j \neq k})$ ▷ Eq. 5
      Upload $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ to PS
  Compute A ▷ Eq. 6 and Eq. 7
  $\theta_0^t \leftarrow \sum_{k=1}^K a_k^t \theta_k^t$
Output θ_0^T

2.3 Self-Adaptive Aggregation

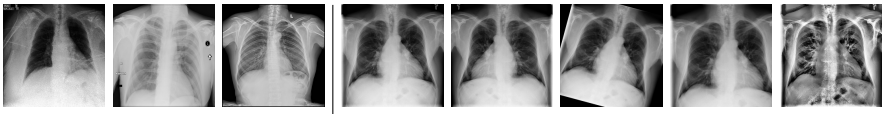
Given locally updated $\{\theta_k^t\}_{k=1}^K$, which are collected at the end of round t, the aggregation step of round t can be formulated as $\theta^t = \theta_0^t = \sum_{k=1}^K a_k \theta_k^t$, where a_k denotes k^{th} element of diagonal matrix A. For example, in `FedAvg` [14], we use $a_k = \frac{n_k}{\sum_{j=1}^K n_j}$ because the number of labels indicates the strength of supervision in each node, which does not work for unsupervised learning. For node k with large n_k and small variation in $\mathcal{D}_k$ (i.e. examples are similar), f_{θ_k} converges faster but θ_k learns less meaningful representations (e.g. overfitted to no findings or certain diseases). But $a_k = \frac{n_k}{\sum_{j=1}^K n_j}$ gives θ_k^t a larger weight and θ_k^0 is dominated by θ_k^t. Instead, we propose *self-adaptive aggregation* to compute matrix A.

Let $-1 \leq r_k \leq 1$ denote representational similarity analysis (RSA) [11] of node k. In t^{th} round of local updates, we first randomly sample $n_k{}'$ examples from $\mathcal{D}_k$ as a subset $\mathcal{D}_k^{'}$ for computational and statistical efficiency. For each $x_i \in \mathcal{D}_k^{'}$, we get two representations $f_{\theta_0^{t-1}}(x_i)$ (aggregated weights at the end of round $t-1$, which are also globally synchronized weights at the beginning of round t) and $f_{\theta_k^t}(x_i)$ (locally updated weights at the end of round t). We define ρ_{ij} as Pearson's correlation coefficient between $f_\theta(x_i)$ and $f_\theta(x_j)$ $\forall\, 0 \leq i, j \leq n_k{}'$ and define $\text{RDM}_{ij} = 1 - \rho_{ij}$, where RDM is representation dissimilarity matrix [11] for f_θ. Then, r_k is defined based on Spearman's rank correlation:

$$r_k = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)} \tag{6}$$

where d_i is the difference between the ranks of i^{th} elements of the lower triangular of RDM for $f_{\theta^{t-1}}$ and RDM for $f_{\theta_k^t}$, and $n = n_k{}'(n_k{}' - 1)/2$. We define A for the aggregation step at the end of round t as

$$a_k^t = \frac{1 - r_k}{\sum_{j=1}^K 1 - r_j}. \tag{7}$$



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 2: CXR visualization. 1. Unsupervised pre-training datasets: (a) CheXpert; (b) ChestX-ray8; (c) VinDr-CXR. 2. Stochastic data augmentation: (d) raw; (e) flip; (f) rotate; (g) crop and resize; (h) histogram equalization.

dataset	size	# of classes	multi-label	multi-view	balanced	resolution
CheXpert [10]	371920	14	✓	✓	✗	390×320
ChestX-ray8 [25]	112120	15	✓	✗	✗	1024×1024
VinDr-CXR [16]	15000	15	✓	✗	✗	512×512
COVID-19 [5]	3886	3	✗	✗	✓	256×256

Table 1: Dataset description.

We use r_k to measure CL performance of node k in round t. Intuitively, given same θ^{t-1} at the beginning of round t, smaller r_k indicates there is a bigger update in the representations, i.e. node k has extracted more meaningful representations. Eq. 7 assigns larger weights to local models with higher potentials in representational power. The complete pseudo-code is given in Algorithm 1.

3 Experiments

3.1 Experimental Setup

For a fair comparison, we use the same set of hyperparameters and the same training strategy for all experiments. We use ResNet18 [8] as the network backbone. We use the Momentum optimizer with momentum 0.9. The initial learning rate is 0.03 and is multiplied by 0.1 and 0.01 at 120 epochs and 160 epochs respectively. The batch size is 64 for each node. The weight decay is 10^{-4}. m in Eq. 2 is 0.999. λ in Eq. 3 is 0.5. τ is 0.2, N is 1024, and η is 0.05 in Eq. 5. We estimate RSA with $n_k{}' = 100$.

As there is no existing FCL for MIA, we compare `FedMoCo` with a strong baseline FCL model, which is an integration of MoCo [7] and `FedAvg` [14], a seminal supervised FL model. Note, `FedAvg` is robust against domain shift cross data nodes. All models are implemented with PyTorch on NVIDIA Tesla V100.

Datasets FCL should work for any type of medical images. Without loss of generality, we demonstrate FCL on anterior and posterior CXRs. We use three public large-scale CXR datasets as the unlabeled pre-training data to simulate the federated environment, namely CheXpert [10], ChestX-ray8 [25], and VinDr-CXR [16] (see Table 1 for descriptions of datasets used in this study.). Three datasets are collected and annotated from different sources independently and express a large variety in data modalities (see Fig. 2(a)-(c) for illustrations). The

original images are cropped and downsampled to 256×256. Note, three datasets contain noisy labels and the label distributions are different.

Data Augmentation Driven by clinical domain knowledge and experimental findings, We provide a augmentation policy as illustrated in Fig. 2(d)-(h). Compared with previous studies of MoCo on medical images [9, 4, 22], we propose stochastic histogram equalization for CL on medical images. The stochasticity comes from uniformly sampling the parameters of CLAHE [18]. For a fair comparison, we use the same augmentation policy for all models in this work.

Linear Classification Protocol We evaluate the performance of unsupervised pre-training by following *linear classification protocol*(LCP) [7, 15, 3, 23]. ResNet18 is first pre-trained on the unlabeled dataset. Then a supervised linear classifier (a fully-connected layer) is trained on top of the frozen features extracted from a labeled dataset for 50 epochs with a constant learning rate 0.1. We report the classification accuracy on the validation set of the labeled data as the performance of FCL. We choose a public COVID-19 CXR dataset [5] with 3886 CXRs. Note, COVID-19 is unseen in three pre-training datasets. We use 50% CXRs for training and 50% for testing.

3.2 Evaluation

The experiments are simulated in a controllable federated environment. To eliminate the effect of n_k, we first evaluate `FedMoCo` in a situation that each node has the same number of CXRs. We create 3 data nodes by randomly sampling 10000 CXRs from each of three datasets. We create 6 nodes by partitioning each data node equally. We evaluate the pre-training performance on $T = 200$ and $T = 400$, where one round is one local epoch and `FedMoCo` is warmed up for $t_w = 50$ epochs. We provide the performance of MoCo trained in a single node with the same data as *Oracle* for centralized CL. LCP results with mean and standard deviation of 3 rounds are present in Table 2. We have two empirical findings for FCL: more nodes will decrease the performance, and more training time may not improve the performance of downstream tasks.

To show the isolated contribution of metadata transfer and self-adaptive aggregation, we simulate two common situations in FL with $K = 3$ and $T = 200$. First, with the same data as above, we create unbalanced data distributions in terms of the number of examples by only using γ% CXRs in nodes of CheXpert and ChestX-ray8. We use `FedMoCo-M` to denote `FedMoCo` with only metadata transfer module and use `FedMoCo-S` to denote `FedMoCo` with only self-adaptive aggregation. The results are presented in Table 3. Second, with the same number of examples in each node, we create unbalanced data distributions in terms of the label distribution by only sampling healthy CXRs (no findings) for the nodes of CheXpert and ChestX-ray8 and only sampling CXRs with disease labels for the node of VinDr-CXR. The results are presented in Table 4. In summary the above results, `FedMoCo` can outperform `FedAvg` consistently under non-IID challenges and FCL can outperform centralized CL when domain shift exists across nodes.

Downstream Task Evaluation To demonstrate the practical value of the representations extracted by FCL under data scarcity challenge, we use COVID-

Model	$T = 200$ (%)	$T = 400$ (%)
`FedAvg`($K = 3$)	94.92 ± 0.16	95.03 ± 0.23
`FedMoCo`($K = 3$)	95.73 ± 0.05	95.58 ± 0.26
`FedAvg`($K = 6$)	94.60 ± 0.32	94.65 ± 0.94
`FedMoCo`($K = 6$)	95.11 ± 0.71	94.92 ± 1.27
Oracle	92.80 ± 0.05	92.37 ± 0.06

Table 2: LCP comparison between FCL models on non-IID distributions with equal numbers of examples.

Model	$\gamma = 5$ (%)	$\gamma = 10$ (%)
`FedAvg`	94.27 ± 1.37	94.77 ± 0.86
`FedMoCo-M`	94.66 ± 1.57	95.23 ± 0.96
`FedMoCo-S`	94.37 ± 0.77	95.01 ± 0.45
`FedMoCo`	94.77 ± 0.80	95.58 ± 0.31
Oracle	95.22 ± 0.05	95.23 ± 0.39

Table 3: LCP comparison between FCL models on non-IID distributions with imbalanced numbers of examples.

Model	$n_k = 5000$ (%)	$n_k = 10000$ (%)
`FedAvg`	93.30 ± 1.21	95.25 ± 0.33
`FedMoCo-M`	93.67 ± 1.19	95.43 ± 0.36
`FedMoCo-S`	94.11 ± 0.88	95.74 ± 0.29
`FedMoCo`	94.38 ± 0.68	96.02 ± 0.25
Oracle	95.78 ± 0.22	94.82 ± 0.11

Table 4: LCP comparison between FCL models on non-IID distributions with imbalanced label distribution.

Model	Accuracy (%)
w/o Pre-Training	79.63
`FedAvg`	87.24
`FedMoCo`	91.56
Oracle(MoCo)	88.42
Oracle(Supervised)	95.78

Table 5: Fine-tuning models pre-trained by FCL for COVID-19 detection.

19 detection [5] as the downstream task. We use the same training set and test set in LCP. We fine-tune the ResNets pre-trained by FCL models in Table 2 ($K = 3$ and $T = 200$) with only 3% of the training set. We train a randomly initialized ResNet with the full training set as the *Oracle* for supervised learning. For a fair comparison, we use a fixed learning rate 0.01 to train all models for 100 epochs. We report the highest accuracy in Table 5. `FedMoCo` outperforms centralized CL and `FedAvg`. Compared with standard supervised learning, `FedMoCo` utilizes only 3% of labels to achieve 90% accuracy, which greatly reduces the annotation cost.

4 Conclusion

In this work, we formulate and discuss FCL on medical images and propose `FedMoCo`. Limited by space, we evaluate the robustness of `FedMoCo` under a few characteristic non-IID challenges and use COVID-19 detection as the downstream task. More investigations will be conducted, but the initial results already provide insights into future research on FCL. In future, we plan to focus on the task affinity [24] between FCL and corresponding downstream tasks. We will quantitatively analyze how the representations extracted by FCL will influence the performance of different downstream tasks.

References

1. Baldi, P., Pineda, F.: Contrastive learning and neural oscillations. Neural Computation **3**(4), 526–545 (1991)
2. Box, G.E., Cox, D.R.: An analysis of transformations. Journal of the Royal Statistical Society: Series B (Methodological) **26**(2), 211–243 (1964)
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: ICML (2020)
4. 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 p. 107826
5. Chowdhury, M.E., Rahman, T., Khandakar, A., Mazhar, R., Kadir, M.A., Mahbub, Z.B., Islam, K.R., Khan, M.S., Iqbal, A., Al Emadi, N., et al.: Can ai help in screening viral and covid-19 pneumonia? IEEE Access **8**, 132665–132676 (2020)
6. Dong, N., Kampffmeyer, M., Liang, X., Wang, Z., Dai, W., Xing, E.: Unsupervised domain adaptation for automatic estimation of cardiothoracic ratio. In: MICCAI. pp. 544–552. Springer (2018)
7. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: CVPR. pp. 9729–9738 (2020)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
9. He, X., Yang, X., Zhang, S., Zhao, J., Zhang, Y., Xing, E., Xie, P.: Sample-efficient deep learning for covid-19 diagnosis based on ct scans. MedRxiv (2020)
10. Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., et al.: Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In: AAAI. vol. 33, pp. 590–597 (2019)
11. Kriegeskorte, N., Mur, M., Bandettini, P.A.: Representational similarity analysis-connecting the branches of systems neuroscience. Frontiers in systems neuroscience **2**, 4 (2008)
12. Li, M., Andersen, D.G., Smola, A.J., Yu, K.: Communication efficient distributed machine learning with the parameter server. In: NIPS. pp. 19–27 (2014)
13. Li, W., Milletarì, F., Xu, D., Rieke, N., Hancox, J., Zhu, W., Baust, M., Cheng, Y., Ourselin, S., Cardoso, M.J., et al.: Privacy-preserving federated brain tumour segmentation. In: International Workshop on Machine Learning in Medical Imaging. pp. 133–141. Springer (2019)
14. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: AISTATS. pp. 1273–1282. PMLR (2017)
15. Misra, I., Maaten, L.v.d.: Self-supervised learning of pretext-invariant representations. In: CVPR. pp. 6707–6717 (2020)
16. Nguyen, H.Q., Lam, K., Le, L.T., Pham, H.H., Tran, D.Q., Nguyen, D.B., Le, D.D., Pham, C.M., Tong, H.T., Dinh, D.H., et al.: Vindr-cxr: An open dataset of chest x-rays with radiologist's annotations. arXiv preprint arXiv:2012.15029 (2020)
17. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
18. Pizer, S.M., Amburn, E.P., Austin, J.D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J.B., Zuiderveld, K.: Adaptive histogram equalization and its variations. Computer vision, graphics, and image processing **39**(3), 355–368 (1987)

19. Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H.R., Albarqouni, S., Bakas, S., Galtier, M.N., Landman, B.A., Maier-Hein, K., et al.: The future of digital health with federated learning. NPJ digital medicine **3**(1), 1–7 (2020)
20. de Sa, V.R.: Learning classification with unlabeled data. In: NIPS. pp. 112–119. Citeseer (1994)
21. Sheller, M.J., Reina, G.A., Edwards, B., Martin, J., Bakas, S.: Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In: International MICCAI Brainlesion Workshop. pp. 92–104. Springer (2018)
22. Sowrirajan, H., Yang, J., Ng, A.Y., Rajpurkar, P.: Moco pretraining improves representation and transferability of chest x-ray models. arXiv preprint arXiv:2010.05352 (2020)
23. Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., Isola, P.: What makes for good views for contrastive learning. In: NIPS (2020)
24. Vandenhende, S., Georgoulis, S., De Brabandere, B., Van Gool, L.: Branched multi-task networks: deciding what layers to share. In: BMVC (2020)
25. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: CVPR. pp. 2097–2106 (2017)
26. Wu, Z., Xiong, Y., Stella, X.Y., Lin, D.: Unsupervised feature learning via non-parametric instance discrimination. In: CVPR (2018)