Weakly-Supervised Transfer Learning With Application in Precision Medicine

Lingchao Mao, *Member, IEEE*, Lujia Wang, *Member, IEEE*, Leland S. Hu, Jenny M. Eschbacher, Gustavo De Leon, Kyle W. Singleton, Lee A. Curtin, Javier Urcuyo, Chris Sereduk, Nhan L. Tran, Andrea Hawkins-Daarud, Kristin R. Swanson, and Jing Li, *Member, IEEE*

***Abstract*— Precision medicine aims to provide diagnosis and treatment accounting for individual differences. To develop machine learning models in support of precision medicine, personalized models are expected to have better performance than one-model-fits-all approaches. A significant challenge, however, is the limited number of labeled samples that can be collected from each individual due to practical constraints. Transfer Learning (TL) addresses this challenge by leveraging the information of other patients with the same disease (i.e., the source domain) when building a personalized model for each patient (i.e., the target domain). We propose Weakly-Supervised Transfer Learning (WS-TL) to tackle two challenges that existing TL algorithms do not address well: (i) the target domain has only a few or even no labeled samples; (ii) how to integrate domain knowledge into the TL design. We design a novel mathematical framework of WS-TL to learn a model for the target domain based on paired samples whose order relationships are inferred from domain knowledge, while at the same time integrating labeled samples in the source domain for transfer learning. Also, we propose an efficient active sampling strategy to select informative paired samples. Theoretical properties were investigated. Finally, we present a real-world application in precision medicine of brain cancer, where WS-TL is used to build personalized patient models to predict Tumor Cell Density (TCD) distribution across the brain based on MRI images. WS-TL has the highest accuracy compared to a variety of existing TL algorithms. The predicted TCD map for each patient can help facilitate individually optimized treatment.**

***Note to Practitioners*—This work was motivated by Precision Medicine applications that need to build personalized machine learning models to account for individual differences. Due to limited data from each person, Transfer Learning (TL) provides a promising approach, which can leverage the information of other patients with the same disease (i.e., the source domain) when building a personalized model for each patient (i.e., the target domain). The proposed WS-TL model addresses the application scenarios with two unique properties: (i) the target domain has a few and even no labeled samples, which is a challenging situation that most existing TL methods do not address well; (ii) there is domain knowledge to provide weak labels for a large number of unlabeled samples in the form of order relationships, which provides an opportunity to integrate the domain knowledge into the TL design. We demonstrate WS-TL in a Precision Medicine application for brain cancer and show promising results. WS-TL has the potential of addressing a broad range of other application areas in building personalized models.**

***Index Terms*— Machine learning, statistical modeling, health care, precision medicine.**

Manuscript received 13 July 2022; revised 2 November 2022; accepted 19 December 2022. Date of publication 23 October 2023; date of current version 16 October 2024. This article was recommended for publication by Associate Editor H. Kim and Editor X. Xie upon evaluation of the reviewers' comments. This work was supported in part by NSF under Grant DMS-2053170 and in part by NIH under Grant U01 CA220378-01. *(Corresponding author: Jing Li.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Barrow Neurological Institute (BNI) Institutional review board approved protocol "Improving Diagnostic Accuracy in Brain Patients Using Perfusion MRI".

Lingchao Mao, Lujia Wang, and Jing Li are with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: lmao31@gatech.edu; lwang724@gatech.edu; jli3175@gatech.edu).

Leland S. Hu is with the Department of Radiology, Mayo Clinic Arizona, Phoenix, AZ 85054 USA (e-mail: Hu.Leland@mayo.edu).

Jenny M. Eschbacher is with the Department of Pathology, Barrow Neurological Institute-St. Joseph's Hospital and Medical Center, Phoenix, AZ 85013 USA (e-mail: jennifer.eschbacher@commonspirit.org).

Gustavo De Leon, Kyle W. Singleton, Lee A. Curtin, Javier Urcuyo, Chris Sereduk, Andrea Hawkins-Daarud, and Kristin R. Swanson are with the Mathematical Neuro-Oncology Laboratory, Department of Neurosurgery, Mayo Clinic Arizona, Phoenix, AZ 85054 USA (e-mail: Leon.Gustavo@mayo.edu; Singleton.Kyle@mayo.edu; Curtin.Lee@mayo.edu; Urcuyo.Javier@mayo.edu; Sereduk.Christopher@mayo.edu; Hawkins-Daarud.Andrea@mayo.edu; Swanson.Kristin@mayo.edu).

Nhan L. Tran is with the Department of Cancer Biology, Mayo Clinic Arizona, Phoenix, AZ 85054 USA (e-mail: Tran.Nhan@mayo.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TASE.2023.3323773.

Digital Object Identifier 10.1109/TASE.2023.3323773

I. Introduction

PRECISION medicine aims to provide diagnosis and treatment accounting for individual differences. To develop machine learning models in support of precision medicine, personalized or patient-specific models are expected to have better performance than one-model-fits-all approaches. A significant challenge, however, is the limited number of labeled samples for each individual due to cost, availability, and other practical constraints.

To tackle this challenge, Transfer Learning (TL) provides a promising approach. TL is a subfield in machine learning, which aims to transfer the information learned from related source domains to help build a model for a target domain. TL has been used to build personalized models in the medical field, such as heart rate failure prediction [1], tumor classification [2], and seizure detection [3]. In these applications, the target domain is each patient of interest whereas the source domain includes other patients with the same or similar disease.

In this paper, we focus on addressing two limitations of the existing TL methods. First, even though TL can leverage the information in the source domain, most existing TL algorithms still need the target domain to have a non-trivial number of labeled samples, although not enough to build a robust model on their own, but enough to bias the model of the source toward the target. This limits the application of TL to areas where the target domain has only a few or even no labeled samples. Second, most existing TL algorithms are purely data-driven, whereas domain knowledge exists in many applications. Integration of domain knowledge into the TL design holds great promise to improve the model performance. In this paper, we focus on the type of domain knowledge that can provide weak labels for a large number of unlabeled samples.

Next, we give a motivating example to further illustrate the research question we aim to address:

A motivating example in precision medicine of brain cancer: Glioblastoma (GBM) is the most aggressive type of brain cancer with a median survival of only 15 months. There is a significant region-to-region variation of Tumor Cell Density (TCD) —the percentage of tumor cells within a regional tissue sample— across the brain. It is important to know the regional TCD so that treatment can be optimized, i.e., regions with higher TCD can be treated more aggressively to prevent tumor growth whereas regions with lower TCD can be treated less aggressively to avoid over-damaging of the brain [4]. To know the TCD of a specific region, the gold-standard approach is to acquire a biopsy sample from that region and obtain TCD measurement through histopathologic analysis. However, due to the invasive nature of biopsy, only a few biopsy samples can be acquired from each patient, leaving many regions where TCD remains unknown. To tackle this challenge, a machine learning model can be trained to predict regional TCD based on imaging (e.g., MRI) features extracted from the same region, $f : \boldsymbol{x} \rightarrow y$, where $\boldsymbol{x}$ contains the imaging features and y is the regional TCD. Once trained, the model can be used to predict the TCD of any unbiopsied region using the imaging features from that region. Since imaging is non-invasive and can portray the whole brain, using the trained machine learning model allows for generating a predictive TCD map for each patient to guide individualized treatment.

Recent findings have provided strong evidence of both interpatient heterogeneity in GBM as well as intralesional heterogeneity within a single GBM tumor [5], [6], [7], calling for a patient-specific model to link imaging features with regional TCD. However, as aforementioned, each patient has only a few biopsy samples with TCD measurement (on average four biopsy samples per patient in our case study), prohibiting the training of a robust model using each patient's data alone. Using TL to transfer the information of other patients (as the source domain) to help build the model for each patient (as the target domain) represents a step in the right direction. However, the patient-wise sample size is still too small for most existing TL algorithms to be applicable. On the other hand, we can potentially leverage a large number of weakly labeled samples in training to compensate for the biopsy/labeled sample shortage, where the weak labels are provided by domain knowledge. Specifically, domain knowledge in cancer biology and imaging physics can help pinpoint certain areas of the brain, $\mathcal{A}_h$, $\mathcal{A}_l$, such that samples from $\mathcal{A}_h$ are likely to have higher TCD than those from $\mathcal{A}_l$. That is, suppose $\boldsymbol{x}_1 \in \mathcal{A}_h$ and $\boldsymbol{x}_2 \in \mathcal{A}_l$ are two samples from the two areas, respectively. Then, $y_1 > y_2$ holds according to the domain knowledge, even though these two samples are not biopsy samples so that their exact TCD values, y_1 and y_2, are unknown —the reason why they are called weakly labeled samples.

To summarize, the goal of this paper is to develop a new TL method, called Weakly Supervised Transfer Learning (WS-TL), to address the application scenarios with two unique properties: (1) the target domain has a few and even no labeled samples; (2) there is domain knowledge to provide weak labels for a large number of unlabeled samples in the form of order relationships.

Although our model was motivated by the need to build personalized models for precision medicine, it can provide value in other science and engineering domains that share the above-mentioned properties. For example, in forest fire management [8], direct fire risk measurements through aerial or ground inspection can only be taken from a few locations due to resource constraints. Domain knowledge about vegetation type or wind direction may indicate that some regions are at higher risk than others, which can be used to create weakly labeled samples to aid the forest fire prediction. Plant phenotyping is another example parallel to precision medicine but for plants [9]. Automated imaging technologies have been used to monitor plant physiology and crop yield. However, in-depth phenotyping at cellular level still requires time-consuming manual measurements taken from leaf surfaces. Applying knowledge about how environmental conditions may influence the growing process to the data collected from environmental sensors can be used to generate weakly labeled samples to aid the prediction of growth phenotypes.

The contributions of this paper are summarized as follows:

- **New TL model**: We design a novel optimization framework of WS-TL to learn a model for the target domain based on paired samples whose order relationships are inferred from domain knowledge, while at the same time integrating labeled samples in the source domain for transfer learning. We develop an Alternating Optimization algorithm to solve the WS-TL formulation with convergence guarantee. We conduct theoretical analysis to reveal beneficial properties of WS-TL such as solution sparseness and robustness.
- **Integration with efficient active sampling strategy**: We propose a novel strategy to select informative paired samples included in WS-TL training, called Active Sampling based on Maximal Model Change (AS-MMC). We conduct theoretical analysis which demonstrates a faster convergence of WS-TL integrated with AS-MMC than random sampling.
- **Contribution to Precision Medicine of brain cancer**: We show the application of WS-TL to a real-world case study for predicting the regional TCD for patients with GBM. WS-TL builds personalized models for each patient and generates predictions with higher accuracy than a variety of competing methods. The results show the

potential of using WS-TL to facilitate precision treatment of GBM tumors.

II. Related Work

Our proposed method is related to two subfields in machine learning: TL and Weakly Supervised Learning (WSL). In this section, we review the existing methods in each subfield and point out the difference in our method.

A. Transfer Learning (TL)

The existing TL methods fall into three main categories: instance transfer, feature transfer, and parameter transfer.

Instance transfer aims to correct the distribution difference between the source and target domains by reweighting the samples. Algorithms differ in terms of their reweighting strategy. For example, Weighted-SVM [10] assigns domain-level weights to source and target samples. Kernel Mean Matching (KMM) [11] reweights source samples based on reduction in Maximum Mean Discrepancy (MMD), a widely used metric in TL to measure distribution difference, and then trains the model using reweighted samples. TrAdaBoost [12] adjusts weights iteratively based on the error of sequentially learned models. Jiang and Zhai [13] incorporated weighted unlabeled samples by assigning them pseudo labels using an auxiliary model trained on labeled samples.

Feature transfer aims to find a feature mapping where the distribution difference between the source and target domains is minimized. For example, Pan et al. [14] proposed to learn a transformation matrix that minimizes the marginal distribution difference measured by MMD and then use principal component analysis to find a lower-dimensional representation. These two steps are later integrated into a unified algorithm called Transfer Component Analysis [15]. TCA is by far one of the most cited methods in the TL literature due to its ability to require only unlabeled samples from the target domain for transfer learning and demonstrated good performance in various applications. Several methods are built upon TCA such as Joint Distribution Adaptation [16] and Adaptation Regularization-based Transfer Learning [17]. These methods either require labeled samples from the target domain or create pseudo labels assigned to unlabeled target samples. There is a risk that pseudo labels may be incorrect when there is a substantial distribution difference between the domains.

Parameter transfer approaches use a pre-trained source model and adapt its parameters to fit the target domain. A-SVM [18] aims to learn a delta function that represents the gap between source and target model parameters. Yang et al. [19] further improved A-SVM by directly regularizing the difference of source and target model parameters, called Adapt-SVM. Duan et al. [20] extended A-SVM into a general formulation that can be used for multi-source problems called Domain Adaptation Machine. Also, ensemble-based methods have been developed to combine outputs of multiple source models with those of the target [21], [22]. In summary, parameter transfer approaches can tackle problems with conditional distribution differences between the source and target domains. However, these methods require enough labeled data from the target domain to adapt the source model.

Our work is different from the existing TL methods in the following way: We target the more challenging situation that the target domain has a few and even no labeled samples. Also, most existing TL algorithms are purely data-driven, whereas domain knowledge exists in many applications. We propose to integrate domain knowledge, which can be used to create weak labels for a large number of unlabeled samples in the form of order relationships into the TL design. This can lead to greater interpretability and sample efficiency. Furthermore, while the majority of existing TL methods focus on classification problems, our proposed WS-TL focuses on regression-type of problems with continuous response variables, which is more relevant to Precision Medicine applications like the one in the motivating example.

B. Weakly Supervised Learning (WSL)

Our work also intersects with WSL, which builds machine learning models by incorporating samples with incomplete, imprecise or inaccurate labels. *Incomplete* supervision is when only a small subset of training data is labeled whereas the other data remain unlabeled. Semi-supervised learning is a representative technique for this type of problems where no there is no supervision for the unlabeled samples. *Imprecise* labels provide inexact supervision such as coarse-grained labels and expected label distributions. For example, Kandemir et al. [23] developed a framework that allows labels to be provided for groups of observations instead of instance-level labels. Lei et al. [24] trained a crowd counting model using large amounts of total count-level annotations together with small amounts of location-level annotations. The field of Partial Label Learning (PLL) is another form of imprecise supervision. PLL tackles problems when each training sample is equipped with a set of candidate labels instead of a single label [26]. *Inaccurate* labels are low-quality labels expected to have a high level of noise. Examples include using crowdsourcing systems [26] to collects large amounts of non-expert labeled data, using user-defined heuristic rules to automatically label data [27], or using external knowledge bases to map unlabeled data [28].

There are several works in weakly supervised regression that tackle the above-mentioned forms of weak supervision. For example, Cao et al. [29] proposed a weakly supervised regression model with a tailored loss function to train from inexact annotations, where the inexact labels are categorical defined based on ranges of the continuous ground truth. Kang et al. [30] proposed a semi-supervised support vector regression based on self-training, i.e. training several models from the initial labeled samples and using these models to generate proxy labels on unlabeled samples. Chung et al. [31] proposed a semi-supervised multi-output Gaussian Process model and placed a prior on the group membership of unlabeled samples as a form of weak supervision.

Different from existing WSL methods, we consider a form of weak supervision through ordering relationships between pairs of unlabeled samples. This type of weak label has not been well studied to our best knowledge. Also, the proposed WS-TL is among the first works that investigate how to leverage order-based weakly labeled samples in the TL setting.

C. Knowledge-Informed Machine Learning

One strategy to improve models with insufficient training data is to use prior knowledge as another source of information. The field of Knowledge-informed Machine Learning studies how to incorporate prior knowledge into data-driven learning to improve the accuracy, robustness, and interpretability of the models.

Existing methods differ by the source of knowledge (e.g. scientific knowledge, expert opinion), the representation of knowledge (e.g. algebraic equations, simulation results), and where in the learning pipeline the knowledge is integrated (e.g. data, model structure, objective function). For example, Wang et al. [5] constrained the predictions of a Gaussian Process model to be consistent with the simulation results from a mechanistic model. Erion et al. [32] used a graph regularizer to encourage functionally related genes from known biology to have similar feature contributions in the trained model. Elmarakeby et al. [33] used domain knowledge from a large biology database to customize the architecture of a neural network, where nodes and layer connections are designed to follow biological pathways. A comprehensive taxonomy of knowledge-informed ML can be found in [34].

Motivated from the brain-cancer case study, we focus on domain knowledge that informs pairwise relationships of unlabeled samples. This knowledge is used to generate weakly labeled data to enlarge the training sample size.

III. Preliminaries: Support Vector Regression (SVR)

SVM is a well-known classification algorithm, which is formulated to find a hyperplane with the maximum margin to separate two classes. SVM represents the optimal hyperplane with a sparse collection of support vectors, thus gaining solution efficiency and good generalizability. Another appealing property is that SVM can efficiently perform nonlinear classification by implicitly mapping the input features into a high-dimensional feature space using the so-called kernel trick. The extension of SVM for predicting a continuous response variable is known as SVR [35]. SVR generalizes the maximum margin concept of SVM by introducing an ε-insensitive margin around the predictive function, called an ε-tube. SVR is formulated as an optimization problem to find an ε-tube that includes as many training samples as possible in trade-off with model complexity.

Specifically, let $\mathcal{X} = \mathbb{R}^d$ be the d-dimensional feature space and $\mathcal{Y} = \mathbb{R}$ be the space of the response variable. The predictive function of a sample $\boldsymbol{x} \in X$ is $f(\boldsymbol{x}) = \boldsymbol{w}^T\phi(\boldsymbol{x}) + b$, where $\phi(\boldsymbol{x})$ includes a transformation function of the original feature vector $\boldsymbol{x}$,$\boldsymbol{w}$ contains the model coefficients, and b is the intercept. Denote a training dataset by $\{(\boldsymbol{x}_i, y_i), i = 1, \ldots, n\} \subset \mathcal{X} \times \mathcal{Y}$. The optimization problem of SVR is formulated as:

$$\min_{\boldsymbol{w},b} \frac{1}{2}||\boldsymbol{w}||_2^2 + \frac{c}{n}\sum\nolimits_{i=1}^{n} \max\left(0, \left|y_i - \left(\boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b\right)\right| - \varepsilon\right), \tag{1}$$

where $||\cdot||_2^2$ is the squared l_2 norm and $\max(0, \left|y_i - \left(\boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b\right)\right| - \varepsilon)$ is known as the ε-insensitive loss. ε is a desired accuracy specified a priori (a.k.a. width of the tube). This loss function penalizes the model if the predicted response for a sample, $f(\boldsymbol{x}_i) = \boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b$, is beyond ε-deviation from the true response y_i. C is a hyperparameter that determines the trade-off between minimizing the training loss and model complexity.

Directly solving the optimization in (1) is possible but requires the transformation function $\phi(\boldsymbol{x})$ to be pre-specified. This limits the form of the predictive function and is also inefficient. To solve (1) with good generalizability and efficiency, (1) can be converted to a constrained optimization and solved in its dual form. In this way, the kernel trick can be used to derive the predictive function based on inner product of samples without having to define the form of $\phi(\boldsymbol{x})$ explicitly [35].

IV. Weakly Supervised Transfer Learning (WS-TL)

A. Formulation and Algorithm

Consider a source domain and a target domain that share some similarity (e.g., patients with the same disease) but have non-identical models, f_s and f_t. Suppose we are given a training dataset from the source domain, which consists of n_s labeled samples, $\mathcal{S} = \left\{\left(\boldsymbol{x}_i^s, y_i^s\right), i = 1 \ldots n_s\right\}$. From the target domain, we have a dataset that contains unlabeled samples and there is domain knowledge to create a set of ordered pairs from these unlabeled samples, $\mathcal{T} = \left\{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) : \boldsymbol{x}_l^t \prec \boldsymbol{x}_h^t, (l,h) \in \Omega_t\right\}$, where $\prec$ means that the response variables of the two samples are known to have an order relationship of $y_l^t < y_h^t$ while the exact values of y_l^t and y_h^t are unknown.

The goal of WS-TL is to learn a model for the target domain, f_t, based on the datasets $\mathcal{S}$ and $\mathcal{T}$. We propose the following optimization form:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C_1 L_{\mathcal{S}} + C_2 L_{\mathcal{T}}, \tag{2}$$

where $L_{\mathcal{S}}$ and $L_{\mathcal{T}}$ are loss functions defined on the training datasets of the source and target domains, $\mathcal{S}$ and $\mathcal{T}$, respectively. Since $\mathcal{S}$ contains labeled samples, we can use the ε-insensitive loss of SVR and define

$$L_{\mathcal{S}} = \frac{1}{n_s}\sum\nolimits_{i=1}^{n_s} \max\left(0, \left|y_i^s - \left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right) + b\right)\right| - \varepsilon\right). \tag{3}$$

If the optimization in (2) only included the first two terms, it would become the SVR model for the source domain. However, our interest is to learn a model for the target domain. Thus, the optimization includes a third term with L_T, a specially designed loss function for the ordered samples in the target domain. This is to bias the source model towards the target, and thus achieving transfer learning. Specifically, we propose the following form for L_T:

$$\begin{aligned} L_{\mathcal{T}} &= \frac{1}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t} \max\left(0, \widehat{y}_l^t - \widehat{y}_h^t\right) \\ &= \frac{1}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t} \max\left(0, \left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right) + b\right)\right. \end{aligned}$$

$$-\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right)+b\right)\Big)$$
$$=\frac{1}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t}\max\left(0,\boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right)-\boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right)\right), \quad (4)$$

which penalizes the predicted responses of each pair of ordered samples, $\widehat{y}_l^t$ and $\widehat{y}_h^t$, if the predicted responses violate the order constraint on the true responses, $y_l^t < y_h^t$.

Inserting (3) and (4) into (2), the final form of the WS-TL optimization is:

$$\min_{\boldsymbol{w},b}\frac{1}{2}||\boldsymbol{w}||_2^2+\frac{c_1}{n_s}\sum\nolimits_{i=1}^{n_s}\max\left(0,\left|y_i^s-\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)+b\right)\right|-\varepsilon\right)+\frac{c_2}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t}\max\left(0,\boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right)-\boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right)\right). \quad (5)$$

To solve this optimization without having to pre-specify the form of ϕ, we borrow the idea from SVR by first converting (5) to a constrained optimization using slack variables, ξ_i, ξ_i', ξ_{lh},

$$\begin{aligned}\min_{\boldsymbol{w},b,\xi_i,\xi_i',\xi_{lh}}\ &\frac{1}{2}||\boldsymbol{w}||_2^2+\frac{c_1}{n_s}\sum\nolimits_{i=1}^{n_s}\left(\xi_i+\xi_i'\right)+\frac{c_2}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t}\xi_{lh}\\ \text{s.t.}\ \ &y_i^s-\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)+b\right)\le\xi_i+\varepsilon\\ &\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)+b\right)-y_i^s\le\xi_i'+\varepsilon\\ &\boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right)-\boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right)\le\xi_{lh}\\ &\xi_i,\xi_i',\xi_{lh}\ge 0,\quad i=1\ldots n_s,(l,h)\in\Omega_t. \end{aligned} \quad (6)$$

Furthermore, we can derive the dual form of the primal problem in (6) by first writing the Lagrangian of (6) using nonnegative Lagrange multipliers $\lambda_i,\lambda_i',\lambda_{lh},\alpha_i,\alpha_i',\beta_{lh}\ge 0$,

$$\begin{aligned}\mathcal{L}=&\frac{1}{2}||\boldsymbol{w}||_2^2+\frac{C_1}{n_s}\sum\nolimits_{i=1}^{n_s}\left(\xi_i+\xi_i'\right)+\frac{C_2}{|\Omega_t|}\sum\nolimits_{(l,h)\in\Omega_t}\xi_{lh}\\ &-\sum\nolimits_{i=1}^{n_s}\left(\lambda_i\xi_i+\lambda_i'\xi_i'\right)-\sum\nolimits_{(l,h)\in\Omega_t}\lambda_{lh}\xi_{lh}\\ &+\sum\nolimits_{i=1}^{n_s}\alpha_i\left(y_i^s-\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)-b-\varepsilon-\xi_i\right)\\ &+\sum\nolimits_{i=1}^{n_s}\alpha_i'\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)+b-y_i^s-\varepsilon-\xi_i'\right)\\ &+\sum\nolimits_{(l,h)\in\Omega_t}\beta_{lh}(\boldsymbol{w}^T\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right)-\xi_{lh}).\end{aligned}$$

It follows from the Karush-Kuhn-Tucker (KKT) conditions that the partial derivates of $\mathcal{L}$ with respect to the primal variables $(\boldsymbol{w},b,\xi_i,\xi'_i,\xi_{lh})$ have to vanish at the saddle point of $\mathcal{L}$. Following the KKT conditions and skipping intermediate steps, the dual form of the primal problem can be written as:

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\alpha}',\boldsymbol{\beta}}\ &\frac{1}{2}\sum\nolimits_{i,j}\left(\alpha_i-\alpha_i'\right)\left(\alpha_j-\alpha_j'\right)\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_j^s\right)-\sum\nolimits_i\left(\alpha_i-\alpha_i'\right)y_i^s\\ &+\sum\nolimits_i\left(\alpha_i+\alpha_i'\right)\varepsilon-\sum\nolimits_{i,(l,h)}\left(\alpha_i-\alpha_i'\right)\beta_{lh}\left[\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_l^t\right)\right.\\ &\left.-\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_h^t\right)\right]\quad+\frac{1}{2}\sum\nolimits_{(l,h),(l',h')}\beta_{lh}\beta_{l'h'}\left[\kappa\left(\boldsymbol{x}_l^t,\boldsymbol{x}_{l'}^t\right)\right.\\ &\left.-2\kappa\left(\boldsymbol{x}_l^t,\boldsymbol{x}_{h'}^t\right)+\kappa\left(\boldsymbol{x}_h^t,\boldsymbol{x}_{h'}^t\right)\right]\\ \text{s.t.}\ \ &\alpha_i,\alpha_i'\in\left[0,\frac{c_1}{n_s}\right],i=1\ldots n_s;\sum\nolimits_{i=1}^{n_s}\left(\alpha_i-\alpha_i'\right)=0\\ &\beta_{lh}\in\left[0,\frac{c_2}{|\Omega_t|}\right],(l,h)\in\Omega_t,\end{aligned} \quad (7)$$

where $\kappa(\boldsymbol{x}_1,\boldsymbol{x}_2)\triangleq\phi\left(\boldsymbol{x}_1\right)^T\phi\left(\boldsymbol{x}_2\right)$ denote the kernel function of two samples which can be directly computed on the input space without going through the transformation function ϕ. Also, $\boldsymbol{\alpha}\triangleq\{\alpha_i;i=1\ldots n_s\}$, $\boldsymbol{\alpha}'\triangleq\left\{\alpha_i';i=1\ldots n_s\right\}$, $\boldsymbol{\beta}\triangleq\{\beta_{lh};(l,h)\in\Omega_t\}$.

The optimization in (7) is not convex with respect to all the unknown parameters, $\boldsymbol{\alpha},\boldsymbol{\alpha}',\boldsymbol{\beta}$, simultaneously. But it is biconvex, i.e., it is convex with respect to $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}'$ while fixing $\boldsymbol{\beta}$, and is convex with respect to $\boldsymbol{\beta}$ while fixing $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}'$. Based on this property, we can use Alternating Optimization (AO) to iteratively solve two sub-optimization problems in (i) and (ii):

(i) Given $\boldsymbol{\alpha}^*$, $\boldsymbol{\alpha}'^*$, the optimization in (7) becomes:

$$\begin{aligned}\min_{\boldsymbol{\beta}}\ &\frac{1}{2}\sum_{(l,h),(l',h')}\beta_{lh}\beta_{l'h'}\left[\kappa\left(\boldsymbol{x}_l^t,\boldsymbol{x}_l^t\right)-2\kappa\left(\boldsymbol{x}_l^t,\boldsymbol{x}_{h'}^t\right)\right.\\ &\left.+\kappa\left(\boldsymbol{x}_h^t,\boldsymbol{x}_{h'}^t\right)\right]\\ &-\sum_{i,(l,h)}\left(\alpha_i^*-\alpha_i'^*\right)\beta_{lh}\left[\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_l^t\right)-\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_h^t\right)\right]\\ \text{s.t.}\ \ &\beta_{lh}\in\left[0,\frac{c_2}{|\Omega_t|}\right],(l,h)\in\Omega_t.\end{aligned}$$

(ii) Given $\boldsymbol{\beta}^*$, the optimization in (7) becomes:

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{\alpha}'}\ &\frac{1}{2}\sum_{i,j}\left(\alpha_i-\alpha_i'\right)\left(\alpha_j-\alpha_j'\right)\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_j^s\right)-\sum_i\left(\alpha_i-\alpha_i'\right)y_i^s\\ &+\sum_i\left(\alpha_i+\alpha_i'\right)\varepsilon\\ &-\sum_{i,(l,h)}\left(\alpha_i-\alpha_i'\right)\beta_{lh}^*\left[\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_l^t\right)-\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}_h^t\right)\right]\\ \text{s.t.}\ \ &\alpha_i,\alpha_i'\in\left[0,\frac{c_1}{n_s}\right],i=1\ldots n_s;\sum_{i=1}^{n_s}\left(\alpha_i-\alpha_i'\right)=0.\end{aligned}$$

Each sub-optimization in (i) and (ii) can be solved by a quadratic programming solver. The iterations between (i) and (ii) are guaranteed to converge to a local optimum. To find the global optimum (or a solution that is close enough), we follow the common practice by trying different initial values and selecting the best solution among the different trials.

Furthermore, denoting the final solution of the dual optimization in (7) by $\widehat{\boldsymbol{\alpha}},\widehat{\boldsymbol{\alpha}}',\widehat{\boldsymbol{\beta}}$, the solution of the primal problem in (5) can be derived as:

$$\widehat{\boldsymbol{w}}=\sum\nolimits_{i=1}^{n_s}\left(\widehat{\alpha}_i-\widehat{\alpha}_i'\right)\phi\left(\boldsymbol{x}_i^s\right)+\sum\nolimits_{(l,h)\in\Omega_t}\widehat{\beta}_{lh}\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right).$$

The intercept $\widehat{b}$ can be estimated using one sample that satisfies the KKT equality constraint, i.e., for any i such that $0<\widehat{\boldsymbol{\alpha}}_i<C_1/n_s$, $\widehat{b}=y_i^s-\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right)-\varepsilon$; or for any j such that $0<\widehat{\boldsymbol{\alpha}}_j'<C_1/n_s$, $\widehat{b}=\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_j^s\right)-y_j^s-\varepsilon$.

Finally, we can predict for new sample in the target domain, $\boldsymbol{x}^t$, using the following predictive function:

$$\widehat{y}^t\triangleq\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}^t\right)+\widehat{b}=\sum\nolimits_{i=1}^{n_s}\left(\widehat{\alpha}_i-\widehat{\alpha}_i'\right)\kappa\left(\boldsymbol{x}_i^s,\boldsymbol{x}^t\right)-\sum\nolimits_{(l,h)\in\Omega_t}\widehat{\beta}_{lh}\left[\kappa\left(\boldsymbol{x}^t,\boldsymbol{x}_l^t\right)-\kappa\left(\boldsymbol{x}^t,\boldsymbol{x}_h^t\right)\right]+\widehat{b}, \quad (8)$$

The predictive function (8) is a combination of the weighted inner product of labeled data from the source and the weighted inner product of unlabeled but ordered pairs from the target domain. Without the latter inner product, the function would become using only the model trained for the source domain to

predict the sample in the target, which overlooks the domain difference. The effect of adding the latter inner product is to use ordered training samples from the target domain to "adjust" the source model to become a suitable model for the target, and thus achieving transfer learning.

A final remark about WS-TL is that, even though it was developed by assuming no labeled training sample from the target is available but only ordered pairs, it can be easily extended to include labeled samples from the target, if available, in a similar way as the labeled samples in the source.

B. Properties of WS-TL

An important and advantageous property of SVM/SVR is its solution sparseness, which means that not all the training samples need to be used in the predictive function for a new sample, but only a small subset called Support Vectors (SVs) while non-SVs can be discarded. This greatly improves computational efficiency and saves memory storage of training data. It turns out that WS-TL enjoys a similar property. We provide the definition of SVs for WS-TL as follows.

Theorem 1 (SVs for WS-TL in dual view): Let $\widehat{\boldsymbol{\alpha}}_i$, $\widehat{\alpha}'_i$, $\widehat{\beta}_{lh}, i = 1 \ldots n_s$, $(l,h) \in \Omega_t$ denote the solution of the dual optimization in (7). Any training sample i in the source domain satisfying $\widehat{\boldsymbol{\alpha}}_i > 0$ or $\widehat{\boldsymbol{\alpha}}'_i > 0$ is an SV. Any training sample pair (l,h) in the target domain satisfying $\widehat{\beta}_{lh} > 0$ is an SV. Other training samples and pairs in the source and target domains are non-SVs.

Proof: It is known from the constraints of the optimization in (7) that $\widehat{\boldsymbol{\alpha}}_i \geq 0$, $\widehat{\alpha}'_i \geq 0$. Also, $\widehat{\boldsymbol{\alpha}}_i$ and $\widehat{\boldsymbol{\alpha}}'_i$ cannot be both positive for the same sample. Thus, for each sample i, the possible combinations of $\left(\widehat{\alpha}_i, \widehat{\alpha}'_i\right)$ are $\left(\widehat{\alpha}_i > 0, \widehat{\boldsymbol{\alpha}}'_i = 0\right)$, $\left(\widehat{\boldsymbol{\alpha}}_i = 0, \widehat{\alpha}'_i > 0\right)$, or $\left(\widehat{\boldsymbol{\alpha}}_i = 0, \widehat{\alpha}'_i = 0\right)$. In the last case, $\left(\widehat{\alpha}_i - \widehat{\alpha}'_i\right) = 0$, and thus $\left(\widehat{\boldsymbol{\alpha}}_i - \widehat{\alpha}'_i\right)\kappa\left(\boldsymbol{x}_i^s, \boldsymbol{x}^t\right) = 0$ in the first summation in (8), which excludes sample i from computing the predictive function in (8), i.e., sample i is not a SV. This implies that sample i would be an SV under the first two cases, i.e., $\widehat{\boldsymbol{\alpha}}_i > 0$ or $\widehat{\boldsymbol{\alpha}}'_i > 0$. Following a similar idea, a pair (l,h) with $\widehat{\beta}_{lh} = 0$ would be excluded from computing the predictive function, which means that a pair with $\widehat{\beta}_{lh} = 0$ would be a non-SV. Since the constraints of the optimization in (7) require $\widehat{\beta}_{lh} \geq 0$, an SV would be one with $\widehat{\beta}_{lh} > 0$.

Note that Theorem 1 defines SVs from the viewpoint of the dual optimization solution. Theorem 2 characterizes these SVs from the primal view. (Proof in Appendix A.)

Theorem 2 (SVs for WS-TL in primal view): Let $V = V^s \cup V^t = \left\{i; i = 1 \ldots n_s, \widehat{\boldsymbol{\alpha}}_i > 0 \text{ or } \widehat{\boldsymbol{\alpha}}'_i > 0\right\} \cup \left\{(l,h); (l,h) \in \Omega_t, \widehat{\beta}_{lh} > 0\right\}$ denote the index set of the SVs defined in Theorem 1. Any SV with $i \in V^s$ satisfies $\left|y_i^s - \left(\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right) + \widehat{b}\right)\right| \geq \varepsilon$. Any SV with $(l,h) \in V^t$ satisfies $\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_l^t\right) \geq \widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_h^t\right)$.

From the primal view, the SVs in the source domain are training samples outside or on the boundary of the ε-tube/margin around the predictive function, i.e., the predicted response of a training sample that is an SV is beyond or equal to ε-deviation from the true response. The SVs in the target domain are training sample pairs whose order relationships are violated or minimally satisfied (i.e., having the same predicted response variable). Only these SVs will be used in the prediction of a new sample, while other non-SVs in the training data can be discarded. This gains computational and memory efficiency. Note that both labeled samples and weakly labeled samples inherit this property from the SVM-family.

In addition, WS-TL is robust to outliers. This is because the coefficients for combining the SVs to generate a prediction are upper-bounded, i.e., $\widehat{\boldsymbol{\alpha}}_i, \widehat{\alpha}'_i \leq C_1/n_s$, $\widehat{\beta}_{lh} \leq C_2/|\Omega_t|$. In case that the SVs include some outliers, the influence from these outliers is at most C_1/n_s and $C_2/|\Omega_t|$, so that the model would not be overly biased.

C. Integration of WS-TL With Active Sampling

It is relatively easy to obtain ordered paired samples in the target domain because the exact values of the response variables for these samples are not needed. As a result, it is common for the training dataset to contain many paired samples. Instead of including all the available pairs to train WS-TL, it is desirable to select a subset of pairs that can achieve a similar level of accuracy at a lower computational expense. We propose a pair selection strategy called Active Sampling based on Maximal Model Change (AS-MMC) as follows.

Let $k = 0, 1, \ldots, K$ denote the iterations of adding pairs to the training set. Let $\mathcal{T}^{(k)}_{AS-MMC}$ denote the set of pairs at the k-th iteration. At the initial $k = 0$, no pair is selected and only labeled samples from the source domain are included. Thus, $\mathcal{T}^{(0)}_{AS-MMC} = \emptyset$. Also, $\mathcal{T}^{(0)}_{AS-MMC} \subset \mathcal{T}^{(1)}_{AS-MMC} \subset \cdots \subset \mathcal{T}^{(K)}_{AS-MMC}$ because pairs are incrementally added.

Further, let $\mathrm{O}\left(\boldsymbol{w}; \mathcal{S}, \mathcal{T}^{(k)}_{AS-MMC}\right)$ denote the objective function of the WS-TL optimization at the k-th iteration, i.e., when the training set includes the labeled samples from the source, $\mathcal{S}$, and selected pairs from the target up to this iteration, $\mathcal{T}^{(k)}_{AS-MMC}$. Let $\boldsymbol{w}^{(k)}$ be the optimization solution, i.e.,

$$\boldsymbol{w}^{(k)} = \underset{\boldsymbol{w}}{\operatorname{argmin}}\, \mathrm{O}\left(\boldsymbol{w}; \mathcal{S}, \mathcal{T}^{(k)}_{AS-MMC}\right).$$

The intercept b is easy to solve so it is not shown for notation simplicity. Suppose a candidate pair $\left\{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) : \boldsymbol{x}_l^t \prec \boldsymbol{x}_h^t\right\}$ is added to the training set. Then, in the next iteration, the objective function will be updated to include the loss on this pair, i.e., $L_T\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) = \max\left(0, \boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right) - \boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right)\right)$. As a result, the optimal solution for the model parameter $\boldsymbol{w}$ will change. The goal of AS-MMC is to select the pair that maximally changes the model. However, it is inefficient to compute the model change by repeatedly solving the WS-TL optimization with each candidate pair. To achieve computational efficiency, we propose to approximate the model change by the gradient of loss at the candidate pair [36], i.e.,

$$\Delta\boldsymbol{w}^{(k+1)} \triangleq \boldsymbol{w}^{(k+1)} - \boldsymbol{w}^{(k)} \approx \frac{\partial L_{\mathcal{T}}\left(x_l^\tau, x_h^t\right)}{\partial \boldsymbol{w}}$$

$$\Delta\boldsymbol{w}^{(k+1)} \approx \begin{cases} \phi\left(\boldsymbol{x}_l^t\right) - \phi\left(\boldsymbol{x}_h^t\right), & \boldsymbol{w}^{(k)^T}\phi\left(\boldsymbol{x}_l^t\right) > \boldsymbol{w}^{(k)^T}\phi\left(\boldsymbol{x}_h^t\right). \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

Then, AS-MMC selects the pair that maximally changes the model, which is:

$$\begin{aligned}\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)^{(k+1)} &= \underset{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T}_c^{(k+1)}}{\operatorname{argmax}} \left\| \Delta \boldsymbol{w}^{(k+1)} \right\|_2 \\ &\approx \underset{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T}_c^{(k+1)}}{\operatorname{argmax}} \left\| \phi\left(\boldsymbol{x}_l^t\right) - \phi\left(\boldsymbol{x}_h^t\right) \right\|_2, \end{aligned} \tag{10}$$

where $\mathcal{T}_c^{(k+1)}$ is the set of candidate pairs satisfying the conditions that (i) these pairs have the ability to change the model (i.e., the order relationships of these pairs are violated under the current model) according to (9); and (ii) they have not been selected in previous iterations. The candidate set can be re-written as $\mathcal{T}_c^{(k+1)} = \left\{ \left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) : \widehat{y}_l^{t^{(k)}} > \widehat{y}_h^{t^{(k)}}, \left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T} \backslash \mathcal{T}_{AS-MMC}^{(k)} \right\}$, where $\widehat{y}_l^{t^{(k)}}$ and $\widehat{y}_h^{t^{(k)}}$ are can be obtained using kernel computations according to (8).

To avoid explicitly pre-defining the transformation ϕ, we employ the kernel trick and write (10) as:

$$\begin{aligned}\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)^{(k+1)} &= \underset{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T}_c^{(k+1)}}{\operatorname{argmax}} \left(\phi\left(\boldsymbol{x}_l^t\right) - \phi\left(\boldsymbol{x}_h^t\right)\right)^{\mathcal{T}} \left(\phi\left(\boldsymbol{x}_l^t\right) - \phi\left(\boldsymbol{x}_h^t\right)\right) \\ &= \underset{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T}_c^{(k+1)}}{\operatorname{argmax}} \kappa\left(\boldsymbol{x}_l^t, \boldsymbol{x}_l^t\right) + \kappa\left(\boldsymbol{x}_h^t, \boldsymbol{x}_h^t\right) - 2\kappa\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \\ &\approx \underset{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in \mathcal{T}_c^{(k+1)}}{\operatorname{argmin}} \kappa\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right). \end{aligned} \tag{11}$$

The last step holds for most commonly used kernels as the kernel function between a sample and itself is a constant. (11) implies that the pair selected by AS-MMC is one with two samples farthest apart in the kernel space (i.e., having the smallest kernel function) within the candidate set $\mathcal{T}_c^{(k+1)}$.

Next, we provide a theoretical backup of the AS-MMC strategy by analyzing its convergence property.

Theorem 3 (Convergence of AS-MMC): Recall that $T = \left\{ \left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) : \boldsymbol{x}_l^t \prec \boldsymbol{x}_h^t, (l, h) \in \Omega_t \right\}$ is the set containing all available pairs in the target domain. Suppose $\left|\left|\phi\left(\boldsymbol{x}_l^t\right) - \phi\left(\boldsymbol{x}_h^t\right)\right|\right|_2 \leq R$ for all $\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) \in T$. Further suppose that there exists an optimal solution $\boldsymbol{w}^*$ such that the ordering relationships of all pairs in T are satisfied by at least $\delta > 0$, i.e., $(\boldsymbol{w}^*)^T \phi\left(\boldsymbol{x}_h^t\right) - (\boldsymbol{w}^*)^T \phi\left(\boldsymbol{x}_l^t\right) \geq \delta$. Let N_{AS-MMC} denote the total number of pairs needed by AS-MMC to reach $\boldsymbol{w}^*$ from $\boldsymbol{w}^{(0)}$, where $\boldsymbol{w}^{(0)}$ is the model parameter using only the labeled samples from the source domain. Let $||\boldsymbol{w}^*||_2 = L$ and $\left|\left|\boldsymbol{w}^{(0)}\right|\right|_2 = M$. Then, the upper bound for N_{AS-MMC} is

$$N_{AS-MMC} \leq O\left(\frac{L}{\delta}\left(\frac{LR^2}{\delta} + M\right)\right). \tag{12}$$

Please see proof in Appendix B. Theorem 3 guarantees that AS-MMC converges within a finite number of pairs defined above. Furthermore, we derive the convergence property of random sampling to compare with AS-MMC.

Theorem 4 (Convergence of random sampling): In the same setting of Theorem 3, suppose the probability of sampling a pair $\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)$ satisfying the inequality $\boldsymbol{w}^T \phi\left(\boldsymbol{x}_l^t\right) > \boldsymbol{w}^T \phi\left(\boldsymbol{x}_h^t\right)$ is P. Then the number of pairs N_{RS} needed to reach $\boldsymbol{w}^*$ from $\boldsymbol{w}^{(0)}$ using random sampling is

$$N_{RS} \leq O\left(\frac{L}{\delta P}\left(\frac{LR^2}{\delta} + M\right)\right). \tag{13}$$

Algorithm 1 WS-TL Integrated With AS-MMC

Input: Labeled samples in the source domain, $\mathcal{S} = \left\{\left(\boldsymbol{x}_i^s, y_i^s\right)\right\}_{i=1}^{n_s}$; available ordered paired samples in the target domain, $\mathcal{T} = \left\{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right) : \boldsymbol{x}_l^t \prec \boldsymbol{x}_h^t, (l, h) \in \Omega_t\right\}$; batch size B; stopping criterion ϵ.

Output: Solution to the WS-TL optimization, $\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\alpha}}', \widehat{\boldsymbol{\beta}}$

1. **Initialize:** $k \leftarrow 0$; $\boldsymbol{\alpha}^{(0)} \leftarrow \mathbf{0}, \boldsymbol{a}'^{(0)} \leftarrow \mathbf{0}, \boldsymbol{\beta}^{(0)} \leftarrow \mathbf{0}$; $\mathcal{T}_{AS-MMC}^{(0)} \leftarrow \emptyset$;
2. **Repeat**
3. Train WS-TL using $\mathcal{S}$ and $\mathcal{T}_{AS-MMC}^{(k)}$:
   3.1 Use the proposed AO algorithm to solve the WS-TL optimization and get $\boldsymbol{\alpha}^*, \boldsymbol{\alpha}'^*, \boldsymbol{\beta}^*$;
   3.2 $\boldsymbol{\alpha}^{(k+1)} \leftarrow \boldsymbol{\alpha}^*, \boldsymbol{a}'^{(k+1)} \leftarrow \boldsymbol{\alpha}'^*, \boldsymbol{\beta}^{(k+1)} \leftarrow \boldsymbol{\beta}^*$;
4. Select B pairs using AS-MMC:
   4.1 $\mathcal{T}_c^{(k+1)} \leftarrow \emptyset$;
   4.2 **For** each $\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)$ in $\mathcal{T}\mathcal{T}_{AS-MMC}^{(k)}$ **do**
   4.3 **If** $\widehat{y}_l^{t^{(k)}} > \widehat{y}_h^{t^{(k)}}$ then
   4.4 $\mathcal{T}_c^{(k+1)} \leftarrow \mathcal{T}_c^{(k+1)} \cup \{(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t)\}$;
   4.5 **End if**
   4.6 **End for**
   4.7 $\left\{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)^{(k+1)}\right\}_{i=1}^{B} \leftarrow$ top B pairs from $\mathcal{T}_c^{(k+1)}$ with minimum $\kappa\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)$;
5. $\mathcal{T}_{AS-MMC}^{(k+1)} \leftarrow \mathcal{T}_{AS-MMC}^{(k)} \cup \left\{\left(\boldsymbol{x}_l^t, \boldsymbol{x}_h^t\right)^{(k+1)}\right\}_{i=1}^{B}$;
6. $k \leftarrow k + 1$;
7. **until** $\left|\left|\boldsymbol{\alpha}^{(k+1)} - \boldsymbol{\alpha}^{(k)}\right|\right|_2 + \left|\left|\boldsymbol{\alpha}'^{(k+1)} - \boldsymbol{\alpha}'^{(k)}\right|\right|_2 + \left|\left|\boldsymbol{\beta}^{(k+1)} - \boldsymbol{\beta}^{(k)}\right|\right|_2 \leq \epsilon$
8. **return** $\boldsymbol{\alpha}^{(k+1)}, \boldsymbol{a}'^{(k+1)}, \boldsymbol{\beta}^{(k+1)}$

The proof is skipped. Because $P \in [0, 1]$, the upper bound of AS-MMC is smaller than random sampling, suggesting that AS-MMC needs fewer pairs to reach the same level of accuracy than random sampling (i.e., a faster convergence).

Finally, we discuss some implementation strategies for AS-MMC: *(i) Batch mode*: even though running AS-MMC to select one pair in each iteration has the best chance to identify a minimally needed subset of pairs, it is computationally intensive. In practice, AS-MMC can be run in a batch mode by including B top-ranked pairs according to the criterion in (11) in each iteration. The batch size, B, can be pre-defined. *(ii) Warm start*: in each iteration when the WS-TL optimization is solved based on AS-MMC selected pairs by far, the optimal solution from the previous iteration can be used to initialize the optimization solver (i.e., the proposed AO algorithm in Sec. IV-A. This warm-start strategy will greatly speed up the convergence rate for the AO algorithm compared with random initial values (i.e., cold start). *(iii) Stopping criteria*: several criteria can be employed in practice, such as a pre-specified number of maximum iterations, a pre-specified number of total pairs selected, or a threshold for insignificant model change. Algorithm 1 lays out the major steps of integrating AS-MMC into WS-TL.

D. Hyper-Parameter Tuning

In this paper, we target the situation when the target domain has only a few or even no labeled samples. Thus, two metrics

can be used to tune the hyper-parameters, C_1, C_2. When labeled samples are available in the target domain, we can select C_1, C_2 to minimize the Mean Absolute Predictive Error (MAPE) based on cross-validation. When there is no labeled sample in the target domain, we can select hyperparameters that minimize the Mean Pair Order Error (MPOE) on a validation set that contains ordered pairs:

$$MPOE = \frac{1}{|\Omega'_t|} \sum_{(l,h)\in\Omega'_t} \mathbb{I}\{\widehat{\boldsymbol{w}}^T \phi(\boldsymbol{x}_h^t) < \widehat{\boldsymbol{w}}^T \phi(\boldsymbol{x}_l^t)\},$$

where Ω'_t is the set of indices of validation pairs and I is the indicator function. MPOE is the proportion of pairs that are ordered wrong by the model and can be used as a proxy of consistency with domain knowledge. When both metrics are applicable, they can cross-reference each other for more robust tuning based on both accuracy and domain knowledge.

V. Simulation Study

In this section, we compared the performance of WS-TL with existing TL methods on simulation data. We evaluated WS-TL under two factors: i) few or no samples in the target domain, and ii) small and large difference between the source and target domain, for linear and nonlinear scenarios. Additionally, we compared WS-TL run with random sampling versus active sampling. All experiments were run on 2.8GHz Intel Core i7 with 16GB RAM under Mac OSX operating system and using Python software.

A. Data Generation Process

(A.1) Linear case: Assume a linear relationship between the features and the response variable in the source and target domains, i.e., $y^s = (\boldsymbol{w}^s)^T \boldsymbol{x}^s + \varepsilon$, $y^t = \left(\boldsymbol{w}^t\right)^T \boldsymbol{x}^t + \varepsilon$. We created the domain difference by making $\boldsymbol{w}_s = \boldsymbol{w}_t + \Delta\boldsymbol{w}$, $\Delta\boldsymbol{w} \sim N(0, \delta)$, where δ is set for multiple values to create varying levels of difference between the domains in our experiments, and $\boldsymbol{w}_t \sim N(5, 1)$. To simulate data for the features, we sampled $\boldsymbol{x}^t, \boldsymbol{x}^s$ from $N(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \in \mathbb{R}^{10}$, with $\boldsymbol{\mu}$ being a vector of 5's, and $\boldsymbol{\Sigma}_{ij} = \rho^{|i-j|}\sigma^2, i, j = 1 \ldots 10, \rho = 0.5, \sigma^2 = 1$ to create feature correlations. To simulate data for the response variable, we used the aforementioned linear equations with $\varepsilon \sim N\left(0, 20^2\right)$. Following this data generation scheme, we created a training set that includes 100 labeled samples from the source domain and 600 ordered pairs from the target domain. Additionally, $n_t = 0, 5, 10, 15$ labeled samples from the target were included to represent the settings of no labeled samples available and various small sample sizes of labeled samples. Finally, we generated a separate test set of 100 labeled samples in the target domain to evaluate model performances. The simulation was conducted 20 times. The average signal to noise ratio (SNR), $\frac{Var(f(\boldsymbol{x}))}{Var(\varepsilon)}$, of target samples was approximately 2.

(A.2) Nonlinear case: To generate nonlinear data, we followed a previously published strategy [37] and adopted a polynomial feature mapping $\phi : \mathbb{R}^5 \rightarrow \mathbb{R}^7$, $\phi(\boldsymbol{x}) = (\boldsymbol{x}_1^2, \boldsymbol{x}_4^2, \boldsymbol{x}_1\boldsymbol{x}_2, \boldsymbol{x}_3\boldsymbol{x}_5, \boldsymbol{x}_2, \boldsymbol{x}_4, 1)$. Let $y^s = (\boldsymbol{w}^s)^T \phi(\boldsymbol{x}^s) + \varepsilon$, $y^t = \left(\boldsymbol{w}^t\right)^T \phi\left(\boldsymbol{x}^t\right) + \varepsilon$, $\varepsilon \sim N\left(0, 60^2\right)$. The rest of the data generation process is similar to (A.1). The average SNR of target samples was approximately 4 across 20 replications.

B. Competing Methods

We compare WS-TL with four existing algorithms: TCA [15], KMM [11], Weighted-SVR [10], and Adapt-SVR [19]. These algorithms are included because they are commonly used TL algorithms. Also, they represent the main categories of TL methods as reviewed in Sec. II-A: instance transfer (KMM, Weighted-SVR), feature transfer (TCA), and parameter transfer (Adapt-SVR).

C. Model Performance With Small Labeled Sample Size in the Target Domain

This experiment compares the different methods with $n_t = 5, 10, 15$ labeled samples in the target domain. The training of weighted-SVR used labeled samples in the source and target domains. To train KMM, the features of all available samples (labeled and unlabeled) from the source and target were used, then a supervised learning model such as an SVR was trained using reweighted source and target samples. Similarly, the feature mapping in TCA was found using all samples, followed by a supervised learning model in the transformed feature space. Adapt-SVR was trained by first pre-training an SVR using source samples, then training the target model using the source model and labeled target samples. To train WS-TL, Algorithm 1 was used. Since the goal is to train the best model for the target, labeled samples from the target were weighted higher than the source. The Radial Basis Function (RBF) kernel was used in all methods for the nonlinear case. The hyper-parameters of all methods were tuned to minimize MAPE using cross-validation. The trained models were applied to the test set to compute MAPE.

(C.1) Results of the linear case: Fig. 1(a) and 1(b) show the average accuracies on test data for different methods. WS-TL has the smallest mean MAPE and MPOE in all settings. KMM and TCA perform the worst because they are designed to transfer-learn the marginal distribution between domains and thus do not have a good mechanism to leverage labeled samples in the target while they exist. Weighted-SVR and Adapt-SVR can leverage labeled samples in the target by directly using them to adjust the model of the source, and thus working better than KMM and TCA. However, they require a relatively larger number of labeled samples in the target to make the "adjustment" work. Therefore, Weighted-SVR and Adapt-SVR do not work as well as WS-TL when the labeled sample size in the target is very small, e.g., $n_t = 5$. Furthermore, when there is a large difference between the source and the target domains (Fig. 1(b)), which is a challenging situation for transfer learning, the benefit of WS-TL is quite substantial compared to the other methods. Note that in the case of small difference between source and target domains, adding a few more labeled samples did not improve the average accuracy of WS-TL. This may be because the labeled source samples were sufficient to learn a reasonably good target model or there exists redundancy of information contained in the few added samples and weakly labeled samples.

(C.2) Results of the nonlinear case: All the observations in the linear case hold for the nonlinear case (Fig. 1(c) and 1(d)). Moreover, WS-TL is better than the other methods in some

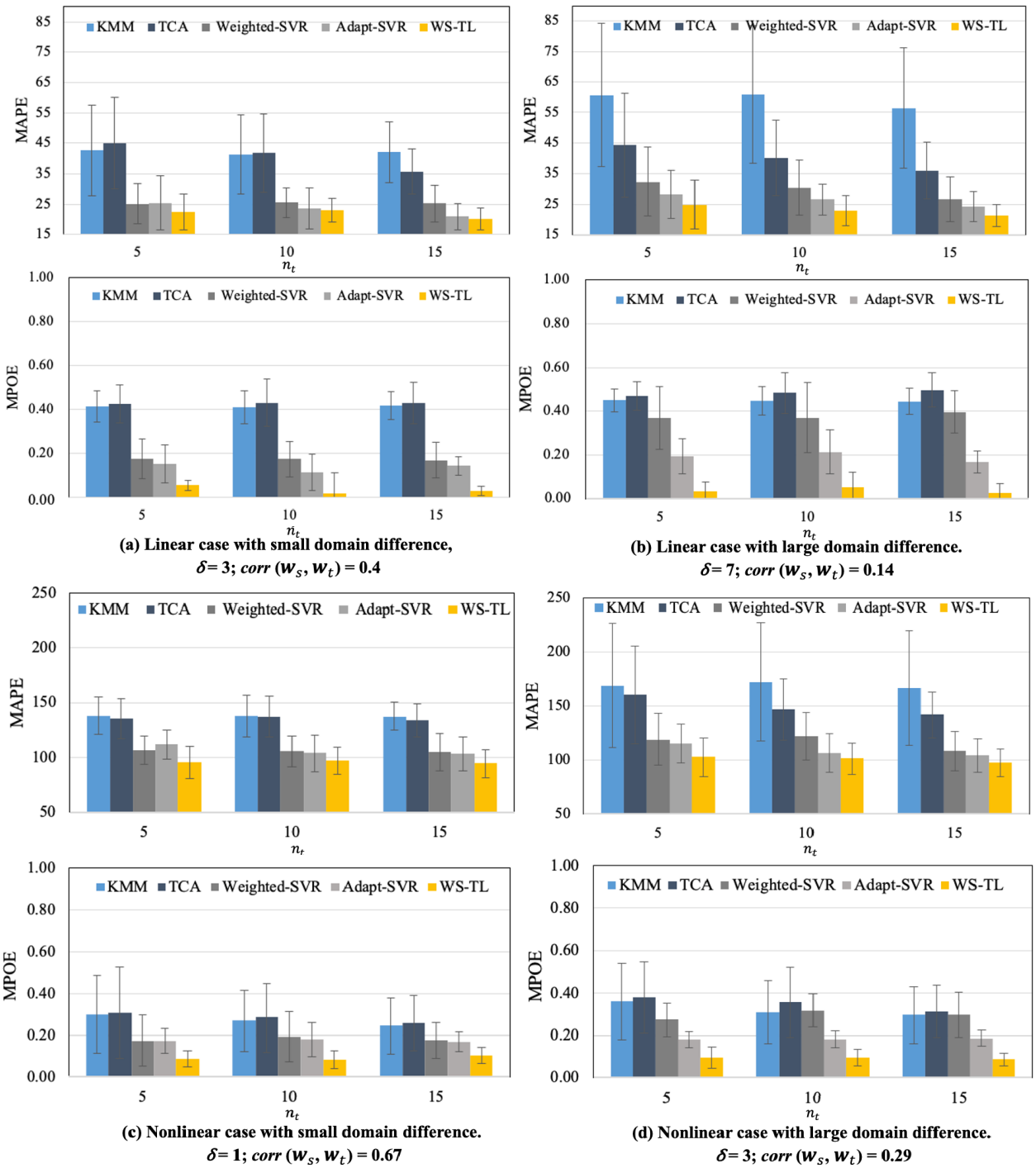

Fig. 1. Test MAPE and MPOE (y-axis) for different methods with varying number of labeled sample sizes in the target domain (x-axis).

additional aspects: When the labeled sample size in the target is very small, e.g., $n_t = 5$, the MAPE gap between the best performer of the competing methods (KMM or TCA) and WS-TL is more substantial than the linear case. Also, this gap is consistently existing regardless of how different the source and target domains are (i.e., in both Fig 1(c) and (d)). In these experiments, to achieve the accuracy of WS-TL trained using five labeled samples from the target domain, Adapt-SVR and Weighted-SVR required approximately 10-20 samples, on average, and KMM and TCA required more than a hundred of samples. These results show that WS-TL can achieve a reasonable level of MAPE requiring less labeled samples compared to other TL methods. Note that WS-TL had much lower MPOE than other models in the linear case compared to the non-linear case. This may be because the ordering relationships depend on the varying local landscapes of the nonlinear function, making test pairs more difficult to infer from the training pairs, whereas directions of increase/decrease are more homogeneous in the linear setting.

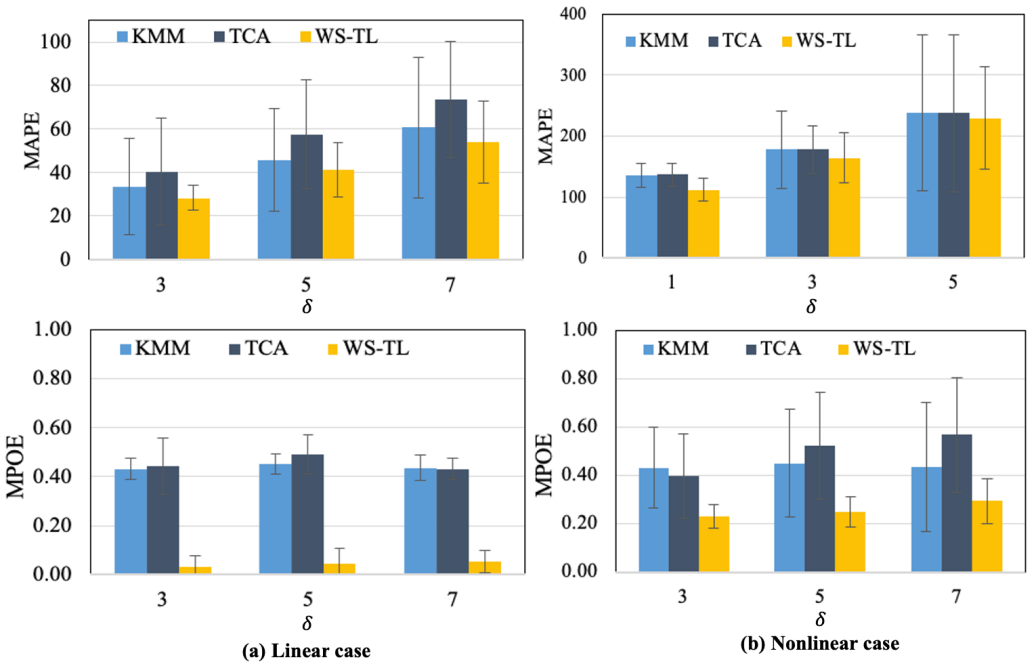

Fig. 2. Test MAPE and MPOE (y-axis) for different methods with varying levels of source-target difference (x-axis) when there is no labeled sample in the target domain.

D. Model Performance With no Labeled Samples in the Target Domain

This experiment compares the different methods when there are no labeled samples in the target domain. Only WS-TL, TCA, and KMM can handle this more challenging situation, and they are trained in the same way as before (just without any labeled sample from the target). In WS-TL, the ordered paired samples from the target are half-half split into a training set and a validation set, with hyper-parameters selected to minimize the validation MPOE. Fig. 2(a) and (b) show the results of the linear and nonlinear cases, respectively. WS-TL had a smaller mean MAPE on test data than the competing methods in all settings.

E. Model Performance With Random Sampling Versus Active Sampling

This experiment compares WS-TL run using pairs selected by random sampling versus pairs selected by active sampling (with warm-start versus cold-start) in both accuracy and computational time. The setting in A.1 with $\delta = 7$ was used conduct this experiment. For each replication, an initial pool of 10,000 pairs was generated to ensure there are enough pairs to be selected by the AS-MMC criteria. The randomly sampling strategy (RS) drew n_p pairs from the pool. The active sampling (AS) strategy iteratively drew batches of 20 pairs without replacement based on the AS-MMC criteria until n_p pairs were obtained (Algorithm 1). Under warm-start, each new batch was used to fine-tune the previously trained model. Under cold-start, the model was completely retrained using all batches selected up to each time point. The experiment was repeated 20 times for $n_p = \{20, 40, \ldots, 500\}$. We report the training time as the total cumulative time to select pairs and train WS-TL.

Fig. 3 shows that WS-TL run with active sampling required less pairs to achieve the same level of MAPE as random sampling at the cost of higher computational time, which can be significantly decreased using the warm-start strategy. Note that using a smaller batch size is expected to result in more effective sampling, i.e. better accuracy, at the expense of more sampling iterations, i.e. higher computational cost. Active sampling had better consistency (i.e. smaller variance across repetitions) in MAPE and MPOE than random sampling. However, active sampling can result in a slightly worse MPOE and saturation of MPOE when a large number of pairs are used. This may be because active sampling limits the pool of candidate pairs, which introduces bias into the distribution of weakly labeled samples, whereas random sampling allows exploration of diverse pairs. In summary, active sampling is beneficial when a small number of pairs is needed to achieve the desirable level of accuracy.

VI. Case Study: Precision Medicine for Brain Cancer

The background of this application was discussed in the motivating example in Sec. I. The goal of this study is to build a patient-specific model to predict regional TCD across the tumoral area in the brain based on MRI.

A. Data Collection and Preprocessing

Patients and biopsy samples: This study uses the data collected by our collaborators at Mayo Clinic for 34 patients with GBM. IRB has been approved. A total of 155 biopsy samples were collected with an average of four samples per patient. For each biopsy sample, the TCD was measured through histopathologic analysis, which is a continuous variable ranging between 0 and 100% measuring the percent of tumor content within the biopsy sample. The higher the

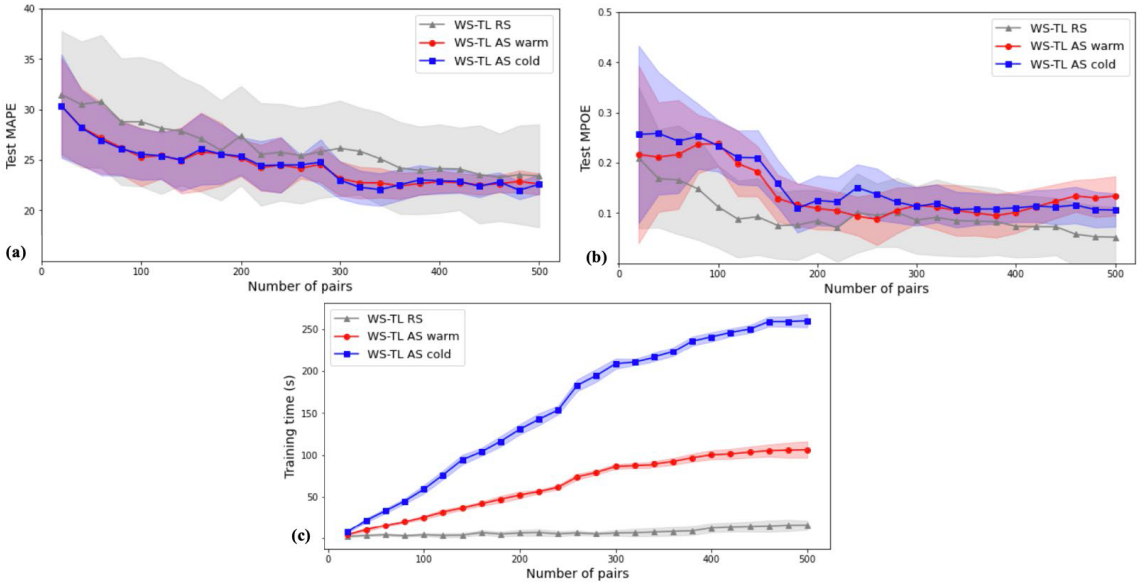

Fig. 3. Average test MAPE (a), test MPOE (b), and training time (c) of WS-TL run with i) random sampling (RS), ii) active sampling with warm start (AS warm), and iii) active sampling with cold start (AS cold). Standard deviations are displayed in shaded areas around the averages.

TCD, the more tumor content in the local brain region from where the biopsy was acquired, which suggests the need for more aggressive treatment to that region. Regional TCD is the response variable in this study.

MRI preprocessing and feature extraction: Each patient underwent a pre-surgical MRI examination, prior to image-localized biopsies. The imaging session generated multiple contrast images such as T1-weighted contrast-enhanced (T1+C), T2-weighted sequences (T2), dynamic contrast enhancement (EPI+C), mean diffusivity (MD), fractional anisotropy (FA), and relative cerebral blood volume (rCBV). Details of the imaging protocols can be found in our prior publications [7], [38]. As our goal is to build a model using MRI to predict regional TCD, regional MRI features were computed using a sliding window approach. That is, an 8×8 pixel^2 window was slid through a pre-segmented tumoral Region of Interest (ROI) within the brain, pixel by pixel. From each window, we computed the average grey level intensity over all the pixels included in that window for each contrast image, which are used as features.

Labeled samples and ordered paired samples: The biopsy samples are the labeled samples. To generate ordered paired samples for each patient, we leveraged the domain knowledge that the TCD near the boundary of the enhancing area of the tumor is likely to be higher than that near the boundary of the non-enhancing area. This phenomenon has been explained and demonstrated in prior papers [39]. Denote the aforementioned boundary areas by $\mathcal{A}_h$ and $\mathcal{A}_l$, respectively. We created each ordered pair by taking one sample from $\mathcal{A}_h$ and another sample from $\mathcal{A}_l$, i.e., $\left\{(\boldsymbol{x}_i, \boldsymbol{x}_j) : \boldsymbol{x}_i \in \mathcal{A}_h, \boldsymbol{x}_j \in \mathcal{A}_l\right\}$, with the domain knowledge that $\boldsymbol{x}_j \prec \boldsymbol{x}_i$. Since $\mathcal{A}_h$ and $\mathcal{A}_l$ each include many pixels, we can create a large number of ordered pairs for each patient.

B. Modeling and Results by Different Methods

We applied WS-TL and the existing methods to this dataset. In applying each method, one model is trained for each patient as the target domain, and this process iterates through all the patients to get patient-specific models. From the target patient/domain, the biopsy/labeled samples and 600 paired samples were included, while the biopsy samples from all other patients were included as labeled samples in the source domain. The process of model training for each method is similar to that in the simulation study in Sec. V-C. The hyperparameters of WS-TL were chosen to minimize cross-validation MAPE of labeled target samples within those that ensure <0.20 MPOE on validation pairs. This tuning strategy requires the model to maintain a reasonable level of consistency with domain knowledge while maximizing accuracy. Because Weighted-SVR and Adapt-SVR require a relatively large number of labeled samples in the target domain, we evaluated them on the 14 patients who has at least five biopsies (100 samples).

For performance comparison between different methods, the MAPE based on leave-one-out cross-validation (LOOCV) was computed. Also, we computed the MPOE on 300 ordered pairs in a validation set from each patient to check the consistency of model prediction with domain knowledge. Tables I and II compare all methods for MAPE and MPOE and shows the p-value of paired t-test that compares the metric means of WS-TL with other methods. WS-TL achieved the best accuracy (smallest MAPE and MPOE) for the scenario with all patients and the scenario including only patients with at least five biopsy samples. We want to point out that all methods performed better than the one-model-fits-all approach of training one SVR model for all patients, which only achieved a MAPE of 0.18.

C. Generation of TCD Prediction Maps

The ultimate goal is to use the trained model to predict the TCD of any unbiopsied region based on image features from the corresponding region, which would allow the generation of a predicted TCD map within the tumoral ROI for each patient to guide informed, individualized treatment. To do this, the

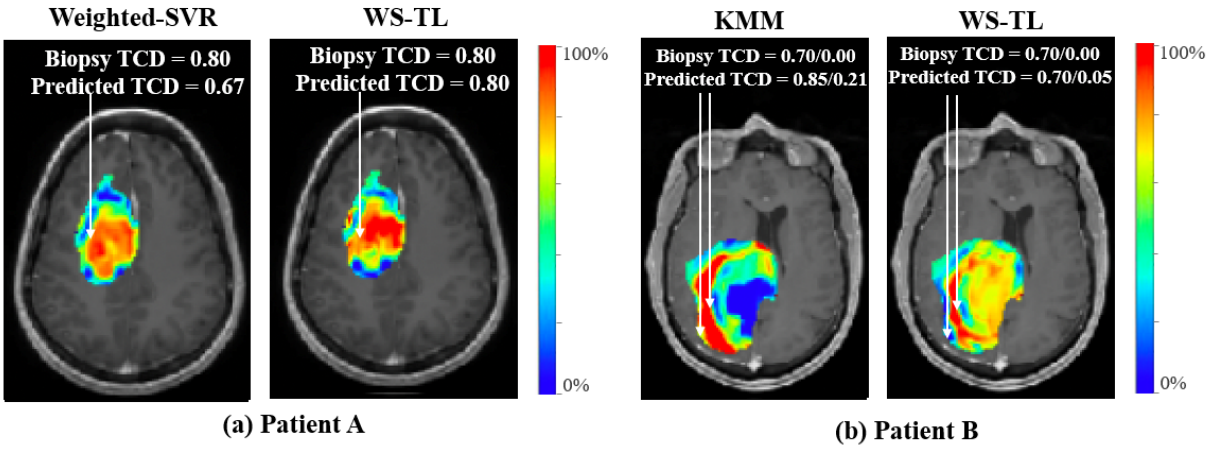

Fig. 4. Predicted TCD maps within the tumoral ROI of two patients by WS-TL and the best-performing competing method (color maps overlaid on the patient's T2 MRI image).

TABLE I
LOOCV MAPE AVERAGED OVER ALL PATIENTS

MAPE		KMM	TCA	Weighted SVR	Adapt SVR	WS-TL (proposed)
All patients	mean	0.113	0.142	-	-	**0.097**
	std	0.104	0.132	-	-	0.088
	p-value	0.010	0.003	-	-	-
Patients with ≥5 biopsy samples	mean	0.136	0.141	0.131	0.139	**0.114**
	std	0.075	0.087	0.062	0.070	0.078
	p-value	0.031	0.056	0.056	0.035	-

TABLE II
VALIDATION MPOE PER PATIENT AVERAGED OVER ALL PATIENTS

MPOE		KMM	TCA	Weighted SVR	Adapt SVR	WS-TL (proposed)
All patients	mean	0.274	0.444	-	-	**0.133**
	std	0.157	0.226	-	-	0.143
	p-value	<0.001	<0.001	-	-	-
Patients with ≥5 biopsy samples	mean	0.305	0.396	0.279	0.258	**0.198**
	std	0.146	0.238	0.153	0.162	0.137
	p-value	<0.001	0.001	0.001	0.005	-

trained model of each method was used to predict regional TCD based on image features from the sliding windows. Then, the predictions of all the sliding windows were visualized by a color map overlaid on the ROI for each patient. Fig. 4 shows the predicted TCD maps by WS-TL for two patients as examples. For comparison, we also show the maps by the best performer among the competing methods for the same patients. Patient A has one biopsy sample shown on this slice of the MRI. The predicted TCD by WS-TL matches the true TCD. Weighted-SVR has the smallest MAPE for this patient among all the competing methods, which still underestimates the TCD. Patient B has two biopsy samples on this slice of the MRI, for which WS-TL predicted more accurately than KMM, the best competing method for this patient. WS-TL can more accurately capture the spatial distribution of TCD because it is trained using not only biopsy samples but also ordered pairs from relatively high and low TCD areas inferred from domain knowledge.

VII. CONCLUSION

We proposed a new TL method, WS-TL, to learn a model for the target domain based on paired samples whose order relationships are inferred from domain knowledge, while at the same time integrating the labeled samples in the source domain for transfer learning. We showed that WS-TL benefits from nice properties such as solution sparseness and robustness. We also proposed a novel strategy to select informative paired samples, AS-MMC, and showed that WS-TL can achieve a faster convergence rate when integrated with AS-MMC than random sampling. The performance of WS-TL was demonstrated in simulation studies under various TL settings. In a real-world case study for predicting the regional TCD distribution for patients with GBM, WS-TL built personalized models for each patient and generated more accurate predictions than a variety of competing methods.

Finally, we provide some discussion regarding the limitations of this work and potential future directions. First, any machine learning model would fail under some conditions. Identifying the failure conditions helps understand the model and use it properly. Theoretically speaking, the failure conditions of WS-TL are that there are too few labeled samples in the target domain (zero in the extreme case) AND the weak labels include too many wrongly ordered pairs. If the first condition is true, the model may still achieve good performance if provided with a good set of paired samples. If the second condition is true, choosing a low C_2 may prevent the model from being misled by the wrong pairs. Thus, the quality of weak labels and the choice of hyperparameter C_2 are crucial to the model performance. It is also worth noting that, while using weakly labeled samples can enlarge training sample size, choosing which domain knowledge or weak labels to include is an additional source of bias. Thus, future research can include development of robust algorithms to wrongful or/and biased domain knowledge. Furthermore, we chose SVR as the base model in this work because of its solid theory and interpretability, but the proposed strategy of using order-based

domain knowledge through weakly labeled samples can be extended to other machine learning models such as neural networks. Future research can explore such extensions. Last but not least, the current formulation of WS-TL combines all patients but one as one source with purpose of identifying a population-average model, which is then customized toward the left-out (i.e., target) patient. There is patient heterogeneity in the source patients. Future research can be done to develop multi-source WS-TL with strategies to select source patients to transfer-learn from and to prevent negative transfer.

APPENDIX A
PROOF OF THEOREM 2

For optimal solutions $\widehat{\boldsymbol{\alpha}},\widehat{\alpha}',\widehat{\beta},\widehat{\boldsymbol{w}},\widehat{b}$, the following KKT conditions must be satisfied:

$$\begin{aligned}
&\widehat{\alpha}_i\left(y_i^s-\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right)-\widehat{b}-\varepsilon-\widehat{\xi}_i\right)=0 \quad \forall i=1\ldots n_s,\\
&\widehat{\alpha}_i'\left(\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right)+\widehat{b}-y_i^s-\varepsilon-\widehat{\xi}_i'\right)=0 \quad \forall i=1\ldots n_s,\\
&\widehat{\beta}_{lh}\left(\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_l^t\right)-\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_h^t\right)-\widehat{\xi}_{lh}\right)=0 \quad \forall(l,h)\in\Omega_t.
\end{aligned}$$

By complementary slackness,

$$\begin{aligned}
&y_i^s-\left(\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right)+\widehat{b}\right)=\varepsilon+\widehat{\xi}_i \quad \text{if } \widehat{\boldsymbol{\alpha}}_i>0,\\
&\left(\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_i^s\right)+\widehat{b}-y_i^s\right)=\varepsilon+\widehat{\xi}_i' \quad \text{if } \widehat{\boldsymbol{\alpha}}_i'>0,\\
&\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_l^t\right)-\widehat{\boldsymbol{w}}^T\phi\left(\boldsymbol{x}_h^t\right)=\widehat{\xi}_{lh} \quad \text{if } \widehat{\beta}_{lh}>0.
\end{aligned}$$

Since $\widehat{\xi}_i,\widehat{\xi}_i',\widehat{\xi}_{lh}\geq 0$,

$$\begin{aligned}
&\left|y_i^s-\left(\boldsymbol{w}^T\phi\left(\boldsymbol{x}_i^s\right)+b\right)\right|\geq\varepsilon \quad \text{if } \widehat{\boldsymbol{\alpha}}_i>0 \text{ or } \widehat{\alpha}_i'>0,\\
&\boldsymbol{w}^T\phi\left(\boldsymbol{x}_l^t\right)\geq\boldsymbol{w}^T\phi\left(\boldsymbol{x}_h^t\right) \quad \text{if } \widehat{\beta}_{lh}>0.
\end{aligned}$$

The result of the theorem follows. ■

APPENDIX B
PROOF OF THEOREM 3

The pair $\left(\boldsymbol{x}_l^t,\boldsymbol{x}_h^t\right)^{(k+1)}$ added to the training set at iteration k by AS-MMC criteria satisfies

$$\boldsymbol{w}^{(k)^T}\phi\left(\boldsymbol{x}_l^t\right)>\boldsymbol{w}^{(k)^T}\phi\left(\boldsymbol{x}_h^t\right)\Rightarrow\boldsymbol{w}^{(k)^T}\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right)>0.$$

Recall that we approximate the model change by the gradient of loss at the candidate pair. From (9),

$$\boldsymbol{w}^{(k+1)}\triangleq\boldsymbol{w}^{(k)}+\Delta\boldsymbol{w}^{(k+1)}\approx\boldsymbol{w}^{(k)}-(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)). \quad (14)$$

Thus, we have

$$\begin{aligned}
\left|\left|\boldsymbol{w}^{(k+1)}\right|\right|_2^2&=\left|\left|\boldsymbol{w}^{(k)}-\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right)\right|\right|_2^2\\
&=\left|\left|\boldsymbol{w}^{(k)}\right|\right|_2^2+R^2-2\boldsymbol{w}^{(k)^T}\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right)\\
&\leq\left|\left|\boldsymbol{w}^{(k)}\right|\right|_2^2+R^2. \quad (15)
\end{aligned}$$

Multiplying both sides of (14) by $\boldsymbol{w}^*$ we have

$$\begin{aligned}
\left(\boldsymbol{w}^{(k+1)}\right)^T\boldsymbol{w}^*&=\left(\boldsymbol{w}^{(k)}\right)^T\boldsymbol{w}^*-\left(\phi\left(\boldsymbol{x}_l^t\right)-\phi\left(\boldsymbol{x}_h^t\right)\right)^T\boldsymbol{w}^*\\
&\geq\left(\boldsymbol{w}^{(k)}\right)^T\boldsymbol{w}^*+\delta. \quad (16)
\end{aligned}$$

Through the deduction of (15) and (16) for N iterations,

$$\left|\left|\boldsymbol{w}^N\right|\right|_2^2\leq\left|\left|\boldsymbol{w}^{(0)}\right|\right|^2+NR^2=M^2+NR^2;$$

and

$$\begin{aligned}
\left(\boldsymbol{w}^N\right)^T\boldsymbol{w}^*&\geq\left(\boldsymbol{w}^{(0)}\right)^T\boldsymbol{w}^*+N\delta=\left|\left|\boldsymbol{w}^{(0)}\right|\right|\left|\left|\boldsymbol{w}^*\right|\right|\cos\theta+N\delta\\
&=LM\cos\theta+N\delta\geq-LM+N\delta,
\end{aligned}$$

where θ is the angle between $\boldsymbol{w}^{(0)}$ and $\boldsymbol{w}^*$.

According to Cauchy-Schwartz inequality,

$$\left(\boldsymbol{w}^N\right)^T\boldsymbol{w}^*\leq\left|\left|\boldsymbol{w}^N\right|\right|\left|\left|\boldsymbol{w}^*\right|\right|.$$

Thus,

$$-LM+N\delta\leq L\sqrt{M^2+NR^2}.$$

Simplifying the above we obtain the upper bound for N:

$$N\leq\frac{L}{\delta}\left(\frac{LR^2}{\delta}+2M\right)=O\left(\frac{L}{\delta}\left(\frac{LR^2}{\delta}+M\right)\right).$$

■

ACKNOWLEDGMENT

The are grateful to all of those who may have contributed to elements of this work, particularly the many surgeons, for collecting the biopsies from the Barrow Neurological Institute; those from Mayo Clinic Arizona, with special thanks to Kris Smith, Peter Nakaji, Bernard Bendok, Devi Patra, and Richard Zimmerman; the Glioma Biopsy Protocol Teams, for aiding in the many logistics ensuring the integrity of the biopsy samples and screenshots and in aiding with clinical data abstraction, including Barrett Anderies, Spencer Bayless, Ashlyn Gonzales, Ryan Hess, Julia Lorence, and Ashley Nespodzany; and finally other past and current members of the Image Analysis Team, for image segmentations, special mention of Cassandra Rickertsen and Lisa Paulson, for their leadership.

REFERENCES

[1] T. De Cooman et al., "Personalizing heart rate-based seizure detection using supervised SVM transfer learning," *Frontiers Neurol.*, vol. 11, pp. 1–13, Feb. 2020, doi: 10.3389/fneur.2020.00145.

[2] B. Q. Huynh, H. Li, and M. L. Giger, "Digital mammographic tumor classification using transfer learning from deep convolutional neural networks," *J. Med. Imag.*, vol. 3, no. 3, Aug. 2016, Art. no. 034501, doi: 10.1117/1.jmi.3.3.034501.

[3] T. de Cooman, C. Varon, A. van de Vel, B. Ceulemans, L. Lagae, and S. van Huffel, "Semi-supervised one-class transfer learning for heart rate based epileptic seizure detection," *Comput. Cardiol.*, vol. 44, pp. 1–4, Jan. 2017, doi: 10.22489/CinC.2017.257-052.

[4] D. Corwin et al., "Toward patient-specific, biologically optimized radiation therapy plans for the treatment of glioblastoma," *PLoS ONE*, vol. 8, no. 11, Nov. 2013, Art. no. e79115, doi: 10.1371/journal.pone.0079115.

[5] L. Wang, A. Hawkins-Daarud, K. R. Swanson, L. S. Hu, and J. Li, "Knowledge-infused global-local data fusion for spatial predictive modeling in precision medicine," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 2203–2215, Jul. 2022, doi: 10.1109/TASE.2021.3076117.

[6] L. S. Hu et al., "Accurate patient-specific machine learning models of glioblastoma invasion using transfer learning," *Amer. J. Neuroradiol.*, vol. 40, no. 3, pp. 418–425, Feb. 2019, doi: 10.3174/ajnr.A5981.

[7] N. Gaw et al., "Integration of machine learning and mechanistic models accurately predicts variation in cell density of glioblastoma using multiparametric MRI," *Sci. Rep.*, vol. 9, no. 1, pp. 1–9, Jul. 2019, doi: 10.1038/s41598-019-46296-4.

[8] H. Cruz, M. Eckert, J. Meneses, and J.-F. Martínez, "Efficient forest fire detection index for application in unmanned aerial systems (UASs)," *Sensors*, vol. 16, no. 6, p. 893, Jun. 2016.

[9] S. Dhondt, N. Wuyts, and D. Inzé, "Cell to whole-plant phenotyping: The best is yet to come," *Trends Plant Sci.*, vol. 18, no. 8, pp. 428–439, Aug. 2013.

[10] G. Schweikert, C. Widmer, B. Schälkopf, and G. Rötsch, "An empirical analysis of domain adaptation algorithms for genomic sequence analysis," in *Proc. NIPS*, 2009, pp. 1433–1440.

[11] J. Huang et al., "Correcting sample selection bias by unlabeled data," in *Proc. NeurIPS*, vol. 19, 2006, pp. 601–608, doi: 10.7551/mitpress/7503.003.0080.

[12] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. 24th Int. Conf. Mach. Learn.*, Jun. 2007, pp. 193–200, doi: 10.1145/1273496.1273521.

[13] J. Jiang and C. X. Zhai, "Instance weighting for domain adaptation in NLP," in *Proc. ACL*, Jun. 2007, pp. 264–271.

[14] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. AAAI*, vol. 2, 2008, pp. 677–682.

[15] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011, doi: 10.1109/TNN.2010.2091281.

[16] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2200–2207, doi: 10.1109/ICCV.2013.274.

[17] M. Long, J. Wang, G. Ding, S. J. Pan, and P. S. Yu, "Adaptation regularization: A general framework for transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1076–1089, May 2014, doi: 10.1109/TKDE.2013.111.

[18] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive SVMs," in *Proc. 15th ACM Int. Conf. Multimedia*, Sep. 2007, pp. 188–197, doi: 10.1145/1291233.1291276.

[19] J. Yang, R. Yan, and A. G. Hauptmann, "Adapting SVM classifiers to data with shifted distributions," in *Proc. 7th IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Oct. 2007, pp. 69–74, doi: 10.1109/ICDMW.2007.37.

[20] L. Duan, D. Xu, and I. W. Tsang, "Domain adaptation from multiple sources: A domain-dependent regularization approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 504–518, Mar. 2012, doi: 10.1109/TNNLS.2011.2178556.

[21] Y. Yang, X. Li, P. Wang, Y. Xia, and Q. Ye, "Multi-source transfer learning via ensemble approach for initial diagnosis of Alzheimer's disease," *IEEE J. Transl. Eng. Health Med.*, vol. 8, pp. 1–10, 2020, doi: 10.1109/JTEHM.2020.2984601.

[22] J. Li, S. Qiu, Y.-Y. Shen, C.-L. Liu, and H. He, "Multisource transfer learning for cross-subject EEG emotion recognition," *IEEE Trans. Cybern.*, vol. 50, no. 7, pp. 3281–3293, Jul. 2020, doi: 10.1109/TCYB.2019.2904052.

[23] M. Kandemir and F. A. Hamprecht, "Computer-aided diagnosis from weak supervision: A benchmarking study," *Computerized Med. Imag. Graph.*, vol. 42, pp. 44–50, Jun. 2015. [Online]. Available: http://hci.iwr.uni-heidelberg.de/Staff/mkandemi/MILBundle.tar.gz

[24] Y. Lei, Y. Liu, P. Zhang, and L. Liu, "Towards using count-level weak supervision for crowd counting," *Pattern Recognit.*, vol. 109, Jan. 2021, Art. no. 107616.

[25] P. Nodet et al., "From weakly supervised learning to biquality learning: An introduction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2021, pp. 1–10.

[26] A. Jain, D. Das, J. K. Gupta, and A. Saxena, "PlanIt: A crowdsourcing approach for learning to plan paths from large scale preference feedback," in *Proc. IEEE ICRA*, Jun. 2015, pp. 877–884, doi: 10.1109/ICRA.2015.7139281.

[27] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Ré, "Snorkel: Rapid training data creation with weak supervision," *Proc. VLDB Endowment*, vol. 11, no. 3, pp. 269–282, Nov. 2017, doi: 10.14778/3157794.3157797.

[28] J. Shang, L. Liu, X. Gu, X. Ren, T. Ren, and J. Han, "Learning named entity tagger using domain-specific dictionary," 2018, *arXiv:1809.03599*.

[29] Y. Cao et al., "Predicting pathogenicity of missense variants with weakly supervised regression," *Hum. Mutation*, vol. 40, no. 9, pp. 1579–1592, 2019.

[30] P. Kang, D. Kim, and S. Cho, "Semi-supervised support vector regression based on self-training with label uncertainty: An application to virtual metrology in semiconductor manufacturing," *Expert Syst. Appl.*, vol. 51, pp. 85–106, Jun. 2016.

[31] S. Chung, R. A. Kontar, and Z. Wu, "Weakly-supervised multi-output regression via correlated Gaussian processes," 2020, *arXiv:2002.08412*.

[32] G. Erion, J. D. Janizek, P. Sturmfels, S. M. Lundberg, and S.-I. Lee, "Improving performance of deep learning models with axiomatic attribution priors and expected gradients," *Nature Mach. Intell.*, vol. 3, no. 7, pp. 620–631, May 2021.

[33] H. A. Elmarakeby et al., "Biologically informed deep neural network for prostate cancer discovery," *Nature*, vol. 598, no. 7880, pp. 348–352, Oct. 2021.

[34] L. von Rueden et al., "Informed machine learning—A taxonomy and survey of integrating prior knowledge into learning systems," 2019, *arXiv:1903.12394*.

[35] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, no. 3, pp. 199–222, Aug. 2004, doi: 10.1023/B:STCO.0000035301.49549.88.

[36] W. Cai, Y. Zhang, S. Zhou, W. Wang, C. Ding, and X. Gu, "Active learning for support vector machines with maximum model change," in *Proc. ECML PKDD*, 2014, pp. 211–226, doi: 10.1007/978-3-662-44848-9_14.

[37] A. Argyriou, T. Evgeniou, and M. Pontil, "Convex multi-task feature learning," *Mach. Learn.*, vol. 73, no. 3, pp. 243–272, Dec. 2008, doi: 10.1007/s10994-007-5040-8.

[38] L. S. Hu et al., "Multi-parametric MRI and texture analysis to visualize spatial histologic heterogeneity and tumor extent in glioblastoma," *PLoS ONE*, vol. 10, no. 11, Nov. 2015, Art. no. e0141506.

[39] N. Sanai and M. S. Berger, "Glioma extent of resection and its impact on patient outcome," *Neurosurgey*, vol. 62, no. 4, pp. 753–766, 2008.

Lingchao Mao (Member, IEEE) received the B.S. degree in statistics and industrial systems engineering from North Carolina State University in 2020. She is currently pursuing the Ph.D. degree in machine learning with the Georgia Institute of Technology. Her research interests include machine learning and statistical modeling.

Lujia Wang (Member, IEEE) received the B.S. degree in mathematics and applied mathematics from Nankai University, Tianjin, China, in 2013, and the M.S. degree in probability and mathematical statistics from the Chinese Academy of Sciences, Beijing, China, in 2016. She is currently pursuing the Ph.D. degree with the School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA. Her research interests include machine learning and biomedical imaging analytics.

Leland S. Hu received the M.D. degree from the University of Texas–Southwestern Medical Center, Dallas, TX, USA. He completed the clinical internship and residency in Diagnostic Radiology with the University of Texas–Southwestern Medical Center. He also completed the two year clinical fellowship in Diagnostic Neuroradiology with the Barrow Neurological Institute, Phoenix, AZ, USA. He received the Board Certification in Diagnostic Radiology and the Subspecialty Certification (CAQ) in Neuroradiology from the American Board of Radiology (ABR). He is currently a Consultant Physician with the Department of Radiology, Mayo Clinic in Arizona, and holds an academic appointment as an Assistant Professor in Radiology with the Mayo Clinic School of Medicine. He oversees the clinical imaging component of the Neuro-Oncology Program at the Mayo Clinic in Arizona. His research interests focus on image-based modeling and clinical imaging applications for the study of brain tumors.

Jenny M. Eschbacher received the degree from the Medical School, Wayne State University. She has completed the Neuropathology Fellowship with the Barrow Neurological Institute, in 2009, where she practices neuropathology. She is the Clinical Principal Investigator of the Biobank and the Laboratory Director of the Ivy Brain Tumor Center. She has research interests in evaluating brain tumors in vivo with confocal microscopy.

Gustavo De Leon received the B.S. degree in biological sciences from Arizona State University. Since then, he has been a research assistant with Dr. Swanson and is planning to pursue a career as a physician's assistant.

Kyle W. Singleton received the Ph.D. degree in biomedical engineering, specializing in medical imaging informatics from the University of California, Los Angeles, in 2016. He is currently a senior post-doctoral fellow with Dr. Swanson, with research interests in machine/deep learning and image processing.

Lee A. Curtin received the Ph.D. degree in mathematics, specializing in mathematical medicine and biology from the University of Nottingham in 2019. He is currently a staff research scientist position with Dr. Swanson, where he works in developing mathematical models and aids in the image-localized biopsy core effort.

Javier Urcuyo received the B.S. degree in applied mathematics and biology from Arizona State University in 2019. He is currently a research assistant with the Dr. Swanson's Group looking at incorporating RNAseq data into math models of the immune systems.

Chris Sereduk received the B.S. degree from Midwestern University. He is a senior technician and the laboratory manager overseeing tissue processing, genomic analysis of clinical specimens, cell-based assays for high throughput drug screening, and functional screening.

Nhan L. Tran received the Ph.D. degree in cancer biology from the University of Arizona. He is a Professor with the Department of Cancer Biology, Mayo Clinic Arizona. His research interest is in the genetics and molecular and cellular signaling of brain tumor invasion, survival, and therapeutic resistance.

Andrea Hawkins-Daarud received the Ph.D. degree in computational sciences engineering and mathematics from The University of Texas at Austin in 2011. She is the Assistant Director with the Mathematical Neuro-Oncology Group, Mayo Clinic Arizona. Her research interests include mathematical oncology, parameter estimation, and uncertainty quantification.

Kristin R. Swanson received the B.S. degree in mathematics from Tulane University in 1996, the M.S. and Ph.D. degrees in mathematical biology from the University of Washington in 1998 and 1999, respectively, and the Post-Doctoral degree from UCSF. She is currently the Co-Director of the Precision NeuroTherapeutics Program and a Professor and the Vice Chair of Research with the Department of Neurosurgery, Mayo Clinic. She is an internationally recognized mathematical oncologist, focused on delivering optimal treatment to patients with brain cancer. Her expertise bridges mathematical modeling, oncology, artificial intelligence, and cancer biology.

Jing Li (Member, IEEE) received the Ph.D. degree in industrial and operations engineering from The University of Michigan, Ann Arbor, MI, USA. She is a Professor with the School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA. Her research interests are statistical modeling and machine learning for health care applications. She is a member of IISE and INFORMS. She was a recipient of the NSF CAREER Award.