Managing Computer-Assisted Detection System Based on Transfer Learning with Negative Transfer Inhibition

Issei Sato
The University of Tokyo
sato@k.u-tokyo.ac.jp

Yukihiro Nomura
The University of Tokyo Hospital
nomuray-tky@umin.ac.jp

Shouhei Hanaoka
The University of Tokyo Hospital
hanaoka-tky@umin.ac.jp

Soichiro Miki
The University of Tokyo Hospital
smiki-tky@umin.ac.jp

Naoto Hayashi
The University of Tokyo Hospital
naoto-tky@umin.ac.jp

Osamu Abe
The University of Tokyo Hospital
abediag-tky@umin.ac.jp

Yoshitaka Masutani
Hiroshima City University
masutani@hiroshima-cu.ac.jp

ABSTRACT

The reading workload for radiologists is increasing because the numbers of examinations and images per examination are increasing due to the technical progress on imaging modalities such as computed tomography and magnetic resonance imaging. A computer-assisted detection (CAD) system based on machine learning is expected to assist radiologists. The preliminary results of a multi-institutional study indicate that the performance of the CAD system for each institution improved using training data of other institutions. This indicates that transfer learning may be useful for developing the CAD systems among multiple institutions. In this paper, we focus on transfer learning without sharing training data due to the need to protect personal information in each institution. Moreover, we raise a problem of negative transfer in CAD system and propose an algorithm for inhibiting negative transfer. Our algorithm provides a theoretical guarantee for managing CAD software in terms of transfer learning and exhibits experimentally better performance compared to that of the current algorithm in cerebral aneurysm detection.

CCS CONCEPTS

• **Computing methodologies** → **Transfer learning**;

KEYWORDS

Medical image analysis, Computer-assisted detection, Machine learning, Transfer learning, Negative transfer

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219868

ACM Reference Format:
Issei Sato, Yukihiro Nomura, Shouhei Hanaoka, Soichiro Miki, Naoto Hayashi, Osamu Abe, and Yoshitaka Masutani. 2018. Managing Computer-Assisted Detection System Based on Transfer Learning with Negative Transfer Inhibition . In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219868

1 INTRODUCTION

Diagnoses with clinical image analysis system, such as computed tomography (CT) and magnetic resonance imaging (MRI) scans, have been introduced at medical facilities and the number of examinations have been increasing. A radiologist analyzes abnormalities from the images of examinations and diagnoses based on the interpretation. In our university hospital, image analysis is conducted for 10 – 30 minutes per patient and it is possible to display from 100 to 1000 images per examination and this number is increasing yearly due to the technical progress of imaging modalities.

The time limitation is expected to be more severe. The number of lesion sites are much less than that of normal sites, which is characteristic in clinical image analysis. A radiologist is required to carefully read images and not miss lesions within a short time. Therefore, a radiologist has to meet very difficult requirements. In this situation, computer-assisted detection (CAD) systems are expected to lighten the burden of radiologists and prevent failures in finding lesions [4, 24].

We developed CAD system, which after reading clinical images, automatically identifies legion sites and displays them to a radiologist. Supervised learning has been used to predict lesion sites within our system. The prediction of lesion sites is handled as binary classification, which predicts whether a class is "positive (abnormal)" or "negative (normal)." The performance of CAD system depends on the quality and quantity of the dataset used for supervised learning. If the data characteristics in development and practical use differ, the performance of CAD system will degrade.

For example, the quality of magnetic resonance angiography (MRA) images depends on the scanner, particularly the strength of the static magnetic field [26]. Ideally, CAD system should be evaluated in advance using a dataset with the same characteristics as those during a practical use, and the classifier should be retained as needed [6, 11, 21]. Hence, data need to be continuously collected for supervised learning for practical use, and CAD system needs to be updated by retraining it with the collected data.

We developed a web-based CAD system processing and evaluation platform, called "clinical infrastructure for radiologic computation of united solutions clinical server (CIRCUS CS)," which executes the on-line processing of CAD system and provides interfaces to evaluate the results obtained from the CAD system (clinical feedback)[14]. This system is used daily in our university hospital. The collected feedback data enable us to improve the performance of CAD system by retraining the classifier. For a multi-institutional study, we implemented our system to a teleradiology environment and has been in practical use since September 2011 [15]. The results of a simulation study show that the performance of the CAD system for each institution improved by retraining it. However, the appropriate dataset depends on the target population, number of training cases, and type of classifier. Finding the appropriate dataset by cut-and-try is difficult for practical use. Moreover, if retraining is carried out at each institution, difficulty occurs in obtaining data on other institutions including datasets of initial development due to the need to protect personal information.

When only a very small amount of data is available in a task despite sufficient training data existing in other tasks, *transfer leaning* is known as an effective solution in machine learning. Transfer learning is aimed to create a better prediction model in a *target domain* using knowledge in *source domains* [9]. In transfer learning, source training data are typically assumed to be available when training the target domain and using the training data on the target and source domains. This paper considers a situation in which the training data in the source domain are not available for training in the target domain due to the need to protect personal information in each institution.

Online transfer learning (OTL) [27] is aimed to train the prediction model in an online fashion and involves an algorithm to train the prediction model step-by-step each times the algorithm receives new data from a data stream in the target domain. That is, all training data in source domains have been passed along in contrast to many other transfer algorithms in batch learning. In this setting, we have prediction functions in source domains that have been previously trained with enough data. We share only outputs of learning algorithms.

The motivation regarding OTL is different from ours, but the OTL algorithm does not use source data, which meets the purpose for this study. However, as we found in our experiments, the OTL algorithm can degrade the performance of the target domain, which is common problem of transfer learning and called "negative transfer." In the training of CAD system with low prevalence, it is preferable to avoid performance degradation due to negative transfer even in the beginning of training with an insufficient amount of training data.

The summary of this paper is as follows.

- We focus on transfer learning without source data due to the need to protect personal information in each institution.
- We formulate negative transfer in CAD software and propose a modified OTL algorithm inhibiting negative transfer.
- Our algorithm has a theoretical guarantee for transfer learning in CAD software and exhibits experimentally better performance compared to that of the current OTL algorithm in cerebral aneurysm detection.

2 RELATED WORK

In the field of medical image processing, several studies utilized transfer learning to adapt various data. Opbroek et al. [25] investigated the supervised algorithm of MRI brain segmentation and showed that transfer learning could improve performance of segmentation across scanners and imaging protocols. Engelen et al. [23] investigated an automated segmentation of plaque components in carotid artery MRI and showed that the combination of feature normalization and transfer learning achieves the best segmentation performance across scanners. Sonoyama et al. [20] investigated the endoscopic image classification taken by different (old and new) endoscope systems. Conjeti et al. [3] proposed a novel supervised domain adaptation of random forests and validated the method by characterizing heterogeneous atherosclerotic tissues on intravascular ultrasound (*in vitro* and *in vivo*).

Recently, deep convolutional neural networks (CNNs) [10] have been used in medical image processing [5]. Though CNNs require a large amount of labeled training data, collecting a large number of labeled medical image datasets is difficult. Therefore, transfer learning, i.e., fine-tuning CNN models pre-trained from natural image datasets to medical image datasets, is a key component for using deep CNNs in medical imaging applications [1, 17, 19, 22].

3 ONLINE TRANSFER LEARNING

The OTL algorithm [27] is formulated as *prediction with experts* [2] where each classifier in source and target domains is regarded as an expert. We first give a general formulation of expert predictions then explain the OTL algorithm.

3.1 Expert predictions

Suppose that there are K predictors, called *experts*, that output the prediction for each round. The problem, called *prediction with expert*, is to create a meta-algorithm that sequentially selects the expert with best prediction.

Each expert $k = 1, 2, \ldots, K$ in round t outputs prediction $h_{k,t} \in \mathcal{H}$ where $\mathcal{H}$ is a convex subset of some vector space and means the prediction value space of each expert. We

predict $\hat{h}_t$ by combining the experts' predictions in round t. Then, the true outcome $y_t \in \mathcal{Y}$ is revealed. Each expert $k = 1, 2, \ldots, K$ and the combined prediction suffers from loss $\ell(h_{k,t}, y_t)$ and $\ell(\hat{h}_t, y_t)$.

Assumption 1. *We make two assumptions on the loss function $\ell : \mathcal{H} \times \mathcal{Y} \to \mathbb{R}$:*

(1) *$\ell(h, y) \in [0, 1]$ for any $h \in \mathcal{H}$, $y \in \mathcal{Y}$,*
(2) *for any fixed $y \in \mathcal{Y}$, loss function $\ell(h, y)$ is convex with respect to $h \in \mathcal{H}$.*

Many other algorithms have been extensively studied for prediction with experts. The exponential weighted average (EWA) algorithm [12] is one of the most common algorithms. It updates weight $w_{k,t}$ for expert k:

$$w_{k,t} = w_{k,t-1} \exp(-\eta \ell(h_{k,t}, y_t)), \tag{1}$$

where $\eta > 0$ is a parameter. Then, it predicts the next outcome by taking the convex combination of experts' predictions with the current weights:

$$\widehat{h}_{t+1} = \sum_{k=1}^{K} p_{k,t} h_{k,t+1}, \; p_{k,t} = \frac{w_{k,t}}{\sum_{k'=1}^{K} w_{k',t}}. \tag{2}$$

Since the EWA algorithm only combines the outputs of experts, it does not need to determine how each expert predicts. The OTL algorithm uses this property of the EWA algorithm.

3.2 OTL algorithm

We define a source demain and a target domain over $\mathcal{X} \times \mathcal{Y}$ where we typically assume $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} = \{-1, 1\}$. The OTL algorithm is aimed to learn prediction function $h : \mathcal{X} \to [0, 1]$ from a sequence of examples $\{(x_t, y_t) | t, = 1, 2, \ldots, T\}$ on the target domain. Let h^{src} and h^{tgt} be classifiers learned in the source and target domains, respectively.

The OTL algorithm predicts by taking the convex combination of the source and target classifiers' predictions with the current weights; the transfer learning classifier, denoted as h^{trs}, is given by

$$h^{\mathrm{trs}}_{t+1}(x) = \frac{w^{\mathrm{src}}_t}{w^{\mathrm{src}}_t + w^{\mathrm{tgt}}_t} h^{\mathrm{src}}_{t+1}(x) + \frac{w^{\mathrm{tgt}}_t}{w^{\mathrm{src}}_t + w^{\mathrm{tgt}}_t} h^{\mathrm{tgt}}_{t+1}(x), \tag{3}$$

where the OTL algorithm suggests the following updating scheme with parameter $\eta > 0$ for adjusting the weights:

$$w^{\mathrm{src}}_t = w^{\mathrm{src}}_{t-1} \exp(-\eta \ell(h^{\mathrm{src}}_t(x_t), y_t))), \tag{4}$$

$$w^{\mathrm{tgt}}_t = w^{\mathrm{tgt}}_{t-1} \exp(-\eta \ell(h^{\mathrm{tgt}}_t(x_t), y_t))). \tag{5}$$

This update indicates that a large value for the loss makes the weight small. That is, we adjust the weights according to the performance of the classifiers for each round. In the next section, we explain how to apply the OTL algorithm to improve CAD software and its drawbacks.

4 PROPOSED FRAMEWORK

In spite of the useful property, "not needing the source data," on the OTL algorithm, we found from the experimental results discussed in Sec. 5 that the OTL algorithm is negatively affected by negative transfer. In this section, we first explain the evaluation of CAD software then formulate negative transfer in CAD software based on the evaluation. Finally, we explain our modified OTL algorithm to inhibit negative transfer.

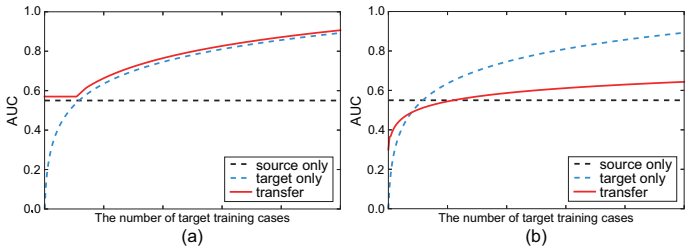

Figure 1: Examples of transfer learning. (a) reasonable transfer, (b) negative transfer.

4.1 Evaluation of CAD software

It is worth considering the evaluation criteria in medical image analysis because the data is typically imbalanced. For example, the classification error rate is not the most pertinent performance measure for imbalanced data. Criteria such as ranking seem more appropriate when evaluating the CAD software. The CAD software suggests lesion candidates that typically contain many negative candidates. In practice, the user confirms only the top listed candidates. Thus, ranking the candidates is more critical than the accuracy of its classification of all candidates, positive or not. A natural criterion often used to measure the ranking quality of a classifier is the area under a receiver operating characteristic (ROC) curve (AUC) [13]. In this paper, we focus on evaluating the CAD software by using the AUC.

4.2 Negative transfer in OTL algorithm

Transfer learning is aimed to improve the prediction of the target domain classifier by incorporating certain information in the source domain. However, the source domain is not necessarily similar to the target domain. One known phenomenon is called *negative transfer* in which knowledge transferred from a source domain sometimes negatively affects learning in the target domain.

Ideally, the predictive performance of transfer learning must always be larger than that of single-domain learning in the target domain or source domain. In the beginning of training, when the amount of training data for the target domain is still small, transferring information from source domains is expected to improve predictive performance. However, the prediction error with transfer learning must not exceed the error by single-domain learning, even when the target task is well trained with sufficient training data.

Figure 1 illustrates the transition in predictive performance throughout the training process in the ideal case. The dotted black and blue lines indicate the predictive performances by using only source and target domain classifiers,

respectively. The solid red line in Figure 1 (a) shows predictive performance by *reasonable transfer learning,* where "reasonable" means that the predictive performance of transfer learning is always better than those of source and target domain classifiers.

If transfer learning does not perform better than either the source or target domain classifiers, as the solid red line in Figure 1 (b) illustrates, we define the algorithm as being affected by negative transfer. It is in fact crucial to construct an algorithm that does not have negative transfer of transfer learning for medical image analysis. We aimed to develop a transfer-learning algorithm with better predictive performance based on theoretical analysis of negative transfer, as in an ideal case represented by the solid red line in Figure 1 (a).

4.3 Formulating negative transfer on convexified AUC

We formulate negative transfer on CAD software based on an AUC evaluation. Denote by $\mathcal{D}$ a set of positive and negative examples: $\mathcal{D} = \{x_i^+\}_{i=1}^m \cup \{x_j^-\}_{j=1}^n$. Let $(h(x_1^+), h(x_2^+), \ldots, h(x_m^+))$ and $(h(x_1^-), h(x_2^-), \ldots, h(x_n^-))$ be the outputs of a classifier on the positive and negative examples, respectively. In the case of cerebral aneurysm detection, $\mathcal{D}$ is a set of lesion candidates extracted using CAD software for each case in which a positive example indicates a cerebral aneurysm site and a negative example indicates a normal site. The number of positive examples is much smaller than that of negative examples.

The AUC of classifier h and case $\mathcal{D}$ is given by

$$\mathrm{AUC}(h, \mathcal{D}) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} 1(h(x_i^+) > h(x_j^-)), \qquad (6)$$

where $1(\mathit{condition}) = 1$ if *condition* is satisfied, otherwise $1(\mathit{condition}) = 0$. The AUC is identical to the value of the Wilcoxon-Mann-Whitney test [7]. Thus, the AUC can be viewed as a measure based on pair wise comparisons between classifications of the two classes. We define a loss function based on the AUC as

$$\ell_{\mathrm{AUC}}(h, \mathcal{D}) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} 1(h(x_i^+) \leq h(x_j^-)), \qquad (7)$$

which we call *AUC loss*. Note that

$$\mathrm{AUC}(h, \mathcal{D}) = 1 - \ell_{\mathrm{AUC}}(h, \mathcal{D}). \qquad (8)$$

Since the AUC loss is non-convex, we approximate the AUC loss by using convex relaxation given by

$$\ell_{\mathrm{cAUC}}(h, \mathcal{D}) = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \max\{0, h(x_j^-) - h(x_i^+))\}, \qquad (9)$$

which we call *convexified AUC (cAUC) loss*. We also define

$$\mathrm{cAUC}(h, \mathcal{D}) = 1 - \ell_{\mathrm{cAUC}}(h, \mathcal{D}). \qquad (10)$$

The cAUC loss follows Assumption 1 (2).

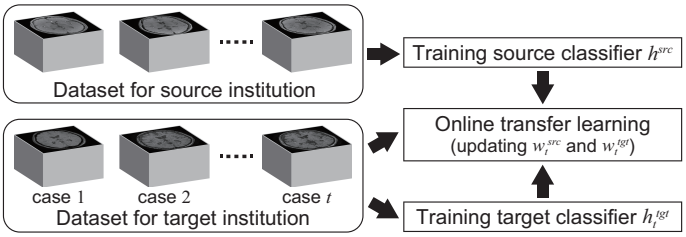

Figure 2: Flowchart of transfer learning in this paper.

The score function of the classifier typically takes a value in $[-1, 1]$. In this case, we normalize the output of classifier score functions by

$$h(x) = \max\{0, \min\{1, s(x)\}\} \ (s(x) \in [-1, 1]), \qquad (11)$$

which takes a value in $[0, 1]$. Since $\max\{0, h(x^-) - h(x^+))\} \in [0, 1]$, the cAUC loss follows Assumption 1 (1).

On the basis of Figure 1, as training in the target domain proceeds, prediction accuracy will improve, enabling the target domain classifier to outperform the source domain classifier. We assume that there exists a reasonable transfer that changes the source and target domain classifiers in accordance with their performance in the learning process. Therefore, the average cAUC of reasonable transfer learning is given by

$$\frac{1}{T} \max_{\tau} \left\{ \sum_{t=1}^{\tau} \mathrm{cAUC}(h_t^{\mathrm{src}}, \mathcal{D}_t), \sum_{t=\tau}^{T} \mathrm{cAUC}(h_t^{\mathrm{tgt}}, \mathcal{D}_t) \right\}. \qquad (12)$$

For simplicity, we denote

$$\mathrm{cAUC}_{[1,T]}^{\mathrm{cls}} = \sum_{t=1}^{T} \mathrm{cAUC}(h^{\mathrm{cls}}, \mathcal{D}_t), \ \mathrm{cls} \in \{\mathrm{src}, \mathrm{tgt}, \mathrm{trs}\}. \qquad (13)$$

We formulate negative transfer as follows.

Definition 2 (Negative transfer error). *The averaged negative transfer error between reasonable transfer learning and a transfer learning algorithm is given by*

$$\frac{1}{T} \max_{\tau} \left\{ \mathrm{cAUC}_{[1,\tau]}^{\mathrm{src}}, \mathrm{cAUC}_{[\tau,\mathrm{T}]}^{\mathrm{tgt}} \right\} - \frac{1}{T} \mathrm{cAUC}_{[1,\mathrm{T}]}^{\mathrm{trs}}. \qquad (14)$$

Next, we discuss our algorithm to minimize this negative transfer error.

4.4 Proposed algorithm

We formulated the negative transfer error in the previous section. In this section, we provide a learning algorithm to prevent the error.

Denote by ℓ_t^{cls} the cAUC loss of classifier $\mathrm{cls} \in \{\mathrm{trs}, \mathrm{src}, \mathrm{tgt}\}$ at round t:

$$\ell_t^{\mathrm{cls}} = \ell_{\mathrm{cAUC}}(h^{\mathrm{cls}}(x_t), \mathcal{D}_t)), \qquad (15)$$

which enables us to use the cAUC loss in the current OTL algorithm: Eqs. (4) and (5).

Proposed algorithm

Set $w_0^{\mathrm{src}} = 1$ and $w_0^{\mathrm{tgt}} = 0$.
Set T according to period to minimize negative transfer error.
Set $\rho = 1/(T-1)$ and $\eta = \sqrt{-\frac{8}{T}\log(\rho\,(1-\rho)^{T-2})}$.
for each round $t = 1, \cdots, T$ **do**
  Get images of case t, $\mathcal{D}_t$, and calculate cAUC loss for each algorithm.
  Update weights for each algorithm:
  $w_t^{\mathrm{src}} = (1-\rho) w_{t-1}^{\mathrm{src}} \exp(-\eta \ell_t^{\mathrm{src}}) + \rho w_{t-1}^{\mathrm{tgt}} \exp(-\eta \ell_t^{\mathrm{tgt}})$,
  $w_t^{\mathrm{tgt}} = (1-\rho) w_{t-1}^{\mathrm{tgt}} \exp(-\eta \ell_t^{\mathrm{tgt}}) + \rho w_{t-1}^{\mathrm{src}} \exp(-\eta \ell_t^{\mathrm{src}})$.
end for

Figure 3: Pseudo code of proposed algorithm.

Instead of Eqs. (4) and (5), we suggest updating the weights of the classifiers learned in the source and target domains as

$$w_t^{\mathrm{src}} = (1-\rho) w_{t-1}^{\mathrm{src}} \exp(-\eta \ell_t^{\mathrm{src}}) + \rho w_{t-1}^{\mathrm{tgt}} \exp(-\eta \ell_t^{\mathrm{tgt}}), \quad (16)$$

$$w_t^{\mathrm{tgt}} = (1-\rho) w_{t-1}^{\mathrm{tgt}} \exp(-\eta \ell_t^{\mathrm{tgt}}) + \rho w_{t-1}^{\mathrm{src}} \exp(-\eta \ell_t^{\mathrm{src}}), \quad (17)$$

where ρ and η are parameters, the settings of which are theoretically determined later. The interesting point of this algorithm is that the source and target domain classifiers are affected by loss, even if a classifier makes a correct prediction when another classifier makes a wrong prediction. The specific algorithm is given by Algorithm 1, where weights w_0^{src} and $w_0^{\mathrm{tgt}} = 1 - w_0^{\mathrm{src}}$ are the prior credibilities of the source and target domain classifiers and where $w_0^{\mathrm{src}} = 1$ and $w_0^{\mathrm{tgt}} = 0$ mean that the source domain classifier is much more credible than the target domain classifier in the initial learning process.

The algorithm provides the following theorem for the negative transfer error.

Theorem 3 (Negative transfer error bound). *Assume that $w_0^{src} = 1$ and $w_0^{tgt} = 0$, and*

$$\eta = \sqrt{\frac{8}{T}\log\frac{1}{p_0^*}}, \ \textit{where } p_0^* = \rho(1-\rho)^{T-2}. \quad (18)$$

The average negative transfer error of the algorithm based on weight updates Eqs. (16) *and* (17) *is bounded:*

$$\frac{1}{T}\max_{\tau}\left\{\mathrm{cAUC}_{[1,\tau]}^{\mathrm{src}}, \mathrm{cAUC}_{[\tau,\mathrm{T}]}^{\mathrm{tgt}}\right\} - \frac{1}{T}\,\mathrm{cAUC}_{[1,\mathrm{T}]}^{\mathrm{trs}} \le \sqrt{\frac{-\log p_0^*}{2T}}. \quad (19)$$

Proof. The details of the proof are in Appendixsec:proof. We used basic mathematical tools in expert prediction and mixed strategy [2, 8]. □

By minimizing the upper bound of the negative transfer error over $[1, T]$, we can decide parameter ρ.

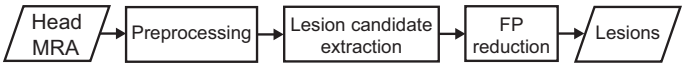

Figure 4: Flowchart of our cerebral aneurysm detection system.

Corollary 4. *Parameter ρ minimizing the upper-bound of the negative transfer error over $[1, T]$ is given by $\rho = \frac{1}{T-1}$.*

Proof. We define $u(\rho) = -\frac{1}{2T}\log(\rho(1-\rho)^{T-2})$. Taking the derivative of $u(\rho)$ and making it equal to zero, $\rho = \frac{1}{T-1}$ is a solution of minimizing the upper-bound of the negative transfer error. □

5 EXPERIMENTS

In this section, we discuss our experimental comparing the AUC performance of the proposed algorithm with that of the current OTL method in cerebral aneurysm detection. First, we explain CAD software we developed based on Adaboost. Then, we describe our datasets and experimental results.

5.1 CAD software based on Adaboost

We developed the CAD software for cerebral aneurysm detection in MRA data[16]. Figure 4 shows a flowchart of the detection algorithm . The system based on this framework is used daily in our hospital and other institutions. In preprocessing, intensity standardization was also carried out to correct scanner-dependent intensity variations. The equation for standardization can be described as follows:

$$I(\boldsymbol{x}) = \begin{cases} 0 & \text{if } I(\boldsymbol{x}) \le \mu_{brain} \\ \min\left(\frac{255(I(\boldsymbol{x})-\mu_{brain})}{\mu_{vessel}-\mu_{brain}}, 511\right) & \text{otherwise} \end{cases} \quad (20)$$

where $I(\boldsymbol{x})$ is the intensity of voxel p, where $\boldsymbol{x}$ is the 3D-coordinates for the position of p, μ_{brain} is the mean intensity of brain voxels, which is calculated from the center quarter region of the two central axial slices of the volume data, and μ_{vessel} is the mean intensity of the extracted vessel voxels. After lesion candidate detection based on a voxel-based classifier, the AdaBoost method [18] was used to calculate the likelihoods of the lesion candidates on the basis of 63 feature values.

5.2 Datasets and Evaluation

This study was approved by the ethical review board of all institutions. We used the following datasets (see Table 1 for details).

- Dataset for source institution: This dataset was collected from our hospital, UTH .
- Datasets for target institution: These datasets were collected from three institutions.

They contain the results obtained from our CAD software and evaluation data. Each positive case includes at least one aneurysm of 2 mm or more in diameter. The datasets for each institution were divided into training data and test data (see

Table 1 for details). All lesion candidates of the CAD software were subsequently inspected by at least one radiologist and classified into as TPs or FPs.

We evaluated the performance improvement of our cerebral aneurysm detection software for each institution based on transfer learning (Fig. 2). In this study, we focused on the classifier for FP reduction as shown in Fig. 4.

5.3 Training of source, target, and transfer learning classifiers

The source classifier h^{src} was trained using the UTH dataset. Since the data on lesion candidates were imbalanced, random undersampling was carried out to reduce the amount of FP data to the amount of TP data. Training with random undersampling and tests were repeated 100 times, and the best performance of the classifier was used as h^{src}.

The OTL algorithm and proposed algorithm work as a framework of online learning, whereas AdaBoost is a batch learning algorithm. Moreover, the data on lesion candidates were imbalanced. Consequently, the target classifier h_t^{tgt} and the classifier for transfer learning h_t^{trs} were trained as follows.

(1) w_t^{src} and w_t^{tgt} were updated using the OTL or proposed algorithm.
(2) h_t^{tgt} was trained using AdaBoost with both the training data and pooled past training data.

5.4 Settings

We compared the performance of the proposed algorithm with that of the OTL algorithm. We also evaluated the performance of a classifier, called $h_t^{src+tgt}$, which was trained using both of source and target data. The number of weak classifiers used for AdaBoost was set to 100 in the experiments. Regarding the OTL algorithm, the parameters w_t^{src} and η were set to 0.5 and 0.5, respectively, according to a previous study [27]. In the proposed algorithm, the number of cases to minimize negative transfer error, T, was set to 100 in accordance with the number of target training cases. To reduce the sampling effect, this procedures was repeated 100 times for each target institution. The AUC was used as the evaluation criterion. The number of aneurysms missed in the initial detection was included in calculating the AUC to evaluate overall performance of the CAD software.

5.5 Results

Figure 5 shows the results of the learning curve for each target institution. These curves showed only the first 50 target training cases with large variation. After the 50th target training case, the curves increased moderately or reached at plateau. The performance of $h_t^{src+tgt}$ depended on the target institution. For Hospital C, the performance of $h_t^{src+tgt}$ was superior to those of h^{src} and h_t^{tgt} However, for Clinics A and B, the performance of $h_t^{src+tgt}$ was inferior to that of h_t^{src} in the initial stage.

In the OTL algorithm, "negative transfer", which means that the performance of h_t^{trs} was inferior to that of h^{src}, was observed in the initial stage of transfer learning regardless of the target institution. Regarding the proposed algorithm, the performance of h_t^{trs} achieved "reasonable transfer" regardless of the target institution.

Theorem 3 and Corollary 4 indicate that when we set $T = 100$,

$$\frac{1}{T}\max_{\tau}\left\{\mathrm{cAUC}^{\mathrm{src}}_{[1,\tau]}, \mathrm{cAUC}^{\mathrm{tgt}}_{[\tau,\mathrm{T}]}\right\} - \frac{1}{T}\,\mathrm{cAUC}^{\mathrm{trs}}_{[1,\mathrm{T}]} \leq 0.1672.$$

This means that the upper-bound of performance deterioration due to the negative transfer is theoretically limited to 0.1672 by using the modified OTL algorithm. It is worth noting that this value is the upper-bound of performance deterioration, so it can take lower values in real data. In fact, $\frac{1}{T}\max_{\tau}\left\{\mathrm{cAUC}^{\mathrm{src}}_{[1,\tau]}, \mathrm{cAUC}^{\mathrm{tgt}}_{[\tau,\mathrm{T}]}\right\} < \frac{1}{T}\,\mathrm{cAUC}^{\mathrm{trs}}_{[1,\mathrm{T}]}$ has been experimentally shown .

6 DISCUSSION

The experimental results indicate that the performance degradation of the cerebral aneurysm detection software was inhibited even in the beginning of training with insufficient amount of training data. Since our algorithm uses only the trained source classifier and feature data for training the target classifier, it is suitable for improving the performance of CAD software at each institution with different distributions of feature values.When the number of target training cases became sufficient, the performance of our algorithm was almost equivalent to that of $h^{src+tgt}$ or the original OTL algorithm. However, in $h^{src+tgt}$ and the original OTL algorithm, negative transfer may occur in the beginning of training with insufficient amount of training data. Minimizing the possibility of negative transfer using our algorithm is useful for continuous performance improvement of CAD software.

The current OTL algorithm does not depend on the type of classifier in both domains. Although we used AdaBoost as both the source and target domain classifier in this study, different classifiers can be used for the source domain and target one. That is, we can use a deep neural net as a source classifier and AdaBoost (light classifier) as a target domain.

We can obtain optimal ρ by minimizing the negative transfer error from Corollary 4 providing period $[1, T]$ where T indicates the number of training positive cases in this experiment. This means that we implicitly assume that negative transfer occurs in that period. More precisely, T indicate the sufficient number of training positive cases with which the performance of the target-domain classifier exceeds that of the source-domain classifier. In this study, we used $T = 100$ because we know that this number of training positive cases is sufficient to learn AdaBoost in our hospital. The effective amount of training data typically depends on the type of dataset and property of machine learning algorithms. Therefore, the appropriate T should be set based on the type of classifier and period to collect sufficient positive cases for training the target classifier.

Table 1: Number of train and test cases in each dataset

Dataset	Institution	vendor	For	Number of cases		Number of candidates	
				Positive	Normal	Positive	Normal
source	UTH	GE (3.0 Tesla)	training	200	200	225	19,525
			test	50	50	53	5,325
target	Clinic A	GE (3.0 Tesla)	training	100	0*	117	1,843
			test	50	50	55	1,681
	Clinic B	Toshiba (1.5 Tesla)	training	100	0*	109	2,517
			test	50	50	51	2,713
	Hospital C	Philips (1.5 Tesla)	training	100	0*	106	2,789
			test	50	50	54	2,724

* Since each positive case includes lesion "candidates", it contains negative ($y = -1$: normal) candidates to train the target classifiers.

7 CONCLUSION

We proposed a modified OTL algorithm and investigated a strategy to improve the performance of CAD system or each institute. According to the experimental results, the performance degradation in the beginning of training was inhibited using our OTL algorithm. This modified OTL algorithm is suitable for improving the performance of CAD system at each institution. We evaluated only one type of CAD system based on Adaboost developed in our hospital. The results may not be applicable to other CAD software. Further evaluations using other CAD software may be necessary.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Shogo Nishiyama (Fuchinobe General Hospital, Kanagawa, Japan) for providing the target datasets of Hospital C. This work was supported by Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Number 15K01325. This work was supported by JST PRESTO Grant Number JPMJPR1302, Japan.

REFERENCES

[1] S. Belharbi, C. Chatelain, R. Herault, S. Adam, S. Thureau, M. Chastan, and R. Modzelewski. 2017. Spotting L3 slice in CT scans using deep convolutional network and transfer learning. *Computers in Biology and Medicine* 87 (2017), 95–103.

[2] Nicolo Cesa-Bianchi and Gabor Lugosi. 2006. *Prediction, Learning, and Games*. Cambridge University Press, New York, NY, USA.

[3] S. Conjeti, A. Katouzian, Roy A.G., L. Peter, F. Sheet, S. Carlier, A. Laine, and N. Navab. 2016. Supervised domain adaptation of decision forests: Transfer of models trained in vitro for in vivo intravascular ultrasound tissue characterization. *Medical Image Analysis* 32 (2016), 1–17.

[4] K.L. Giger, H.P. Chan, and J. Boone. 2008. Anniversary paper: History and status of CAD and quantitative image analysis: the role of Medical Physics and AAPM. *Medical Physics* 35, 12 (2008), 5799–5820.

[5] H. Greenspan, B. van Ginneken, and R.M. Summers. 2016. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging* 35, 5 (2016), 1153–1159.

[6] N.P. Gruszauskas, K. Drukker, M.L. Giger, R.F. Chang, C.A. Sennett, W.K. Moon, and L.L. Pesce. 2009. Breast US computer-aided diagnosis system: Robustness across urban populations in South Korea and the United States. *Radiology* 253, 3 (2009), 661–671.

[7] J. A. Hanley and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* 143, 1 (1982), 29–36.

[8] M. Herbster and M. K. Warmuth. 1998. Tracking the best expert. *Machine Learning* 32, 2 (1998), 151–178.

[9] P.S. Jialin and Q. Yang. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (2010), 1345–1359.

[10] Y. LeCun, Y. Bengio, and G. Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.

[11] H. Li, M.L. Giger, Y. Yuan, W. Chen, K. Horsch, L. Lan, A.R. Jamieson, C.A. Sennett, and S.A. Jansen. 2008. Evaluation of computer-aided diagnosis on a large clinical full-field digital Mammographic Dataset. *Academic Radiology* 15, 11 (2008), 1437–1445.

[12] Nick Littlestone and Manfred K. Warmuth. 1994. The weighted majority algorithm. *Radiology* 108, 2 (1994), 212–261.

[13] Michael C. Mozer, Robert H. Dodier, Michael D. Colagrosso, Cesar Guerra-Salcedo, and Richard H. Wolniewicz. 2001. Prodding the ROC Curve: Constrained Optimization of Classifier Performance. In *Advances in Neural Information Processing Systems 14*. 1409–1415.

[14] Y. Nomura, N. Hayashi, Y. Masutani, T. Yoshikawa, M. Nemoto, S. Hanaoka, S. Miki, E. Maeda, and K. Ohtomo. 2010. CIRCUS: an MDA platform for clinical image analysis in hospitals. *Transactions on Mass-Data Analysis of Images and Signals* 2, 1 (2010), 112–127.

[15] Y. Nomura, Y. Masutani, S. Miki, S. Hanaoka, M. Nemoto, T. Yoshikawa, N. Hayashi, and K. Ohtomo. 2013. Training Strategy for Performance Improvement in Computer-Assisted Detection of Lesions: Based on Multi-institutional Study in Teleradiology Environment. In *2013 First International Symposium on Computing and Networking*. 320–323.

[16] Y. Nomura, Y. Masutani, S. Miki, M. Nemoto, S. Hanaoka, T. Yoshikawa, N. Hayashi, and K. Ohtomo. 2014. Performance improvement in computerized detection of cerebral aneurysms by retraining classifier using feedback data collected in routine reading environment. *Journal of Biomedical Graphics and Computing* 4, 4 (2014), 12–21.

[17] R. K. Samala, H. P. Chan, L. Hadjiiski, M. A. Helvie, J. Wei, and K. Cha. 2016. Mass detection in digital breast tomosynthesis: Deep convolutional neural network with transfer learning from mammography. *Medical Physics* 43, 12 (2016), 6654–6666.

[18] R.E. Schapire, Y. Freund, P. Bartlett, and W.S. Lee. 1998. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics* 26, 5 (1998), 1651–1686.

[19] H.C. Shin, H.R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R.M. Summers. 2016. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Transactions on Medical Imaging* 35, 5 (May 2016), 1285–1298.

[20] S. Sonoyama, T. Hirakawa, T. Tamaki, T. Kurita, B. Raytchev, K. Kaneda, T. Koide, S. Yoshida, Y. Kominami, and S. Tanaka. 2015. Transfer learning for Bag-of-Visual words approach to NBI

endoscopic image classification. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 785–788.
[21] R.M. Summers, L.R. Handwerker, P.J. Pickhardt, R.L. van Uitert, K.K. Deshpande, S. Yeshwant, J. Yao, and M. Franaszek. 2008. Performance of a previously validated CT colonography computer-aided detection system in a new patient population. *AJR American Journal of Roentgenology* 191, 1 (2008), 168–174.
[22] N. Tajbakhsh, J.Y. Shin, S.R. Gurudu, R.T. Hurst, C.B. Kendall, M.B. Gotway, and J. Liang. 2016. Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Transactions on Medical Imaging* 35, 5 (May 2016), 1299–1312.
[23] A. van Engelen, A.C. van Dijk, M.T.B. Truijman, R. van't Klooster, A. van Opbroek, A. van der Lugt, W.J. Niessen, M.E. Kooi, and M. de Bruijne. 2015. Multi-center MRI carotid plaque component segmentation using feature normalization and transfer learning. *IEEE Transactions on Medical Imaging* 34, 6 (June 2015), 1294–1305.
[24] B. van Ginneken, C.M. Schaefer-Prokop, and M. Prokop. 2011. Computer-aided diagnosis: how to move from the laboratory to the clinic. *Radiology* 261, 3 (2011), 719–732.
[25] A. van Opbroek, M.A. Ikram, M.W. Vernooij, and M. de Bruijne. 2015. Transfer learning improves supervised image segmentation across imaging protocols. *IEEE Transactions on Medical Imaging* 34, 5 (2015), 1018–1030.
[26] W.A. Willinek, M. Born, Simon B., H.J. Tschampa, C. Krautmacher, Gieseke J., H. Urbach, H.J. Textor, and H.H. Schild. 2003. Time-of-flight MR angiography: comparison of 3.0-T imaging and 1.5-T imaging-initial experience. *Radiology* 229, 3 (2003), 913–920.
[27] P. Zhao and S.C.H. Hoi. 2010. OTL: a framework of online transfer learning. In *Proceedings of the 27th International Conference on Machine Learning*. 1231–1238.

A PROOF OF THEOREM 3

Define

$$L^{\mathrm{cls}}_{[t_1,t_2]} = \sum_{t=t_1}^{t_2} \ell_{\mathrm{cAUC}}(h^{\mathrm{cls}}_t, \mathcal{D}_t),\ \mathrm{cls} \in \{\mathrm{src}, \mathrm{tgt}, \mathrm{trs}\}. \quad (21)$$

Theorem 3 is identical to the following theorem.

THEOREM 5. *Assume that* $w^{src}_0 = 1$ *and* $w^{tgt}_0 = 0$*, and*

$$\eta = \sqrt{\frac{8}{T}\log\frac{1}{p^*_0}},\ \textit{where}\ p^*_0 = \rho(1-\rho)^{T-2}. \quad (22)$$

The averaged negative transfer error of the algorithm based on weight updates (16) *and* (17) *is bounded as*

$$\frac{1}{T}L^{trs}_{[1,T]} - \frac{1}{T}\min_{\tau}\{L^{src}_{[1,\tau]} + L^{tgt}_{[\tau,T]}\} \le \sqrt{\frac{1}{2T}\log\frac{1}{p^*_0}}. \quad (23)$$

Theorem 5 holds as a consequence of Lemmas 7 and 8 . Our main analysis tool is to use the properties of the EWA algorithm. First, we provide the general result of the EWA algorithm then we apply the result to our problem.

Let $\hat{L}_T = \sum_{t=1}^T \ell(\hat{h}_t, y_t)$ be the cumulative loss of the EWA algorithm given by Eq. (2), and let $L_{k,T} = \sum_{t=1}^T \ell(h_{k,t}, y_t)$ be the cumulative loss of expert k. Denote by $p_{k,0}$ the prior credible probability of expert k where $\sum_{k=1}^K p_{k,0} = 1,\ p_{k,0} \ge 0$ and $w_{k,0} = p_{k,0}$.

THEOREM 6 ([2] P.37). *For any* $\eta > 0$*, the EWA algorithm holds that*

$$\hat{L}_T - \min_{k\in 1,\dots,K} L_{k,T} \le \frac{1}{\eta}\log\frac{1}{p_{k^*,0}} + \frac{T}{8}\eta, \quad (24)$$

where $k^* = \operatorname{argmin}_{k\in 1,\dots,K} L_{k,T}$.

The left side $\hat{L}_T - \min_{k\in 1,\dots,K} L_{k,T}$ is called *Regret* in online learning. The probability, $p_{k,0}$, acts on how much we believe in expert k initially. That is, this theorem means that when we can assign a high probability on expert k^* initially, the regret will be small.

We introduce *compound expert* defined by mixed strategy $\boldsymbol{s} = (s_1, s_2, \dots)$, which is a sequence of base experts [8]. In this study, the base experts were the source and target domain classifiers. The strategy with $s_t = \mathrm{src/tgt}$ indicates that we believe that the performance of the source/target domain classifier is better than that of the target/source domain classifier at round t. That is, the prediction of mixed strategy $\boldsymbol{s}$ at round t, $h_{\boldsymbol{s},t}$, is given by

$$h_{\boldsymbol{s},t}(x) = h^{s_t}_t(x),\ s_t \in \{\mathrm{src}, \mathrm{tgt}\}. \quad (25)$$

The reasonable transfer in Figure 1 (a) takes a strategy that $s_t = \mathrm{src}$ for $1 \le t \le \tau*$ and $s_t = \mathrm{tgt}$ for $\tau* \le t$, where the performance of the source and target domain classifers changes at round $\tau*$.

Defining by $w_t(\boldsymbol{s})$ the weight of mixed strategy $\boldsymbol{s}$ at round t, the EWA algorithm for mixed strategy $\boldsymbol{s}$ updates the weight by

$$w_t(\boldsymbol{s}) = w_{t-1}(\boldsymbol{s})\exp(-\eta\ell_{\mathrm{cAUC}}(h^{s_t}_t, \mathcal{D}_t)), \quad (26)$$

and predicts

$$\widehat{h}_{t+1}(x) = \sum_{\boldsymbol{s}} p_t(\boldsymbol{s}) h_{\boldsymbol{s},t}(x),\ p_t(\boldsymbol{s}) = \frac{w_t(\boldsymbol{s})}{\sum_{\boldsymbol{s}'} w_t(\boldsymbol{s}')}, \quad (27)$$

where $\sum_{\boldsymbol{s}}$ is the summation over all combination of mixed strategy $\boldsymbol{s}$ and where $w_0(\boldsymbol{s}) = p_0(\boldsymbol{s})$ is a credible probability of mixed strategy $\boldsymbol{s}$ in the beginning.

Let $\mathcal{S}$ be a set of candidates for transfer strategy:

$$\mathcal{S}_T = \{\boldsymbol{s} | s_t = \mathrm{src}\ (1 \le t \le \tau),\ s_t = \mathrm{tgt}\ (\tau \le t \le T),\ \tau \ge 1\}. \quad (28)$$

That is, the optimization problem for reasonable transfer

$$\min_{\tau}\left\{\sum_{t=1}^{\tau}\ell_{\mathrm{cAUC}}(h^{\mathrm{src}}_t, \mathcal{D}_t) + \sum_{t=\tau}^{T}\ell_{\mathrm{cAUC}}(h^{\mathrm{tgt}}_t, \mathcal{D}_t)\right\}$$

is reformulated as

$$\min_{\boldsymbol{s}\in\mathcal{S}_T}\sum_{t=1}^{T}\ell_{\mathrm{cAUC}}(h^{s_t}_t, \mathcal{D}_t). \quad (29)$$

We define the cumulative loss of mixed strategy $\boldsymbol{s}$:

$$L_T(\boldsymbol{s}) = \sum_{t=1}^{T}\ell_{\mathrm{cAUC}}(h^{s_t}_t, \mathcal{D}_t), \quad (30)$$

and $\widehat{L}_T$ is the cumulative loss of the classifier defined in Eqs. (26) and (27).

Lemma 7. *If* $\eta = \sqrt{\frac{8}{T}\log\frac{1}{p_0(\boldsymbol{s}^*)}}$, *then*

$$\widehat{L}_T - \min_{\boldsymbol{s}\in\mathcal{S}_T} L_T(\boldsymbol{s}) \leq \sqrt{\frac{T}{2}\log\frac{1}{p_0(\boldsymbol{s}^*)}}. \tag{31}$$

Proof. By applying Theorem 6 for the EWA algorithm for mixed strategy, for any $\eta > 0$, the EWA algorithm for mixed strategy holds that

$$\widehat{L}_T - \min_{\boldsymbol{s}\in\mathcal{S}_T} L_T(\boldsymbol{s}) \leq \frac{1}{\eta}\log\frac{1}{p_0(\boldsymbol{s}^*)} + \frac{T}{8}\eta, \tag{32}$$

where $\boldsymbol{s}^* = \operatorname*{argmin}_{\boldsymbol{s}\in\mathcal{S}_T} L_T(\boldsymbol{s})$. □

Therefore, we show in the next step that the algorithm composed of update Eq. (26) and prediction Eq. (27) is equivalent to the proposed algorithm composed of update Eqs., (16), (17) and prediction Eq. (3). That is, we show the following lemma.

Lemma 8.

$$\begin{aligned} w_t^{src} =& (1-\rho)w_{t-1}^{src}\exp(-\eta\ell_t(s_t = src)) \\ &+ \rho w_{t-1}^{tgt}\exp(-\eta\ell_t(s_t = tgt)), \end{aligned} \tag{33}$$

$$\begin{aligned} w_t^{tgt} =& (1-\rho)w_{t-1}^{src}\exp(-\eta\ell_t(s_t = src)) \\ &+ \rho w_{t-1}^{tgt}\exp(-\eta\ell_t(s_t = tgt)). \end{aligned} \tag{34}$$

Proof. We use the notation "$\boldsymbol{s} : s_t = \mathrm{src}, s_{t+1} = \mathrm{tgt}$," which means that strategy $\boldsymbol{s}$ takes $s_t = \mathrm{src}$ and $s_{t+1} = \mathrm{tgt}$ at rounds t and $t+1$.

The first equation is given by

$$\begin{aligned} w_t^{\mathrm{src}} &= \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src}} w_t(\boldsymbol{s}) = \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src}} w_0(\boldsymbol{s})\exp(-\eta L_t(\boldsymbol{s})) \\ &= \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src}} w_0(\boldsymbol{s})\exp(-\eta\ell_t(s_t))\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &= \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src},s_t=\mathrm{src}} w_0(\boldsymbol{s})\exp(-\eta\ell_t(s_t))\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &\quad + \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src},s_t=\mathrm{tgt}} w_0(\boldsymbol{s})\exp(-\eta\ell_t(s_t))\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &= \exp(-\eta\ell_t(\mathrm{src})) \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src},s_t=\mathrm{src}} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &\quad + \exp(-\eta\ell_t(\mathrm{tgt})) \sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src},s_t=\mathrm{tgt}} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &\quad (\Downarrow \text{Lemma 9}) \\ &= \exp(-\eta\ell_t(\mathrm{src}))(1-\rho)w_{t-1}^{\mathrm{src}} + \exp(-\eta\ell_t(\mathrm{tgt}))\rho w_{t-1}^{\mathrm{tgt}} \\ &= (1-\rho)w_{t-1}^{\mathrm{src}}\exp(-\eta\ell_t(\mathrm{src})) + \rho w_{t-1}^{\mathrm{tgt}}\exp(-\eta\ell_t(\mathrm{tgt})). \end{aligned} \tag{35}$$

The first equation is given in a similar way to the first one. □

Lemma 9.

$$\sum_{\boldsymbol{s}:s_{t+1}=src,s_t=tgt} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) = \rho w_{t-1}^{tgt}, \tag{36}$$

$$\sum_{\boldsymbol{s}:s_{t+1}=src,s_t=src} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) = (1-\rho)w_{t-1}^{src}, \tag{37}$$

$$\sum_{\boldsymbol{s}:s_{t+1}=tgt,s_t=tgt} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) = (1-\rho)w_{t-1}^{tgt}, \tag{38}$$

$$\sum_{\boldsymbol{s}:s_{t+1}=tgt,s_t=src} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) = \rho w_{t-1}^{src}. \tag{39}$$

Proof. Equation (36) is given by

$$\begin{aligned} &\sum_{\boldsymbol{s}:s_{t+1}=\mathrm{src},s_t=\mathrm{tgt}} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &= \sum_{\boldsymbol{s}\setminus\{s_{t+1},s_t\}} w_0(\boldsymbol{s} : s_{t+1} = \mathrm{src}, s_t = \mathrm{tgt})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &(\Downarrow \text{Lemma 10}) \\ &= \sum_{\boldsymbol{s}\setminus\{s_t\}} (1-\rho)w_0(\boldsymbol{s} : s_t = \mathrm{tgt})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &= \rho \sum_{\boldsymbol{s}:s_t=\mathrm{tgt}} w_0(\boldsymbol{s})\exp(-\eta L_{t-1}(\boldsymbol{s})) \\ &= \rho w_{t-1}^{\mathrm{tgt}}. \end{aligned} \tag{40}$$

Equations (37), (38) and (39) are given in a similar way to Eq. (36). □

Lemma 10.

$$w_0(\boldsymbol{s} : s_{t+1} = src, s_t = src) = (1-\rho)w_0(\boldsymbol{s} : s_t = src), \tag{41}$$

$$w_0(\boldsymbol{s} : s_{t+1} = src, s_t = tgt) = \rho w_0(\boldsymbol{s} : s_t = tgt), \tag{42}$$

$$w_0(\boldsymbol{s} : s_{t+1} = tgt, s_t = src) = \rho w_0(\boldsymbol{s} : s_t = src), \tag{43}$$

$$w_0(\boldsymbol{s} : s_{t+1} = tgt, s_t = tgt) = (1-\rho)w_0(\boldsymbol{s} : s_t = tgt). \tag{44}$$

Proof. Equation (41) is given by

$$\begin{aligned} &w_0(\boldsymbol{s} : s_{t+1} = \mathrm{src}, s_t = \mathrm{src}) \\ &= \frac{w_0(\boldsymbol{s} : s_{t+1} = \mathrm{src}, s_t = \mathrm{src})}{w_0(\boldsymbol{s} : s_t = \mathrm{src})} w_0(\boldsymbol{s} : s_t = \mathrm{src}) \\ &= \frac{p_0(\boldsymbol{s} : s_{t+1} = \mathrm{src}, s_t = \mathrm{src})}{p_0(\boldsymbol{s} : s_t = \mathrm{src})} w_0(\boldsymbol{s} : s_t = \mathrm{src}) \\ &= p_0(s_{t+1} = \mathrm{src}|s_t = \mathrm{src})w_0(\boldsymbol{s} : s_t = \mathrm{src}) \\ &= (1-\rho)w_0(\boldsymbol{s} : s_t = \mathrm{src}). \end{aligned} \tag{45}$$

Equations (42), (43), and (44) are given in a similar way to Eq. (41). □

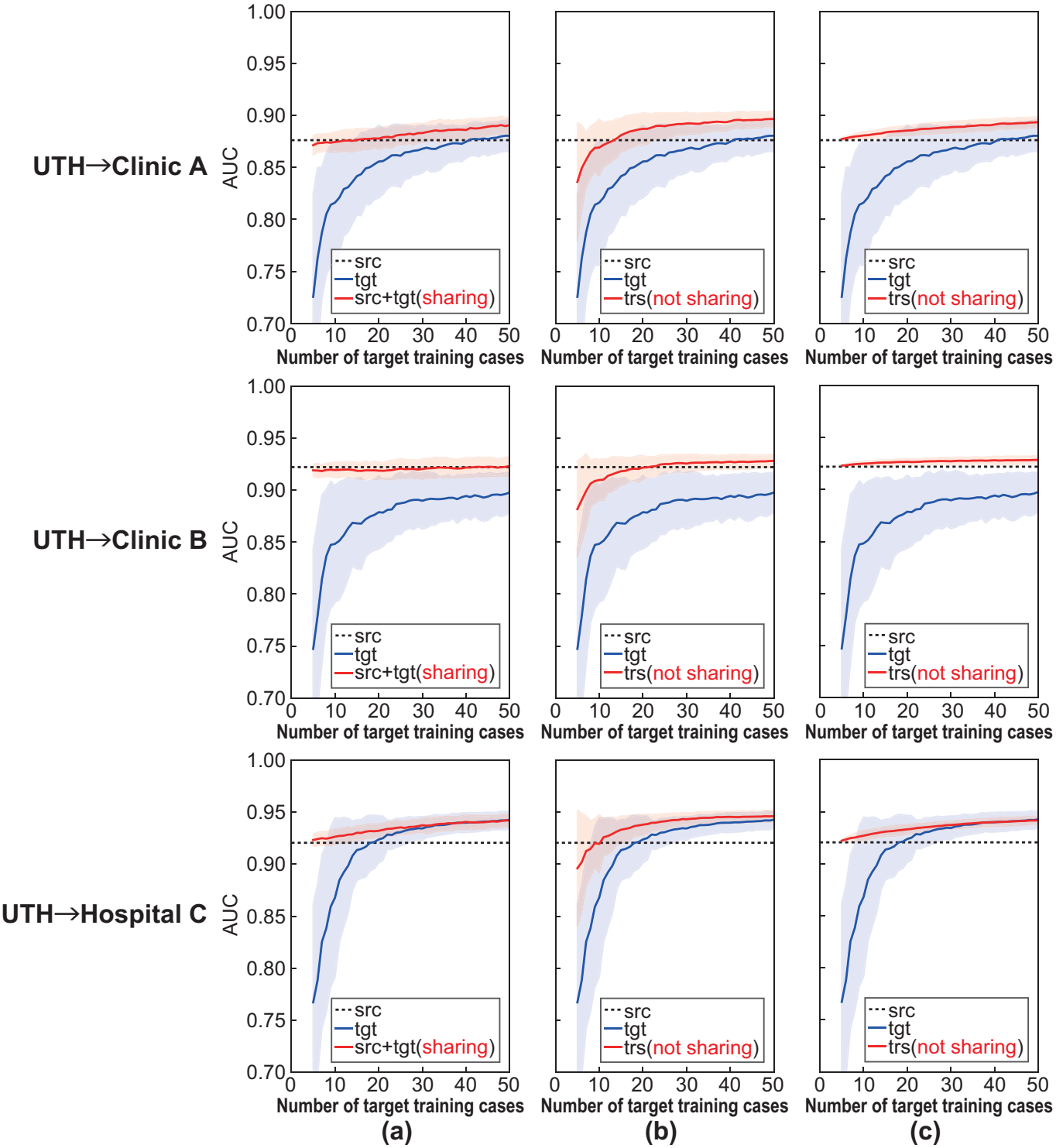

Figure 5: Results of learning curve. (a) $h_t^{src+tgt}$, (b) OTL, (c) proposed algorithm. From top to bottom: target: Clinic A, target: Clinic B, and target: Hospital C. Solid line indicates average AUC, and the shaded area shows range of $\pm$ one standard deviation.