

UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

Improve the performance of CT-based pneumonia classification via source data reweighting

Permalink

<https://escholarship.org/uc/item/38z7c377>

Journal

Scientific Reports, 13(1)

ISSN

2045-2322

Authors

Xie, Pengtao

Zhao, Xingchen

He, Xuehai

Publication Date

2023

DOI

10.1038/s41598-023-35938-3

Copyright Information

This work is made available under the terms of a Creative Commons Attribution License, available at <https://creativecommons.org/licenses/by/4.0/>

Peer reviewed



OPEN

Improve the performance of CT-based pneumonia classification via source data reweighting

Pengtao Xie^{1✉}, Xingchen Zhao^{2,4} & Xuehai He^{3,4}

Pneumonia is a life-threatening disease. Computer tomography (CT) imaging is broadly used for diagnosing pneumonia. To assist radiologists in accurately and efficiently detecting pneumonia from CT scans, many deep learning methods have been developed. These methods require large amounts of annotated CT scans, which are difficult to obtain due to privacy concerns and high annotation costs. To address this problem, we develop a three-level optimization based method which leverages CT data from a source domain to mitigate the lack of labeled CT scans in a target domain. Our method automatically identifies and downweights low-quality source CT data examples which are noisy or have large domain discrepancy with target data, by minimizing the validation loss of a target model trained on reweighted source data. On a target dataset with 2218 CT scans and a source dataset with 349 CT images, our method achieves an F1 score of 91.8% in detecting pneumonia and an F1 score of 92.4% in detecting other types of pneumonia, which are significantly better than those achieved by state-of-the-art baseline methods.

Pneumonia is a life-threatening disease caused by bacteria, virus, and fungi. It is a type of acute respiratory infection in the lungs. According to the World Health Organization, pneumonia caused about 0.74 million deaths of children in 2019, which accounts for 14% of all deaths of children under 5 years old¹. In particular, the recent COVID-19 pneumonia caused 6.5 millions deaths globally. Chest computed tomography (CT) scans are broadly used to diagnose pneumonia (including COVID-19 and other types of pneumonia)²⁻⁴, differentiate different types of pneumonia⁵, assess the severity of pneumonia⁶, etc. In medically under-served areas such as rural areas, well-trained radiologists who can accurately interpret CT scans to detect and assess the severity of pneumonia are lacking. To assist radiologists in accurately and efficiently detecting pneumonia and distinguishing different types of pneumonia from CT scans, many deep learning methods have been developed⁷⁻²². For example, Qian et al.⁷ proposed a multi-task multi-slice deep neural network to screen pneumonia from CT scans. Abdel-Basset¹² developed a two-stage deep learning method to distinguish community-acquired pneumonia from COVID-19 pneumonia based on CT scans. Ortiz²³ developed a deep neural network which leverages CT scans and clinical metadata to distinguish COVID-19 pneumonia from other viral pneumonia.

Deep neural networks typically have a large number of weight parameters. To effectively train them, large amounts of CT scans with pneumonia annotations are needed. Due to data privacy concerns and high costs of annotating pneumonia, it is very difficult to obtain a large number of annotated CT scans. Without sufficient training data, deep neural networks perform unsatisfactorily on test cases. One way of addressing this problem is transfer learning (TL)²⁴, which leverages data from source domains to help train a target model in a target domain. It is often the case that some source examples have low quality and should be down-weighted. For example, some source examples are noisy and some examples have large domain discrepancy with target data. It is important to automatically identify such low-quality source examples and down-weight them during the transfer learning process. Many methods²⁵⁻³³ have been developed for source example reweighting. These methods do not learn weights of source examples by maximizing the performance of the target model on a held-out validation set. As a result, the target model trained (implicitly) using reweighted source examples is not guaranteed to have good generalization performance on test data. Bi-level optimization (BLO)³⁴ based

¹Department of Electrical and Computer Engineering, University of California San Diego, San Diego, USA. ²Department of Electrical and Computer Engineering, Northeastern University, Boston, USA. ³Department of Computer Science and Engineering, University of California Santa Cruz, Santa Cruz, USA. ⁴These authors contributed equally: Xingchen Zhao and Xuehai He. ✉email: p1xie@eng.ucsd.edu

approaches^{35–40} have been proposed for data reweighting by explicitly minimizing a validation loss, where the first-level trains network weights on a training dataset and the second level learns data weights on a validation set. Applying these approaches for source data reweighting in TL, it is required to train source model and target model in a multi-task learning (MTL) way at the first level. As noted by Zhuang et al.²⁴, MTL formulations are not suitable for TL since MTL focuses on training both source and target models well while the goal of TL is to train the target model well.

To address the problems of existing methods, we propose a three-level optimization based method which performs three learning stages end-to-end. At the first stage, a source model is trained by minimizing weighted losses on source examples. Source example weights are tentatively fixed at this stage. At the second stage, we train a target model by transferring knowledge from source to target. We propose a new ranking-based knowledge transfer approach that allows source model and target model to have different architectures, different dimensions of encodings, different weight parameters, etc. Our method uses the source model to generate a ranking on target examples. Then we train the target model by letting it predict the generated ranking. At the third stage, we evaluate the trained target model on a validation set. Note the validation loss is a function of source example weights. We update weights of source examples by minimizing the validation loss.

Compared with existing methods, our method has the following advantages: (1) our method learns source example weights by explicitly minimizing the validation loss of the target model, which therefore can make the target model generalize well on test data; (2) our method focuses on improving the target model by transferring knowledge from source to target, instead of training source and target models in a multi-task learning framework.

We apply the proposed method to detect pneumonia and distinguish COVID-19 and other types of pneumonia, on a target dataset with 2218 CT scans and a source dataset with 349 CT images. Our method achieves an F1 score of 91.8% for COVID-19 pneumonia and an F1 score of 92.4% for other types of pneumonia. Our method outperforms state-of-the-art baseline methods.

The major contributions of this paper are as follows:

- To accurately detect pneumonia from CT scans and distinguish COVID-19 pneumonia and other types of pneumonia, we propose a three-level optimization based method which leverages external CT data from a source domain to help train a target model and automatically down-weights low-quality source data examples. Our framework learns source example weights by explicitly minimizing the validation loss of the target model and performs knowledge transfer from source to target instead of training source and target models simultaneously in a multi-task learning way.
- We propose a new ranking-based knowledge transfer approach where a source model generates a ranking and a target model predicts this ranking.
- We demonstrate the effectiveness of the proposed method in detecting pneumonia from CT scans and differentiating COVID-19 pneumonia and other types of pneumonia. By leveraging a source dataset with 349 CT images, our method achieves an F1 score of 91.8% for COVID-19 and an F1 score of 92.4% for other types of pneumonia on a target dataset with 2218 CT scans. Our method outperforms state-of-the-art baseline methods significantly.

Related works

Deep learning for detecting pneumonia from CT scans. Many deep learning methods have been developed for detecting pneumonia from CT scans. Qian et al.⁷ proposed a multi-task multi-slice deep neural network to screen pneumonia from CT scans. Li et al.⁸ developed a deep learning method to analyze thick-section CT scans for assessing the severity and progression of COVID-19. Amyar et al.⁹ proposed a multi-task deep learning method to identify and segment COVID-19 from CT scans. Ni et al.¹⁰ developed a deep neural network for COVID-19 pneumonia classification, segmentation, and localization from CT scans. Zhang et al.⁴¹ leveraged a human-in-the-loop training strategy to learn a segmentation network for quantifying the volumes of COVID-19 infection on CT scans. Ko et al.⁴² proposed a transfer learning method which leverages pretrained convolutional neural networks to detect COVID-19 pneumonia from a single chest CT image. Maghdid et al.⁴³ used pretrained deep convolution network to detect pneumonia from CT scans and chest X-rays. Xu et al.¹¹ developed a 3D deep convolutional network model to distinguish COVID-19 from influenza-A viral pneumonia and healthy cases from CT scans. Abdel-Basset¹² developed a two-stage deep learning method to distinguish community-acquired pneumonia from COVID-19 pneumonia based on CT scans. Chen⁴⁴ applied UNet++ to detect COVID pneumonia from CT scans. Zhou et al.⁴⁵ applied YOLOv3 to differentiate novel coronavirus pneumonia from influenza pneumonia based on CT scans. Chaudhary et al.⁴⁶ developed a two-stage convolutional neural network (CNN) to detect COVID-19 and community acquired pneumonia (CAP) from CT scans. Bermejo-Peláez et al.⁴⁷ proposed a deep neural network to analyze COVID-19 patterns from CT scans to assess disease severity and predict clinical outcomes. Yao et al.⁴⁸ developed an atrous convolution network to diagnose mild COVID-19 pneumonia from CT scans. Song et al.⁴⁹ developed a Details Relation Extraction neural network to diagnose COVID-19 from CT images. Bratt et al.⁵⁰ developed a deep learning method to predict usual interstitial pneumonia histopathology from CT images. Shiri et al.⁵¹ developed a deep neural network to assess the severity of COVID-19 based on CT radiomics features. Ortiz²³ developed a deep neural network which leverages CT scans and clinical metadata to distinguish COVID-19 pneumonia from other viral pneumonia. Existing methods do not consider leveraging external CT data from source domains to mitigate the lack of CT scans that have pneumonia labels or cannot automatically identify and downweight source CT data examples that are noisy and have large domain discrepancy with target domain. Our method bridges these gaps.

Source data reweighting in transfer learning. Many methods^{25–33,52–56} have been developed to reweight source data examples according to their fitness for training target models, based on bi-level optimization^{35,37–40}, reinforcement learning^{57–59}, adversarial learning³¹, curriculum learning^{27,32}, entropy^{60,61}, Bayesian optimization²⁶, multi-task learning²⁵, etc. In bi-level optimization based methods, an inner optimization problem trains a model on reweighted data and an outer optimization problem learns data weights by minimizing validation loss of the trained model. These methods reweight source data by comparing their similarity with training data in the target domain. As a result, a target model trained using these reweighted source data may overfit to the training data distribution and generalizes poorly on unseen data. Our method reweights source examples by measuring how a target model trained (implicitly) using reweighted source data generalizes to validation examples, and therefore is more robust to overfitting.

Transfer learning and multi-task learning. The goal of transfer learning (TL)^{24,62–66} is to leverage data in a source domain to help with model training in a target domain. Existing TL methods can be roughly categorized into the following groups: (1) latent space projection^{67–70}, (2) distribution alignment^{53,71–73}, (3) adversarial domain-invariant representation learning^{74–77}, and (4) regularization^{78–80}. Multi-task learning (MTL)^{81,82} aims to improve multiple models simultaneously by training them jointly and transferring knowledge across models. Various MTL approaches have been proposed, based on (1) hard parameter sharing^{83–87}, where multiple models share the same weight parameters, such as encoder weights; (2) soft parameter sharing⁸⁸, where parameters of different models are constrained to be similar; (3) task similarity learning^{89–91}, which identifies similarity between tasks and encourages similar tasks to share more commonalities; (4) loss weighting⁸⁷, which weighs each model's loss, and so on. Different from previous transfer learning and multi-task learning methods, our method is based on three-level optimization and can automatically identify and downweight source data that is noisy or have large domain discrepancy with target data by minimizing target model's validation loss.

Bi-level optimization. Bi-level Optimization (BLO)⁹² has been broadly applied for hyperparameter tuning⁹³, neural architecture search⁹⁴, meta learning⁹⁵, data reweighting^{37–39}, learning rate adjustment⁹⁶, label denoising⁹⁷, data generation⁹⁸, etc. In these methods, meta parameters (e.g. hyperparameters, neural architectures, data weights, etc.) are optimized by minimizing validation losses and model weights are learned by minimizing training losses. Our method goes beyond bi-level optimization and solves a three-level optimization problem for source data reweighting.

Methods

In this section, we present the method for reweighting source CT data based on three-level optimization. We aim to train a target model M_t to detect pneumonia from CT scans, on a dataset D_t from the target domain which contains CT scans with pneumonia class labels. To mitigate the deficiency of labeled target data, we leverage a CT dataset D_s from a source domain which has pneumonia class labels. A source model M_s is trained on D_s . Some examples in D_s are noisy and some examples have large domain discrepancies with D_t . We aim to downweight such low-quality source examples by automatically learning a weight for each source example. For M_t , it has an encoder E_t and a head H_t . For M_s , it has an encoder E_s and a head H_s . Knowledge transfer is conducted from E_s to E_t . Note that we allow E_s and E_t to have different architectures, different dimensions of encodings, and different weight parameters.

A three-level optimization framework. We propose a three-level optimization based framework (Fig. 1(top)) to perform reweighting of source CT data. The framework consists of three learning stages which are performed end-to-end. At the first stage, we train E_s and H_s on D_s . For each source example in $D_s = \{d_{s,i}\}_{i=1}^M$, an importance weight $c \in [0, 1]$ is to be learned. The training loss L of a source example, which is a cross-entropy based classification loss, is multiplied with the importance weight of this example³⁵. If c is close to 0, it means this example is noisy or has large domain-discrepancy with target data; accordingly, the loss (after multiplied with c) is made close to 0, which effectively excludes this example from the training process. We aim to automatically learn these importance weights, which will be detailed later on. This stage amounts to solving the following problem:

$$E_s^*(C), H_s^*(C) = \operatorname{argmin}_{E_s, H_s} \sum_{i=1}^M c_i L(E_s, H_s, d_{s,i}), \quad (1)$$

where $C = \{c_i\}_{i=1}^M$. The importance weights C are needed to calculate training losses, but they should not be updated at this stage. Otherwise, the values of C will all be zero. Note that $E_s^*(C)$ depends on C since it depends on the training loss which is a function of C .

At the second stage, we train E_t and H_t on D_t by minimizing a cross-entropy based classification loss L . We propose a novel way of transferring knowledge from source model to target model, based on predicting cross-domain relative similarity relationships. Given two target examples, we use the source encoder to label which of them is closer to the source examples. For each target example x_t , its distance to source dataset D_s is calculated as $e(x_t, D_s; E_s^*(C)) = \min_{d_s \in D_s} c(x_t, d_s; E_s^*(C))$, where $c(x_t, d_s; E_s^*(C))$ is the L2 distance between x_t and a source example d_s , both encoded by the source encoder $E_s^*(C)$. For two target examples x_t and y_t , if $e(x_t, D_s; E_s^*(C)) < e(y_t, D_s; E_s^*(C))$, x_t is labeled as being closer to the source dataset (denoted by $x_t \succ y_t | E_s^*(C)$). The source encoder labels many such pairs. Then these labeled pairs are used to train the target encoder. Given a cross-domain relative similarity (CDRS) relationship between two target examples, the target encoder is trained to predict this relationship. Let $e(x_t, D_s; E_t)$ (defined in a similar way as $e(x_t, D_s; E_s^*(C))$) denotes the distance

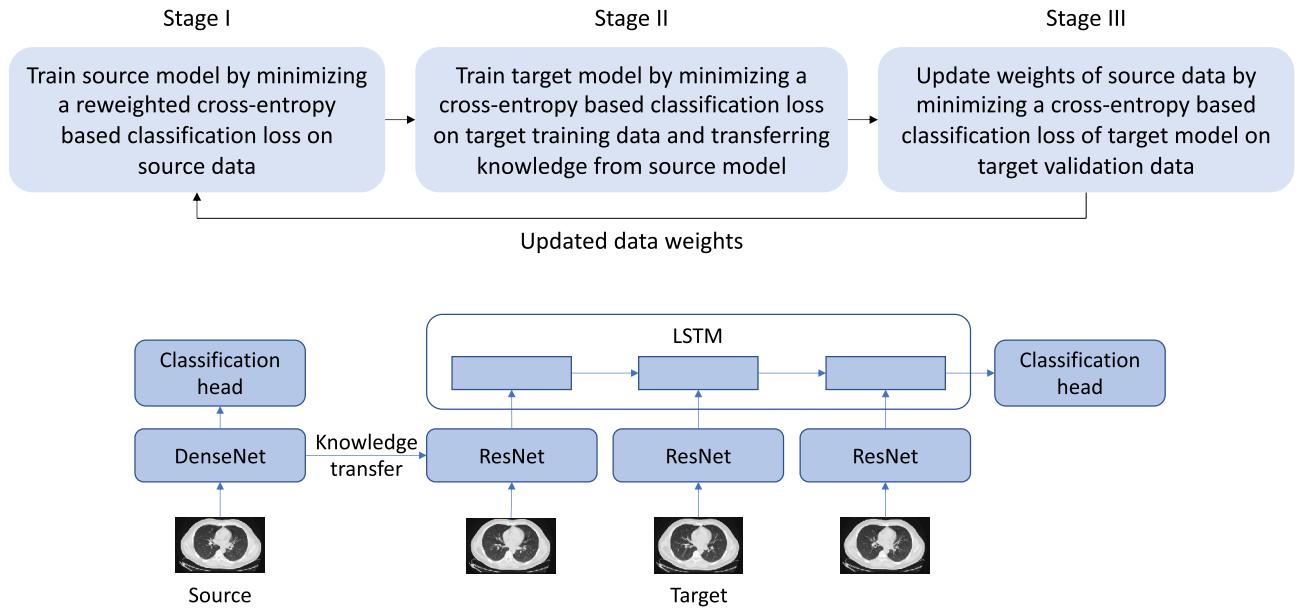


Figure 1. (Top) Overview of the proposed three-level optimization framework. (Bottom) Network architecture of target and source models.

between x_t and D_s , encoded using the target encoder. If $x_t > y_t | E_s^*(C)$, then $e(x_t, D_s; E_t)$ is required to be smaller than $e(y_t, D_s; E_t)$. The optimization problem at the second stage is:

$$E_t^*(E_s^*(C)), H_t^* = \operatorname{argmin}_{E_t, H_t} L(E_t, H_t, D_t) \\ \text{s.t. } \forall x_t > y_t | E_s^*(C), e(x_t, D_s; E_t) < e(y_t, D_s; E_t)$$

CDRS captures the relationships among target examples by generating a global ranking among them, based on their distances to the source domain (such a global ranking can be induced from all pairwise rankings). This global ranking reveals a lot of semantic information of target CT images. For instance, CT images at adjacent positions in the ranking are more similar than those at faraway positions. Grouping similar CT images together using the ranking can facilitate pneumonia classification. In pneumonia classification, there are multiple classes, such as common pneumonia, COVID-19, and no pneumonia. Grouping CT images in the same pneumonia class together based on their positions in the global ranking can better distinguish different classes and improve classification performance.

At the third stage, we use the trained target model consisting of $E_t^*(E_s^*(C))$ and H_t^* to make predictions on a validation dataset $D_t^{(val)}$ in the target domain. We update C by minimizing the validation loss L which is also a cross-entropy based classification loss.

$$\min_C L(E_t^*(E_s^*(C)), H_t^*, D_t^{(val)}) \tag{2}$$

Putting the three learning stages together, we have the following three-level optimization framework:

$$\min_C L(E_t^*(E_s^*(C)), H_t^*, D_t^{(val)}) \\ \text{s.t. } E_t^*(E_s^*(C)), H_t^* = \operatorname{argmin}_{E_t, H_t} L(E_t, H_t, D_t) \\ \text{s.t. } \forall x_t > y_t | E_s^*(C), e(x_t, D_s; E_t) < e(y_t, D_s; E_t) \\ \text{s.t. } E_s^*(C), H_s^*(C) = \operatorname{argmin}_{E_s, H_s} \sum_{i=1}^M c_i L(E_s, H_s, d_{s,i}) \tag{3}$$

The three stages are mutually dependent on each other and are conducted end-to-end. The output of Stage I, which is $E_s^*(C)$, is used as input of Stage II. The outputs of Stage II, which are $E_t^*(E_s^*(C))$ and H_t^* , are used as inputs of Stage III. After C is updated at Stage III, the objective function in Stage I, which is a function of C , will change accordingly, rendering $E_s^*(C)$ to change as well. By solving the three interdependent optimization problems jointly in the multi-level optimization framework, we can perform the three learning stages end-to-end.

The constraint in the above equation is highly discrete, which is not end-to-end differentiable and renders the optimization problem difficult to solve. We address this issue by performing a continuous relaxation of the constraint using pairwise hinge loss:

$$\lambda \max \left(0, - (e(x_t, D_s; E_s^*(C)) - e(y_t, D_s; E_s^*(C))) (e(x_t, D_s; E_t) - e(y_t, D_s; E_t)) \right), \tag{4}$$

where λ is a tradeoff parameter. Let $p = e(x_t, D_s; E_s^*(C)) - e(y_t, D_s; E_s^*(C))$ and $q = e(x_t, D_s; E_t) - e(y_t, D_s; E_t)$. If $pq > 0$, the target model predicts the CDRS relationship between x_t and y_t correctly. In this case, the hinge loss is 0 and there is no penalty. Otherwise, the hinge loss is $-pq$, which penalizes the target model to correct its prediction.

Optimization algorithm. We develop a gradient-based algorithm to solve the three-level optimization problem. Drawing insights from⁹⁴, we approximate $E_s^*(C)$ using one-step gradient descent update of E_s :

$$E_s^*(C) \approx E_s' = E_s - \eta_{es} \nabla_{E_s} \sum_{i=1}^M c_i L(E_s, H_s, d_{s,i}), \quad (5)$$

and update H_s using:

$$H_s \leftarrow H_s - \eta_{hs} \nabla_{H_s} \sum_{i=1}^M c_i L(E_s, H_s, d_{s,i}). \quad (6)$$

We plug $E_s^*(C) \approx E_s'$ into the loss function at the second stage and get an approximated objective:

$$O = L(E_t, H_t, D_t) + \lambda \sum_{x_t, y_t} \max(0, -(e(x_t, D_s; E_s') - e(y_t, D_s; E_s'))(e(x_t, D_s; E_t) - e(y_t, D_s; E_t))). \quad (7)$$

Then we approximate $E_t^*(E_s^*(C))$ and H_t^* using one-step gradient descent update of E_t and H_t w.r.t O :

$$E_t^*(E_s^*(C)) \approx E_t' = E_t - \eta_{et} \nabla_{E_t} O, \quad (8)$$

$$H_t^* \approx H_t' = H_t - \eta_{ht} \nabla_{H_t} O. \quad (9)$$

Finally, we plug these approximations into the validation loss at the third stage and update C and R_s by minimizing the approximated loss using gradient descent:

$$C \leftarrow C - \eta_c \nabla_C L(E_t', H_t', D_t^{(val)}). \quad (10)$$

For $\nabla_C L(E_t', H_t', D_t^{(val)})$, it can be computed as:

$$\nabla_C L(E_t', H_t', D_t^{(val)}) = \frac{\partial E_s'}{\partial C} \frac{\partial E_t'}{\partial E_s'} \frac{\partial L(E_t', H_t', D_t^{(val)})}{\partial E_t'}, \quad (11)$$

where

$$\frac{\partial E_t'}{\partial E_s'} = -\eta_{et} \nabla_{E_s', E_t}^2 O, \quad (12)$$

$$\frac{\partial E_s'}{\partial C} = -\eta_{es} \nabla_{C, E_s}^2 \sum_{i=1}^M c_i L(E_s, H_s, d_{s,i}). \quad (13)$$

These steps iterate until convergence. For the calculation of $e(x_t, D_s; E_s^*(C))$, in theory, the complexity is quadratic $O(n^2)$, where n is the number of data examples: calculating the distance between each pair of (source, target) examples. In practice, the calculation is conducted on a minibatch of target examples and a minibatch of source examples. The actual complexity is $O(k^2)$, where k is the minibatch size. k is at most a few hundred, which is much smaller than n . Similar to DARTS⁹⁴, the matrix-vector multiplication in Eq. (11) are approximated using finite differences, which can be calculated efficiently. With this approximation, the complexity is reduced from quadratic to linear (in terms of parameter numbers).

Dataset

For the target domain dataset, we used the China Consortium of Chest CT Image Investigation (CC-CCII)⁹⁹. It contains 2218 3D CT scans from 557 common pneumonia patients (CP), 682 COVID-19 patients (NCP), and 979 normal controls. The common pneumonia (CP) group includes viral pneumonia (e.g., adenoviral, influenza, and parainfluenza pneumonia), bacterial pneumonia, and mycoplasma pneumonia, which were diagnosed based on standard clinical, radiological, culture/molecular assay results. Novel coronavirus patient (NCP) diagnosis was based on reverse transcriptase PCR. The CTs scans are obtained from Sun Yat-sen Memorial Hospital, Third Affiliated Hospital of Sun Yat-sen University, The first Affiliated Hospital of Anhui Medical University, West China Hospital, Nanjing Renmin Hospital, Yichang Central People's Hospital, and Renmin Hospital of Wuhan University. The dataset is split into a train, validation, and test set with a ratio of 7:1.5:1.5. For the source domain dataset, we use the one collected by Yang et al.¹⁰⁰. This dataset contains 2D CT slices extracted from COVID-19 related papers. Each slice is labeled with whether containing COVID-19 findings. It consists of 349 COVID-19 CT images from 216 patients and 463 non-COVID-19 CTs. These CT slices are extracted from 760 preprints about COVID-19 in medRxiv and bioRxiv, posted from Jan 19th to Mar 25th. The minimum, average, and maximum height of these images are 153, 491, and 1853. The minimum, average, and maximum width of these images are 124, 383, and 1485. In this dataset, some images are noisy, annotated with non-clinical artifacts

such as bounding boxes and arrows. Some images have large domain differences from target data in terms of resolution, appearance, texture, color, scale, etc.

Experiments

In this section, we present experimental results.

Experimental settings. We leverage the proposed three-level optimization framework to reweight the noisy source CT data. The source model performs binary classification on 2D CT slices in the source dataset: given a 2D CT slice, predict whether it contains COVID-19. The source model consists of a 2D image encoder E_s and a classification head H_s . The target model performs three-class classification on 3D CT scans in the target dataset, which classifies a 3D CT scan into one of the three classes: NPC, CP, and Normal. The target model consists of an LSTM-CNN encoder¹⁰¹ and a classification head H_t . The LSTM-CNN encoder is used to encode a 3D CT scan, which contains a sequence of 2D CT slices. For each 2D slice, it is encoded by a CNN encoder E_t . Then the sequence of CNN encodings are fed into an LSTM¹⁰² network V_t to extract a holistic representation of the entire 3D CT scan. The encoding of the 3D CT scan is fed into the classification head to predict the class label. When training E_t , we transfer knowledge from E_s to E_t .

In the LSTM-CNN target encoder, we set the hidden size to 128 and set the CNN encoder to ResNet-50¹⁰³. For the source encoder, we set it to DenseNet¹⁰⁴. Dimensions of embeddings generated by the source and target CNN encoders are different. The classification heads H_t and H_s are linear. Figure 1 (bottom) shows the network architecture of source and target models. The tradeoff parameter λ was set to 0.1. The initial learning rate was set to $1e-3$ with the Adam¹⁰⁵ optimizer used. The learning rate decayed with cosine scheduling. The momentum for Adam was (0.5, 0.999). The batch size was set to 64. The model was trained for 150 epochs. The dropout¹⁰⁶ rate was set to 0.3. Weight decay was set to $5e-4$. To determine the convergence of the three-level optimization problem, we check the values of the validation loss in the third stage and those of training losses in the first and second stage. For every loss, if the standard deviation of its values in the most recent four epochs is less than 5% of these values' mean, we consider the algorithm has converged and stop the optimization process.

To tune the hyperparameters, we randomly split the validation set into two equal-sized subsets: denoted by A and B . For each configuration of hyperparameters, we used the validation set A to learn importance weights of source data. Then we measure the performance of the trained model on validation set B . Hyperparameter values yielding the best performance on validation set B were selected. To ensure a fair comparison, we spent approximately the same time on tuning hyperparameters for each method (including ours and baselines). The tuning time per method is about 16 hours.

Every experiment ran 5 times with different random initializations. Precision, recall, and F1 scores were used as evaluation metrics. For all experiments, we performed significance tests using double-sided t-tests. The p-values of our methods against baselines are all less than 0.001, which shows that our methods are significantly better than baselines. The experiments were conducted on A100 GPU. Our method takes about 26 hours to converge.

Baselines. We compared our method with the following baselines. The baseline models were trained on the combination of training and validation sets of the target dataset.

- No source data (NoSrc)¹⁰¹: we do not leverage source data for model training.
- No reweighting (NoWt)¹⁰¹: we use all source data examples for model training without reweighting. NoWt uses CDRS-based knowledge transfer. We first train a source model on all source examples without reweighting, then use this source model to label CDRS relationships. These labeled CDRS relationships are utilized as constraints to train the target model.
- Pretrain¹⁰³: we first pretrain E_s on source data, then use E_s to initialize E_t .
- BO²⁶: Bayesian optimization based data selection for transfer learning.
- MGTL³¹: a minimax game based model for selective transfer learning.
- Online meta-learning (OML)³⁵ for data reweighting: we first unify the formats of 3D CT scans and 2D source CT slices by labeling all slices in a positive/negative CT scan as positive/negative; then on 2D slices, we use an online meta-learning³⁵ approach, which is based on bi-level optimization (BLO), to reweight source slices.
- MentorNet¹⁰⁷: a curriculum learning method for data selection. Similar to OML, format unification is applied.
- Multi-task learning (MTL)¹⁰⁸: BLO is applied for reweighting source data, where source and target models are trained simultaneously by minimizing the weighted sum of their training losses; source and target encoder weights are encouraged to be similar using L2 regularization to transfer knowledge between source and target encoders and let them help each other to learn.
- Weights sharing (WS)⁸⁶: similar to MTL, except that target and source models share the same encoder.
- We compare our CDRS knowledge transfer approach with: (1) L2 regularization on encoder weights (RegW)¹⁰⁹: encouraging target encoder's weights to have small L2 distance with source encoder's weights; (2) L2 regularization on embeddings (RegE)¹¹⁰: encouraging embeddings generated by target encoder to have small L2 distance with those generated by source encoder; (3) pseudo-labeling (PL)¹¹¹: the source model generates pseudo-labels regarding whether CT slices are positive, which are used to train the target model; and (4) pairwise similarity (PS)¹¹²: the source encoder annotates whether two images are similar or dissimilar and the target encoder predicts these similarity labels.
- We also compare with six CT-based pneumonia classification methods: (1) CC-CCII⁹⁹, (2) RapidAI¹¹³, (3) 3DCNN¹¹, (4) Li et al.¹¹⁴, (5) Shamsi et al.¹¹⁵, and (6) Shaik et al.¹¹⁶.

Results and analysis. Table 1 shows the results. Our method achieves an F1 score of 91.8% in detecting COVID-19 pneumonia (NCP) and an F1 score of 92.4% in detecting other types of pneumonia (CP). Our method performs better than NoSrc. In NoSrc, no source data is leveraged for learning representations, which is a waste. Many source 2D CT slices contain abnormalities related to COVID-19. An encoder trained using these CT slices can learn representations capturing such abnormalities, which is helpful for classifying COVID-19. Our method performs better than NoWt. In NoWt, all source examples are used without reweighting. Many source examples are noisy or incorrectly labeled. Trained using such low-quality data, the source encoder E_s may learn poor representations. Transferring low-quality representations from E_s to E_t , the effectiveness of E_t may be degraded as well, which yields inferior performance in classifying 3D CT scans. In contrast, our method performs reweighting of each source data example by checking whether it can help to reduce the validation loss of the target model. If a source example hurts classification performance on 3D CT scans, our method automatically assigns a small weight to it. Our method outperforms state-of-the-art methods developed for pneumonia classification, including CC-CCII⁹⁹, RapidAI¹¹³, 3DCNN¹¹, Li et al.¹¹⁴, Shamsi et al.¹¹⁵, and Shaik et al.¹¹⁶. The reason is that these methods do not leverage auxiliary source data or lack capability of reweighting source data.

Our method outperforms MTL. In MTL, target model and source model are trained simultaneously by minimizing the weighted sum of their loss functions. In our experiments, we found that these two models have a competing relationship during training: improving performance of one model incurs performance degradation of the other model. Figure 2(left) shows how test performances of source and target models vary with the weight β of source model's training loss (the weight of target model's training loss is set to 1). The source model is evaluated on a clean 2D CT test set. As β increases (more attention is paid to minimizing the source model's loss), F1 of source model increases while F1 of target model decreases. This demonstrates that MTL incurs a competition between the two models. Our method addresses this problem by training these two models in two different optimization problems in an end-to-end framework. We first train the source model, then leverage transfer learning to train the target model. In this way, the source model helps the target model to learn, instead of competing with it. We provide some empirical evidence by doing the following experiments. First, we train three source encoders that have different accuracy. Three clean 2D CT validation sets with increasing example numbers and one clean 2D test set are collected. Using each validation set, we learn training data weights based on the method proposed by Ren et al.³⁵ and train a model accordingly. Using these learned source models as regularization, we train three target models. Figure 2(right) shows that the performance of target models increases

	NCP			CP			Normal		
	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)
CC-CCII ⁹⁹	84.5 ₅	87.2 ₇	85.8 ₄	88.9 ₈	87.5 ₆	88.2 ₂	85.9 ₆	86.3 ₇	86.1 ₇
RapidAI ¹¹³	83.1 ₂	86.8 ₅	84.9 ₁	86.3 ₅	86.0 ₄	86.1 ₃	83.8 ₈	85.5 ₆	84.6 ₈
3D-CNN ¹¹	86.6 ₇	87.2 ₅	86.9 ₅	85.4 ₇	87.7 ₉	86.5 ₇	86.8 ₅	85.9 ₉	86.3 ₂
Li et al. ¹¹⁴	84.2 ₃	88.1 ₆	86.1 ₄	89.4 ₄	88.5 ₃	88.9 ₃	86.6 ₇	87.2 ₄	86.9 ₅
Shamsi et al. ¹¹⁵	85.4 ₅	87.9 ₄	86.6 ₅	90.1 ₆	89.1 ₆	89.6 ₅	88.5 ₈	88.9 ₅	88.7 ₇
Shaik et al. ¹¹⁶	84.9 ₃	88.7 ₉	86.8 ₅	88.2 ₈	88.3 ₂	88.2 ₆	87.2 ₃	87.4 ₅	87.3 ₅
NoSrc ¹⁰¹	85.9 ₉	87.5 ₆	86.7 ₆	88.1 ₉	89.2 ₆	88.6 ₉	86.2 ₉	87.1 ₉	86.6 ₅
NoWt ¹⁰¹	86.4 ₇	87.1 ₄	86.7 ₄	88.6 ₄	89.8 ₅	89.2 ₇	88.1 ₃	88.5 ₈	88.3 ₂
Pretrain ¹⁰³	88.6 ₄	90.1 ₈	89.3 ₃	89.1 ₅	90.9 ₄	90.0 ₂	88.6 ₇	87.2 ₅	87.9 ₇
BO ²⁶	87.3 ₆	88.5 ₃	87.9 ₈	89.2 ₉	90.5 ₈	89.8 ₆	89.2 ₂	88.5 ₆	88.8 ₄
MGTL ³¹	87.5 ₅	90.7 ₆	89.1 ₆	90.2 ₇	91.7 ₃	90.9 ₅	88.5 ₆	89.3 ₄	88.9 ₉
OML ³⁵	87.1 ₈	88.4 ₈	87.7 ₈	89.5 ₄	89.7 ₉	89.6 ₅	88.1 ₄	88.3 ₈	88.2 ₁
MentorNet ¹⁰⁷	87.6 ₆	88.2 ₂	87.9 ₄	89.8 ₆	90.3 ₇	90.0 ₇	88.4 ₄	89.1 ₅	88.7 ₅
MTL ¹⁰⁸	89.1 ₄	90.7 ₇	89.9 ₇	90.9 ₅	91.2 ₄	91.0 ₅	88.2 ₆	88.7 ₇	88.4 ₈
WS ⁸⁶	89.2 ₃	89.7 ₇	89.4 ₄	90.3 ₆	91.1 ₅	90.7 ₈	87.3 ₇	88.1 ₈	87.7 ₂
No-LSTM	90.0 ₄	91.4 ₆	90.7 ₅	91.3 ₃	92.0 ₄	90.2 ₄	89.2 ₅	90.3 ₃	90.1 ₅
Same-ResNet	90.5 ₂	89.9 ₉	91.6 ₅	90.8 ₉	91.4 ₅	91.1 ₈	89.4 ₇	90.1 ₄	90.9 ₅
Same-DenseNet	90.3 ₅	90.0 ₆	89.7 ₆	90.5 ₂	91.3 ₈	89.7 ₆	89.4 ₄	89.0 ₇	89.2 ₅
Ours+RegW ¹⁰⁹	88.9 ₂	91.0 ₃	89.9 ₂	91.1 ₄	89.8 ₃	90.4 ₆	87.8 ₄	89.2 ₆	88.5 ₉
Ours+RegE ¹¹⁰	88.6 ₅	89.8 ₅	89.2 ₁	91.3 ₂	90.1 ₆	90.7 ₄	87.3 ₃	90.2 ₅	88.7 ₅
Ours+PL ¹¹¹	89.2 ₃	90.4 ₈	89.8 ₉	90.7 ₅	91.4 ₄	91.0 ₃	88.7 ₅	87.9 ₆	88.3 ₆
Ours+PS ¹¹²	89.5 ₄	88.6 ₆	89.0 ₁	90.8 ₃	89.9 ₅	90.3 ₃	88.9 ₈	88.1 ₄	88.5 ₂
Ours+CDRS	91.4 ₂	92.2 ₂	91.8 ₁	92.1 ₂	92.8 ₃	92.4 ₈	90.3 ₃	91.5 ₄	90.9 ₇

Table 1. Precision, recall, and F1 for pneumonia classification from CT scans. Source data is 2D CT slices from COVID-19 related papers. In the X_Y formatted entry, X denotes mean and Y denotes standard deviation in 5 runs with different random initialization.

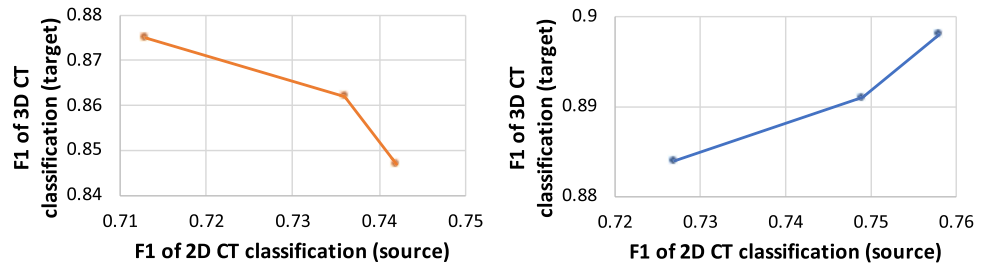


Figure 2. (Left) Evidence of model competition in MTL. (Right) How the performance of the target model varies with the performance of the source model in our framework.

when the test performance of source models (used for regularization) increases. This shows that a better source model yields a better target model in our method.

Our method works better than RegE and PL. These two baselines require embeddings learned by source and target models to be similar in an absolute sense (L2 difference or KL divergence), which is more restrictive than requiring embeddings to preserve ranking as our method does. Besides, these two baselines transfer knowledge on individual data instances without considering the relationship between instances. In contrast, our method compares which target slice is more close to the source dataset during knowledge transfer. Our method works better than PS. PS is limited to capturing second-order relationships among instances while each piece of knowledge in our CDRS method involves two target instances and all source instances, which therefore can capture higher-order relationships. Figure 3 shows 4-nearest source neighbors retrieved by different methods for some randomly sampled target CT slices. As can be seen, compared with PS and PL, nearest neighbors retrieved by CDRS are more semantically similar to query slices. The similarity is evaluated by physicians and is determined based on whether clinical findings in two images are clinically close. A better ability of comparing source and target slices can help to better identify low-quality source data: if a source slice is very different from target slices, it is likely to be noisy or out of target domain. Our method works better than RegW. RegW requires target and source encoders to have the same architecture so that the distance between their weights can be calculated. This requirement prohibits learning representations tailored to specific datasets. 2D CT slices in CC-CCII have different properties than those in COVID-Papers. CC-CCII CTs are obtained from medical imaging databases in hospitals while COVID-Papers CTs are extracted from PDF-format papers. Using the same encoder architecture to represent them fails to account for such differences. In contrast, our method allows different encoders to have different architectures. Our method performs better than WS. In WS, target and source models share the same 2D CT encoder, which prohibits the learning of dataset-specific representations. We performed an ablation study to further investigate whether it is beneficial to let the 2D CT slice encoders in source and target models have different architectures. Specifically, we set the two encoders in Ours+CDRS to be 1) both ResNet-50, and 2) both DenseNet. Table 1 shows the results of these two ablation settings denoted as Same-ResNet and Same-DenseNet. As can be seen, these two ablation settings perform worse than Ours+CDRS which uses different architectures for the two encoders. This further demonstrates that it is beneficial to use different architectures for source and target encoders to learn dataset-specific representations.

Our method works better than BO and MGTL. These two methods reweight source data without considering the performance of the target model on a held-out validation set, which leads to worse generalization performance on test data. Our method outperforms OML and MentorNet. In OML and MentorNet, 3D CT scans

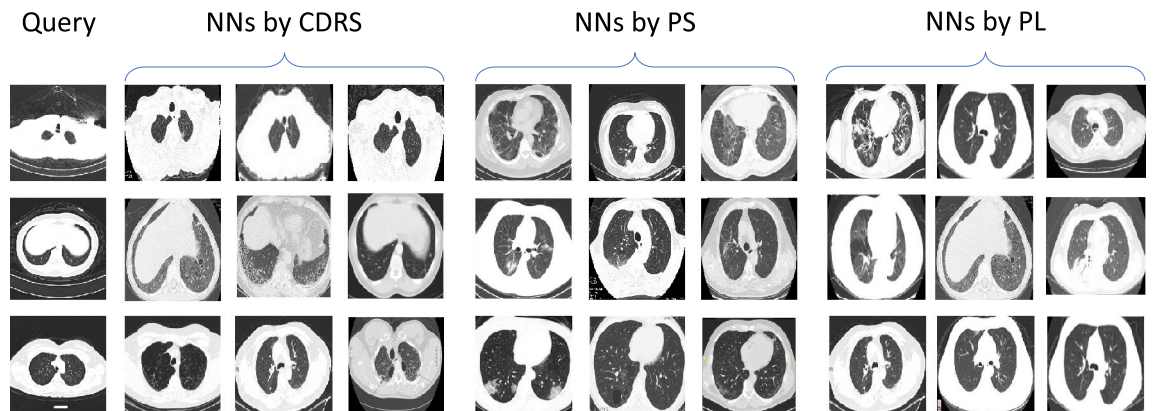


Figure 3. 4-nearest source neighbors retrieved by different methods for some randomly sampled target CT slices. Compared with PS and PL, nearest neighbors retrieved by our CDRS method are more semantically similar to query slices. The similarity is evaluated by physicians and is determined based on whether clinical findings in two images are clinically close.

are reduced to 2D slices for the sake of making target data have compatible format as source data. As a result, temporal information in 3D scans is lost, which leads to worse classification performance. In OML and MentorNet, format compatibility is required on both input data and output labels where the input data needs to be 2D CT slices and the output label is COVID/Non-COVID. To meet such a requirement, when reducing 3D CT scans into 2D slices, each slice needs to be given a COVID/Non-COVID label using heuristics: every 2D slice in a COVID-positive 3D CT scan is labeled as COVID. This heuristic is noisy: it could be possible that some 2D slices in a COVID-positive 3D scan do not contain COVID-related abnormalities. In our method, when calculating $e(x_t, D_s; E_s^*(C))$, format compatibility is only required on input data, not on output labels. Therefore, our method does not suffer from the noisy labeling problem of OML and MentorNet. Our method performs better than Pretrain. This is because Pretrain learns source model and target model separately while our method trains these two models jointly end-to-end.

We also performed experiments where a 3D CT scan dataset¹¹⁷ is used as source data. It contains 753 CT scans of COVID-19 patients. The architecture of the source model in our method is set to be the same as that of the target model described in the Experimental Settings section. Hyperparameters are the same as those described in the Experimental Settings section. For the OML baseline, no format unification between source and target data is needed. Table 2 shows the results. Our method outperforms all baselines. The analysis of reasons is similar to that for results in Table 1.

We also investigated whether the global ranking (of target CT images) generated in our proposed CDRS approach can group images with the same class together. We calculated the percentage of adjacent images (in the ranking) that have the same class label for model checkpoints of our method at different epochs. Figure 4(left) shows how this same-class percentage varies with training epochs, together with each checkpoint's average F1 of the three classes on test data. As can be seen, as our method runs for more epochs, the same-class percentage increases. This demonstrates that our method can encourage images from the same class to be grouped together in the global ranking. In addition, test F1 increases as the same-class percentage increases. This demonstrates that a better grouping of same-class images in the global ranking facilitates pneumonia classification.

	NCP			CP			Normal		
	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)	Precision (%)	Recall (%)	F1 (%)
CC-CCII ⁹⁹	84.5 ₅	87.2 ₇	85.8 ₄	88.9 ₈	87.5 ₆	88.2 ₂	85.9 ₆	86.3 ₇	86.1 ₇
RapidAI ¹¹³	83.1 ₂	86.8 ₅	84.9 ₁	86.3 ₅	86.0 ₄	86.1 ₃	83.8 ₈	85.5 ₆	84.6 ₈
3D-CNN ¹¹	86.6 ₇	87.2 ₅	86.9 ₅	85.4 ₇	87.7 ₉	86.5 ₇	86.8 ₅	85.9 ₉	86.3 ₂
Li et al. ¹¹⁴	86.2 ₅	82.3 ₄	84.2 ₅	90.9 ₇	90.3 ₄	90.6 ₅	89.4 ₄	87.9 ₇	88.6 ₆
Shamsi et al. ¹¹⁵	86.4 ₃	87.4 ₅	86.9 ₆	91.7 ₃	89.9 ₅	90.8 ₄	90.4 ₆	89.2 ₃	89.8 ₅
Shaik et al. ¹¹⁶	87.0 ₆	89.6 ₃	88.3 ₄	89.7 ₅	89.1 ₄	89.4 ₄	87.8 ₂	87.7 ₄	87.8 ₃
NoSrc ¹⁰¹	85.9 ₉	87.5 ₆	86.7 ₆	88.1 ₉	89.2 ₆	88.6 ₉	86.2 ₉	87.1 ₉	86.6 ₅
NoWt ¹⁰¹	89.1 ₄	86.6 ₅	87.8 ₅	88.2 ₆	90.7 ₈	89.4 ₆	90.4 ₅	91.2 ₃	90.8 ₃
Pretrain ¹⁰³	89.9 ₇	92.5 ₂	91.2 ₆	89.6 ₃	91.5 ₈	90.5 ₆	87.9 ₄	88.5 ₇	88.2 ₆
BO ²⁶	89.6 ₃	89.8 ₆	89.7 ₄	91.5 ₅	92.1 ₄	91.8 ₅	89.9 ₄	87.6 ₃	88.7 ₃
MGTL ³¹	90.3 ₈	93.4 ₃	91.8 ₅	91.2 ₄	91.9 ₅	91.5 ₄	90.8 ₉	90.3 ₆	90.5 ₇
OML ³⁵	88.3 ₄	88.9 ₆	88.6 ₅	90.5 ₇	91.3 ₅	90.9 ₆	88.3 ₇	89.5 ₂	88.9 ₄
MentorNet ¹⁰⁷	87.1 ₄	88.7 ₃	87.9 ₃	90.6 ₇	91.1 ₄	90.8 ₅	89.3 ₁	90.5 ₈	89.9 ₆
MTL ¹⁰⁸	91.4 ₆	91.2 ₄	91.3 ₅	90.3 ₉	91.9 ₁	91.1 ₇	90.6 ₈	88.3 ₃	89.4 ₅
WS ⁸⁶	90.4 ₈	90.3 ₄	90.4 ₅	91.7 ₃	91.8 ₇	91.8 ₅	89.9 ₅	89.4 ₃	89.6 ₄
Ours	93.7 ₄	94.6 ₃	94.1 ₃	93.1 ₄	93.6 ₆	93.3 ₅	91.6 ₅	92.9 ₄	92.2 ₃

Table 2. Precision, recall, and F1 for pneumonia classification from CT scans. Source data is 3D CT scans. In the X_Y formatted entry, X denotes mean and Y denotes standard deviation in 5 runs with different random initialization.

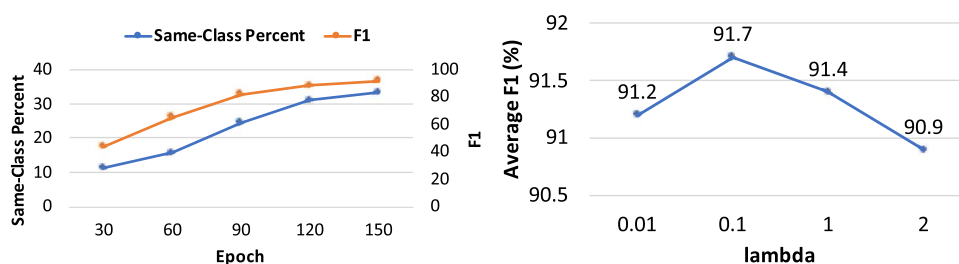


Figure 4. (Left) How same-class percentage and F1 change with epochs in our method. (Right) How the performance of Ours+CDRS changes with λ .

We performed an ablation study on the LSTM component in the target model, by replacing it with an averaging operation: instead of feeding the representations of CT slices into the LSTM network, we average these representations and feed the averaged representation into the classification head. Table 1 shows the results of this ablation setting (denoted as No-LSTM). The performance of No-LSTM is worse than Ours+CDRS which uses LSTM. This is because No-LSTM cannot capture the sequential relationship between CT slices while LSTM can.

Figure 4(right) shows how the performance (average F1 of NCP, CP, and Normal) of Ours+CDRS changes with λ in Eq. (4). As can be seen, a λ value in the middle ground yields the best performance. If λ is too small (e.g., 0.01), there is not sufficient knowledge transfer from source to target. If λ is too large (e.g., 2), the target encoder is excessively influenced by the source encoder and therefore is less capable of learning representations that are tailored to the target dataset.

Figure 5 shows some source examples whose importance weights learned by our method are close to 0 (indicating these examples are noisy or have large domain discrepancy with target data). The range of importance weights after optimization is between [0,1]. As can be seen, some of these examples are indeed noisy. For instance, some contain artifacts such as bounding boxes. In some examples, lung regions are distorted. Though images on the last row do not contain obvious artifact noise, their appearance, texture, color, scale, and positions of lungs are different from those in the target dataset. This shows that our method is not only able to identify obviously noisy source examples, but also those having domain discrepancy with target data. MTL and OML incorrectly assign close-to-zero weights to some images that are clean and have large domain similarity to target data. Another finding is that our method tends to give COVID-19 CTs more weight to make the two classes (COVID and Non-COVID) more balanced. Without reweighting, the ratio between these two classes is about 0.75. After reweighting, the ratio is 0.84, getting closer to 1. As a result, performance of different classes in the 3D CT scan test set is more balanced. Without reweighting (NoWt), the ratio of F1 scores achieved on NCP, CP, and Normal is 1:1.03:1.02. After reweighting (ours), the ratio becomes 1:1.01:0.99.

Figure 6 shows two failure cases of our method. Each row contains some slices of a target CT scan. The first CT scan is from the CP class, but is incorrectly predicted as being from the NCP class. The second CT scan is

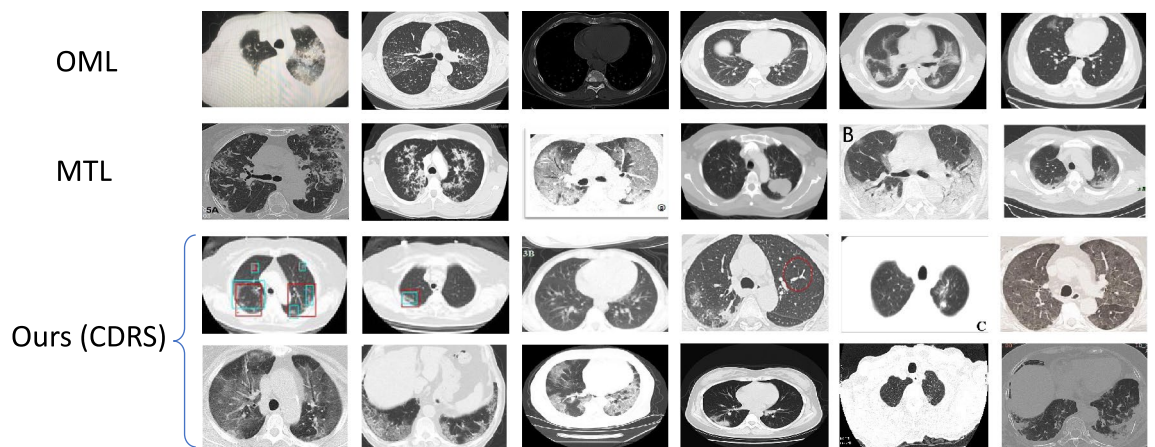


Figure 5. Randomly-sampled source images whose importance weights learned by different methods are close to 0. Our method can successfully identify images containing artifact noises such as bounding boxes or having large domain discrepancies with target data in terms of appearance, texture, color, scale, etc. In contrast, OML and MTL incorrectly assign close-to-zero weights to some images that are clean and have large domain similarity to target data.

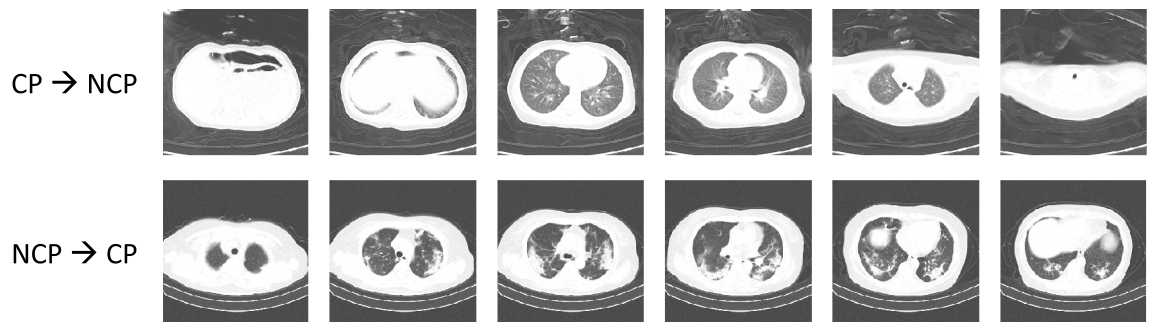


Figure 6. Two failure cases of our method. Each row contains some slices of a target CT scan. The first CT scan is from the CP class, but is incorrectly predicted as being from the NCP class. The second CT scan is from the NCP class, but is incorrectly predicted as being from the CP class. The cause of failure is that the two types of pneumonia have similar findings in CTs, which makes it challenging to distinguish them.

from the NCP class, but is incorrectly predicted as being from the CP class. The cause of failure is that the two types of pneumonia have similar findings in CTs, which makes it challenging to distinguish them.

Conclusions

To accurately detect pneumonia from CT scans and differentiate COVID-19 and other types of pneumonia, we propose a three-level optimization based method which leverages CT images from a source domain to improve the training of a target model. Our method automatically identifies and down-weights low-quality source CT data examples that are noisy or have large discrepancies with target domain, by checking whether a source example is helpful in reducing validation loss of the target model. Our framework involves three learning stages. At the first stage, we train a source model on weighted source data. At the second stage, by transferring knowledge from source model to target model, we train a target model. We propose a novel knowledge transfer approach based on cross-domain relative similarity. At the third stage, we learn the importance weights of source examples by minimizing the validation loss of the target model. The three stages are performed end-to-end. Our method achieves an F1 score of 91.8% in detecting COVID-19 and an F1 score of 92.4% in detecting other types of pneumonia and outperforms state-of-the-art baselines significantly.

Data availability

The China Consortium of Chest CT Image Investigation (CC-CCII) is used as target dataset, which is available at <http://ncov-ai.big.ac.cn/download?lang=en> The source dataset is available at <https://github.com/UCSD-AI4H/COVID-CT> and <https://wiki.cancerimagingarchive.net/display/Public/CT+Images+in+COVID-19#702271073df18fec0c954449874468ec045f67d>

Received: 9 October 2022; Accepted: 26 May 2023

Published online: 09 June 2023

References

- <https://www.who.int/news-room/fact-sheets/detail/pneumonia>.
- Syrjälä, H., Broas, M., Suramo, I., Ojala, A. & Lähde, S. High-resolution computed tomography for the diagnosis of community-acquired pneumonia. *Clin. Infect. Dis.* **27**(2), 358–363 (1998).
- Koo, H. J. *et al.* Radiographic and ct features of viral pneumonia. *Radiographics* **38**(3), 719–739 (2018).
- Duan, Y., Zhu, Y., Tang, L. & Qin, J. Ct features of novel coronavirus pneumonia (covid-19) in children. *Eur. Radiol.* **30**(8), 4427–4433 (2020).
- Reitner, P., Ward, S., Heyneman, L., Johkoh, T. & Müller, N. L. Pneumonia: high-resolution ct findings in 114 patients. *Eur. Radiol.* **13**(3), 515–521 (2003).
- Upchurch, C. P. *et al.* Community-acquired pneumonia visualized on ct scans but not chest radiographs: Pathogens, severity, and clinical outcomes. *Chest* **153**(3), 601–610 (2018).
- Qian, X. *et al.* M3 lung-sys: A deep learning system for multi-class lung pneumonia screening from ct imaging. *IEEE J. Biomed. Health Inf.* **24**(12), 3539–3550 (2020).
- Li, Z. *et al.* From community-acquired pneumonia to covid-19: A deep learning-based method for quantitative analysis of covid-19 on thick-section ct scans. *Eur. Radiol.* **30**(12), 6828–6837 (2020).
- Amyar, A., Modzelewski, R., Li, H. & Ruan, S. Multi-task deep learning based ct imaging analysis for covid-19 pneumonia: Classification and segmentation. *Comput. Biol. Med.* **126**, 104037 (2020).
- Ni, Q. *et al.* A deep learning approach to characterize 2019 coronavirus disease (covid-19) pneumonia in chest ct images. *Eur. Radiol.* **30**(12), 6517–6527 (2020).
- Xu, X., Jiang, X., Ma, C., Du, P., Li, X., Lv, S., Yu, L., Chen, Y., Su, J., Lang, G., *et al.* Deep learning system to screen coronavirus disease 2019 pneumonia. arXiv preprint [arXiv:2002.09334](https://arxiv.org/abs/2002.09334) (2020).
- Abdel-Basset, M., Hawash, H., Moustafa, N. & Elkomy, O. M. Two-stage deep learning framework for discrimination between covid-19 and community-acquired pneumonia from chest ct scans. *Pattern Recognit. Lett.* **152**, 311–319 (2021).
- Wu, W., Guo, X., Chen, Y., Wang, S., & Chen, J. Deep embedding-attention-refinement for sparse-view ct reconstruction. *IEEE Trans. Instrum. Measur.* (2022).
- Weiwien, W. *et al.* Drone: Dual-domain residual-based optimization network for sparse-view ct reconstruction. *IEEE Trans. Med. Imaging* **40**(11), 3002–3014 (2021).
- You, C. *et al.* Class-aware adversarial transformers for medical image segmentation. *Adv. Neural Inf. Process. Syst.* **35**, 29582–29596 (2022).
- You, C., Dai, W., Liu, F., Su, H., Zhang, X., Staib, L., & Duncan, J. S. Mine your own anatomy: Revisiting medical image segmentation with extremely limited labels. arXiv preprint [arXiv:2209.13476](https://arxiv.org/abs/2209.13476) (2022).
- You, C., Zhou, Y., Zhao, R., Staib, L. & Duncan, J. S. Simcvd: Simple contrastive voxel-wise representation distillation for semi-supervised medical image segmentation. *IEEE Trans. Med. Imaging* **41**(9), 2228–2237 (2022).
- You, C., Zhao, R., Staib, L. H., & Duncan, J. S. Momentum contrastive voxel-wise representation learning for semi-supervised volumetric medical image segmentation. In *Medical Image Computing and Computer Assisted Intervention—MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part IV*, 639–652 (Springer, 2022).
- You, C., Dai, W., Staib, L., & Duncan, J. S. Bootstrapping semi-supervised medical image segmentation with anatomical-aware contrastive distillation. arXiv preprint [arXiv:2206.02307](https://arxiv.org/abs/2206.02307) (2022).
- You, C., Dai, W., Min, Y., Staib, L., Sekhon, J., & Duncan, J. S. Action++: Improving semi-supervised medical image segmentation with adaptive anatomical contrast. arXiv preprint [arXiv:2304.02689](https://arxiv.org/abs/2304.02689) (2023).
- You, C., Dai, W., Min, Y., Staib, L., & Duncan, J. S. *Implicit Anatomical Rendering for Medical Image Segmentation with Stochastic Experts*. arXiv preprint [arXiv:2304.03209](https://arxiv.org/abs/2304.03209) (2023).
- You, C., Dai, W., Min, Y., Liu, F., Zhang, X., Feng, C., Clifton, D. A., Kevin Zhou, S., Staib, L. H., & Duncan, J. S. Rethinking semi-supervised medical image segmentation: A variance-reduction perspective. arXiv preprint [arXiv:2302.01735](https://arxiv.org/abs/2302.01735) (2023).
- Ortiz, A. *et al.* Effective deep learning approaches for predicting covid-19 outcomes from chest computed tomography volumes. *Sci. Rep.* **12**(1), 1–10 (2022).
- Zhuang, F. *et al.* A comprehensive survey on transfer learning. *Proc. IEEE* **109**(1), 43–76 (2020).
- Ge, W., & Yu, Y. Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning. CoRR [abs/1702.08690](https://arxiv.org/abs/1702.08690) (2017).
- Ruder, S., & Plank, B. Learning to select data for transfer learning with bayesian optimization. CoRR, [arXiv:abs/1707.05246](https://arxiv.org/abs/1707.05246) (2017).

27. Zhang, Y., David, P., & Gong, B. Curriculum domain adaptation for semantic segmentation of urban scenes. CoRR, abs/1707.09465 (2017).
28. Guo, H., Pasunuru, R., & Bansal, M. Autosem: Automatic task selection and mixing in multi-task learning. CoRR abs/1904.04153 (2019).
29. Liu, H., Long, M., Wang, J., & Jordan, M. Transferable adversarial training: A general approach to adapting deep classifiers. In Chaudhuri, K., & Salakhutdinov, R., (editors) *Proceedings of the 36th International Conference on Machine Learning, volume 97 of Proceedings of Machine Learning Research*, 4013–4022. PMLR, 09–15 (2019).
30. Tang, H., & Jia, K. Discriminative adversarial domain adaptation. CoRR, arXiv:abs/1911.12036 (2019).
31. Wang, M., Qiu, X., Wang, Y., Li, Y., Gong, X., Zeng, J., Huang, B., Zheng, D., Cai, J., Zhou, A. A minimax game for instance based selective transfer learning. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2019).
32. Wang, Y., Zhao, D., Li, Y., Chen, K.-J., & Xue, H. The most related knowledge first: A progressive domain adaptation method. In *PAKDD* (2019).
33. Bateson, M., Kervadec, H., Dolz, J., Lombaert, H., & Ayed, I. B. *Source-relaxed domain adaptation for image segmentation*. CoRR abs/2005.03697 (2020).
34. Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., & Pontil, M. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, 1568–1577 (PMLR, 2018).
35. Ren, M., Zeng, W., Yang, B., & Urtasun, R. Learning to reweight examples for robust deep learning. arXiv preprint arXiv:1803.09050 (2018).
36. Hu, Z., Tan, B., Salakhutdinov, R., Mitchell, T. M., & Xing, E.P. Learning data manipulation for augmentation and weighting. CoRR abs/1910.12795 (2019).
37. Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., & Meng, D. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems*, 1919–1930 (2019).
38. Ren, Z., Yeh, R., & Schwing, A. Not all unlabeled data are equal: Learning to weight data in semi-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems, volume 33*, 21786–21797. Curran Associates, Inc. (2020).
39. Wang, Y., Guo, J., Song, S., & Huang, G. *Meta-semi: A meta-learning approach for semi-supervised learning*. CoRR, abs/2007.02394 (2020).
40. Wang, X., Pham, H., Michel, P., Anastasopoulos, A., Carbonell, J., & Neubig, G. Optimizing data usage via differentiable rewards. In *International Conference on Machine Learning*, 9983–9995. (PMLR, 2020).
41. Zhang, J. *et al.* Automated detection and quantification of covid-19 pneumonia: Ct imaging analysis by a deep learning-based software. *Eur. J. Nucl. Med. Mol. Imaging* **47**(11), 2525–2532 (2020).
42. Ko, H., Chung, H., Kang, W. S., Kim, K. W., Shin, Y., Kang, S. J., Lee, J. H., Kim, Y. J., Kim, N. Y., Jung, H., *et al.* Covid-19 pneumonia diagnosis using a simple 2d deep learning framework with a single chest ct image: model development and validation. *J. Med. Internet Res.* **22**(6), e19569 (2020).
43. Maghdid, H. S., Asaad, A. T., Ghafoor, K. Z., Sadiq, A. S., Mirjalili, S., & Khan, M. K. Diagnosing covid-19 pneumonia from x-ray and ct images using deep learning and transfer learning algorithms. In *Multimodal image exploitation and learning 2021, volume 11734*, 99–110. SPIE (2021).
44. Chen, J., *et al.* Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography: A prospective study. *medRxiv* (2020).
45. Zhou, M. *et al.* Deep learning for differentiating novel coronavirus pneumonia and influenza pneumonia. *Ann. Transl. Med.* **9**(2), 1 (2021).
46. Chaudhary, S., Sadbhawna, S., Jakheti, V., Subudhi, B. N., Baid, U., & Guntuku, S. C. Detecting covid-19 and community acquired pneumonia using chest ct scan images with deep learning. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8583–8587. (IEEE, 2021).
47. Bermejo-Peláez, D. *et al.* Deep learning-based lesion subtyping and prediction of clinical outcomes in covid-19 pneumonia using chest ct. *Sci. Rep.* **12**(1), 1–11 (2022).
48. Yao, J.-C. *et al.* Ai detection of mild covid-19 pneumonia from chest ct scans. *Eur. Radiol.* **31**(9), 7192–7201 (2021).
49. Song, Y. *et al.* Deep learning enables accurate diagnosis of novel coronavirus (covid-19) with ct images. *IEEE/ACM Trans. Comput. Biol. Bioinf.* **18**(6), 2775–2780 (2021).
50. Bratt, A. *et al.* Predicting usual interstitial pneumonia histopathology from chest ct imaging with deep learning. *Chest* **1**, 1 (2022).
51. Shiri, I. *et al.* High-dimensional multinomial multiclass severity scoring of covid-19 pneumonia using ct radiomics features and machine learning algorithms. *Sci. Rep.* **12**(1), 1–12 (2022).
52. Jiang, J., & Zhai, C. X. *Instance weighting for domain adaptation in nlp* (ACL, 2007).
53. Foster, G., Goutte, C., & Kuhn, R. Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 451–459 (2010).
54. Moore, R. C., & Lewis, W. *Intelligent selection of language model training data* (2010).
55. Axelrod, A., He, X., & Gao, J. Domain adaptation via pseudo in-domain data selection. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 355–362 (2011).
56. Sivasankaran, S., Vincent, E., & Illina, I. Discriminative importance weighting of augmented training data for acoustic model training. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4885–4889 (IEEE, 2017).
57. Patel, Y., Chitta, K., & Jasani, B. Learning sampling policies for domain adaptation. CoRR abs/1805.07641 (2018).
58. Qu, C., Ji, F., Qiu, M., Yang, L., Min, Z., Chen, H., Huang, J., & Croft, W. B. Learning to selectively transfer: Reinforced transfer learning for deep text matching. CoRR, abs/1812.11561 (2018).
59. Liu, M., Song, Y., Zou, H., & Zhang, T. Reinforced training data selection for domain adaptation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, July 2019*. Association for Computational Linguistics (2019).
60. Song, Y. *Entropy-based training data selection for domain adaptation*.
61. Wang, R., Utiyama, M., Liu, L., Chen, K., & Sumita, E. Instance weighting for neural machine translation domain adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1482–1488, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.
62. Pratt, L. Y. Discriminability-based transfer between neural networks. *Advances in neural information processing systems*, 204–204 (1993).
63. Mihalkova, L., Huynh, T., & Mooney, R. J. Mapping and revising Markov logic networks for transfer learning. In *Aaai, volume 7*, 608–614 (2007).
64. Niculescu-Mizil, A., & Caruana, R. Inductive transfer for bayesian network structure learning. In *Artificial intelligence and statistics*, 339–346. PMLR (2007).
65. Pan, S. J. & Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2009).
66. Luo, Z., Zou, Y., Hoffman, J., & Fei-Fei, L. Label efficient learning of transferable representations across domains and tasks. arXiv preprint arXiv:1712.00123 (2017).

67. Borgwardt, K. M. *et al.* Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics* **22**(14), e49–e57 (2006).
68. Pan, S. J., Tsang, I. W., Kwok, J. T. & Yang, Q. Domain adaptation via transfer component analysis. *IEEE Trans. Neural Networks* **22**(2), 199–210 (2010).
69. Long, M., Wang, J., Ding, G., Sun, J., & Yu, P. S. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, 2200–2207 (2013).
70. J. Wang, Y. Chen, S. Hao, W. Feng, & Shen, Z. Balanced distribution adaptation for transfer learning. In *2017 IEEE International Conference on Data Mining (ICDM)*, 1129–1134 (IEEE, 2017).
71. Huang, J., Gretton, A., Borgwardt, K., Schölkopf, B. & Smola, A. Correcting sample selection bias by unlabeled data. *Adv. Neural Inf. Process. Syst.* **19**, 601–608 (2006).
72. Wang, R., Utiyama, M., Liu, L., Chen, K., & Sumita, E. Instance weighting for neural machine translation domain adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1482–1488 (2017).
73. Ngiam, J., Peng, D., Vasudevan, V., Kornblith, S., Le, Q. V. & Pang, R. Domain adaptive transfer learning with specialist models. arXiv preprint [arXiv:1811.07056](https://arxiv.org/abs/1811.07056) (2018).
74. Ganin, Y. *et al.* Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(1), 2030–2096 (2016).
75. Long, M., Cao, Z., Wang, J., & Jordan, M. I. Conditional adversarial domain adaptation. arXiv preprint [arXiv:1705.10667](https://arxiv.org/abs/1705.10667) (2017).
76. Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., & Darrell, T. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, 1989–1998. PMLR (2018).
77. Zhang, Y., Tang, H., Jia, K., & Tan, M. Domain-symmetric networks for adversarial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5031–5040 (2019).
78. Luo, P., Zhuang, F., Xiong, H., Xiong, Y., & He, Q. Transfer learning from multiple source domains via consensus regularization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 103–112 (2008).
79. Tommasi, T., Orabona, F., & Caputo, B. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3081–3088 (IEEE, 2010).
80. Duan, L., Xu, D. & Tsang, I. W.-H. Domain adaptation from multiple sources: A domain-dependent regularization approach. *IEEE Trans. Neural Networks Learn. Syst.* **23**(3), 504–518 (2012).
81. Ruder, S. An overview of multi-task learning in deep neural networks. arXiv preprint [arXiv:1706.05098](https://arxiv.org/abs/1706.05098) (2017).
82. Zhang, Y., & Yang, Q. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering* (2021).
83. Caruana, R. A. *Multitask learning: A knowledge-based source of inductive bias, ser* (Springer, Learning to Learn. US, 1998).
84. Long, M., Cao, Z., Wang, J., & Yu, P. S. Learning multiple tasks with multilinear relationship networks. arXiv preprint [arXiv:1506.02117](https://arxiv.org/abs/1506.02117) (2015).
85. Doersch, C., & Zisserman, A. Multi-task self-supervised visual learning. In *Proceedings of the IEEE International Conference on Computer Vision*, 2051–2060 (2017).
86. Kokkinos, I. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6129–6138 (2017).
87. Leang, I., Sistu, G., Bürger, F., Bursuc, A., & Yogamani, S. Dynamic task weighting methods for multi-task networks in autonomous driving systems. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 1–8. (IEEE, 2020).
88. Yang, Y., & Hospedales, T. M. Trace norm regularised deep multi-task learning. arXiv preprint [arXiv:1606.04038](https://arxiv.org/abs/1606.04038) (2016).
89. Williams, C., Bonilla, E. V., & Chai, K. M. Multi-task gaussian process prediction. *Advances in neural information processing systems*, 153–160 (2007).
90. Zhang, Yu. & Yeung, D.-Y. A regularization approach to learning task relationships in multitask learning. *ACM Trans. Knowl. Discov. Data (TKDD)* **8**(3), 1–31 (2014).
91. Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., & Savarese, S. Which tasks should be learned together in multi-task learning? In *International Conference on Machine Learning*, 9120–9132 (PMLR, 2020).
92. Dempe, S. *Foundations of Bilevel Programming* (Springer, 2002).
93. Feurer, M., Springenberg, J., & Hutter, F. Initializing Bayesian hyperparameter optimization via meta-learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29 (2015).
94. Liu, H., Simonyan, K., & Yang, Y. DARTS: differentiable architecture search. In *ICLR* (2019).
95. Finn, C., Abbeel, P., & Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, Vol. 70, 1126–1135 (JMLR. org, 2017).
96. Baydin, A. G., Cornish, R., Martínez-Rubio, D., Schmidt, M., & Wood, F. D. Online learning rate adaptation with hypergradient descent. CoRR abs/1703.04782 (2017).
97. Zheng, G., Awadallah, A. H. & Dumais, S. T. Meta label correction for learning with weak supervision. CoRR [arXiv:abs/1911.03809](https://arxiv.org/abs/1911.03809) (2019).
98. Such, F. P., Rawal, A., Lehman, J., Stanley, K. O. & Clune, J. Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data. CoRR abs/1912.07768 (2019).
99. Zhang, K. *et al.* Clinically applicable ai system for accurate diagnosis, quantitative measurements, and prognosis of covid-19 pneumonia using computed tomography. *Cell* **181**(6), 1423–1433 (2020).
100. Yang, X., He, X., Zhao, J., Zhang, Y., Zhang, S., & Xie, P. *Covid-ct-dataset: A ct scan dataset about covid-19* (2020).
101. Ng, J. Y.-H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., & Toderici, G. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4694–4702 (2015).
102. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997).
103. He, K., Zhang, X., Ren, S., & Sun, J. Deep residual learning for image recognition. In *CVPR* (2016).
104. Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700–4708 (2017).
105. Kingma, D., & Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 12 (2014).
106. Krizhevsky, A., Sutskever, I., & Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *NIPS* (2012).
107. Jiang, L., Zhou, Z., Leung, T., Li, L. J., & Fei-Fei, L. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, 2304–2313 (PMLR, 2018).
108. Maninis, K. K., Radosavovic, I., & Kokkinos, I. Attentive single-tasking of multiple tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1851–1860 (2019).
109. Rajeswaran, A., Finn, C., Kakade, S., & Levine, S. Meta-learning with implicit gradients. arXiv preprint [arXiv:1909.04630](https://arxiv.org/abs/1909.04630) (2019).
110. Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., & Bengio, Y. Fitnets: Hints for thin deep nets. arXiv preprint [arXiv:1412.6550](https://arxiv.org/abs/1412.6550) (2014).
111. Pham, H., Xie, Q., Dai, Z., & Le, Q. V. Meta pseudo labels. arXiv preprint [arXiv:2003.10580](https://arxiv.org/abs/2003.10580) (2020).
112. Chen, T., Kornblith, S., Norouzi, M., & Hinton, G. A simple framework for contrastive learning of visual representations. arXiv preprint [arXiv:2002.05709](https://arxiv.org/abs/2002.05709) (2020).

113. Gozes, O., Frid-Adar, M., Greenspan, H., Browning, P. D., Zhang, H., Ji, W., Bernheim, A., & Siegel, E. *Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection & patient monitoring using deep learning ct image analysis*. arXiv preprint [arXiv:2003.05037](https://arxiv.org/abs/2003.05037) (2020).
114. Li, C., Yang, Y., Liang, H. & Boying, W. Transfer learning for establishment of recognition of covid-19 on ct imaging using small-sized training datasets. *Knowl.-Based Syst.* **218**, 106849 (2021).
115. Shamsi, A. *et al.* An uncertainty-aware transfer learning-based framework for covid-19 diagnosis. *IEEE Trans. Neural Networks Learn. Syst.* **32**(4), 1408–1417 (2021).
116. Shaik, N. S. & Cherukuri, T. K. Transfer learning based novel ensemble classifier for covid-19 detection from chest ct-scans. *Comput. Biol. Med.* **141**, 105127 (2022).
117. An, P. *et al.* Ct images in covid-19 [data set]. *Cancer Imag. Arch.* **10**, 1 (2020).

Author contributions

P.X. proposed the research problem, designed the method, implemented part of the method, conducted part of the experiments, and wrote the manuscript. X.Z. and X.H. implemented part of the method and conducted part of the experiments. All authors have reviewed the manuscript.

Competing Interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to P.X.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023