Combating Negative Transfer From Predictive Distribution Differences

Chun-Wei Seah, Yew-Soon Ong, and Ivor W. Tsang

Abstract*—Domain adaptation (DA), which leverages labeled data from related source domains, comes in handy when the label information of the target domain is scarce or unavailable. However, as the source data do not come from the same origin as that of the target domain, the predictive distributions of the source and target domains are likely to differ in reality. At the extreme, the predictive distributions of the source domains can differ completely from that of the target domain. In such case, using the learned source classifier to assist in the prediction of target data can result in prediction performance that is poorer than that with the omission of the source data. This phenomenon is established as *negative transfer* with impact known to be more severe in the multiclass context. To combat negative transfer due to differing predictive distributions across domains, we first introduce the notion of *positive transferability* for the assessment of synergy between the source and target domains in their prediction models, and we also propose a criterion to measure the positive transferability between sample pairs of different domains in terms of their prediction distributions. With the new measure, a *predictive distribution matching* (*PDM*) regularizer and a PDM framework learn the target classifier by favoring source data with large positive transferability while inferring the labels of target unlabeled data. Extensive experiments are conducted to validate the performance efficacy of the proposed PDM framework using several commonly used multidomain benchmark data sets, including Sentiment, Reuters, and Newsgroup, in the context of both binary-class and multiclass domains. Subsequently, the PDM framework is put to work on a real-world scenario pertaining to water cluster molecule identification. The experimental results illustrate the adverse impact of negative transfer on several state-of-the-art DA methods, whereas the proposed framework exhibits excellent and robust predictive performances.

***Index Terms*—Domain adaptation (DA), logistic regression (LR), negative transfer, predictive distribution matching (PDM), support vector machines (SVMs).**

I. Introduction

In TRADITIONAL learning tasks, the decision function f is typically attained by minimizing the expected risk functional of the form

$$\min_{f} \int L(\mathbf{x}, y, f) dP(\mathbf{x}, y) \tag{1}$$

with respect to the joint distribution $P(\mathbf{x}, y)$ of the target domain and a loss function L, where $\mathbf{x}$ and y denote the input feature vector and class label, respectively, of the problem of interest. The joint distribution can be further factorized as $P(\mathbf{x}, y) = P(y|\mathbf{x})P(\mathbf{x})$, where $P(y|\mathbf{x})$ and $P(\mathbf{x})$ are the predictive distribution and marginal distribution of the target domain, respectively. However, when no label information is available in the target domain, $P(y|\mathbf{x})$ cannot be well estimated. To address this, some are turning to domain adaptation (DA) techniques which use labeled data from related source domains to augment the performance of learning tasks on the target domain.

Taking the marketing strategy of sales personnel as a motivating example, for instance, sentiment classification serves as an important task to predict the sentiment polarity of a new product (target domain) based on the multiway scale of user reviews collected from other similar products. Each review is graded based on a five-star rating, and the higher is the rating, the better the feedback is perceived. Since the user review feedbacks are usually described by some common words, the annotated sentiment reviews from several other categories of products (source domains) may benefit the prediction of the star rating on unannotated sentiment reviews of new products (target domain). Hence, DA methods generally assume that the source domains share a similar predictive function with the target domain. Aside from sentiment classification, DA methods are also widely studied in natural language processing [1]–[3], text categorization [4], computer vision [5]–[8], Wi-Fi localization [4], remote sensing [9], and recommendation systems [10]. Other applications in which DA can be useful also exist in abundance, including gene expression data [11], cell-phenotype images [12], and aerodynamic design [13], where labeled data in the target domain of interest are generally scarce.

Recent DA methods [5], [14]–[16] have been proposed for learning from multiple source domains. In [16], for instance, the authors proposed a multiple convex combination of support vector machine (SVM) using the data from multiple source domains and the target domain. Since the data do not come from the same origin, the distributions $P(\mathbf{x}, y)$ of the source and target domains are likely to differ. In such situation, target

Manuscript received March 9, 2012; revised July 10, 2012; accepted September 27, 2012. This work was supported in part by the Singapore Nanyang Technological University, Agency for Science, Technology and Research, Science and Engineering Research Council under Grant 112 172 0013 and in part by the Multi plAtform Game Innovation Centre (MAGIC). MAGIC is supported by the Interactive Digital Media Programme Office (IDMPO) hosted by the Media Development Authority of Singapore. IDMPO was established in 2006 under the mandate of the National Research Foundation to deepen Singapore's research capabilities in interactive digital media and fuel innovation and to shape the future of media. This paper was recommended by Editor S. Hu.

The authors are with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: Seah0116@ntu.edu.sg; asYSOng@ntu.edu.sg; IvorTsang@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2012.2225102

labeled data are often required to measure the relatedness from the source domains [5], [14], [17]; then, source domains can assist in learning the target task. However, when target labeled data are unavailable, DA methods [1], [18]–[20] usually assume that the prediction distribution $P(y|\mathbf{x})$ is shared among different domains and minimize the dissimilarities among the source and target domains with regard to the marginal distribution $P(\mathbf{x})$ only. The dissimilarity in marginal distributions among domains is commonly known as covariate shift [20] that adjusts the weight of each source sample by means of $P^t(\mathbf{x})/P^s(\mathbf{x})$ as a common remedy used to resolve such an issue, with $P^s(\mathbf{x})$ and $P^t(\mathbf{x})$ denoting the marginal distribution of the source domain and target domain, respectively. For example, the kernel-mean matching (KMM) method [18] estimates the weight of each source sample by minimizing the maximum mean discrepancy (MMD) criterion [21] between the source samples and target unlabeled samples; then, reweighted source samples are used for training a classifier for the target data.

As an alternative to reweighting methods, others have also considered the extraction of useful features from the source domains to augment the original feature space to train the classifier [3], [22]–[26] so that the augmented feature space leads to similar marginal distribution between the source and target domains. For instance, an alternative to KMM that minimizes the MMD criterion for the purpose of minimizing the distributions of the source and target domains, minimizing the quadratic distance [27] and geodesic distance [25], is recently proposed. For another instance, the feature augmentation (FA) approach [3] augments features belonging to the same domain by twice that of the original features to bias the classifier in treating the data of the same domain twice as much than those of differing domains. Furthermore, the FA approach is also considered as a multitask learning algorithm since its model parameters θ_r in the rth domain are decomposed as θ_c and θ'_r, where θ_c is shared among all domains and θ'_r is for each individual domain.

In general, multitask methods [28], [29] simultaneously learn the models of all the tasks by sharing some common parameters such that the learned model can classify each individual task well, whereas DA methods focus on classifying well on the target task only. Another major difference is that the source and target tasks in DA are the same but different in data distribution, whereas the tasks in multitask are different but related. In particular, DA methods generally address the marginal distribution differences between the source and target domains, and this paper further combats negative transfer from predictive distribution differences. In contrast, task clustering for multitask learning discovers hidden structure within a set of related tasks for a robust learning [29], [30].

Frankly, each domain has its own predictive distribution $P(y|\mathbf{x})$ in real applications; as a result, the phenomenon of *negative transfer* [31] is known to creep in, leading to the impediment on the performances of DA approaches [31], [32]. Thus, negative transfer can be deem to have occurred when the DA method is observed to deteriorate over the prediction performance of its respective non-DA counterpart. To this end, more recent works, including domain adaptation SVM (DASVM) [9], maximal margin target label learning (MMTLL) [33], and TARget learning Assisted by Source Classifier Adaptation (TARASCA) [34], have also attempted to maintain the consistency of the joint distributions $P(\mathbf{x}, y)$ across the different domains.

In spite of the recent advancements made in DA, many fundamental problems of negative transfer resulting from the differences in predictive distribution have remained unresolved. In particular, to perform well, reweighting methods require the source and target domains to share similar predictive distributions. Furthermore, a considerable large number of target labeled data are typically required to robustly reweight the training instances reliably. Like general DA approaches, feature DA techniques are also plagued by the issues of predictive distribution dissimilarities among the source and target domains. When many overlapping sources and target data with conflicting class label information exist, general DA approaches such as DASVM do not function well. Moreover, DASVM is unable to deal with multiclass problems as the adopted progressive transductive SVM [35] strategy in DASVM is unable to infer multiclass pseudolabels of target unlabeled patterns. In contrast to the proposed method, MMTLL and TARASCA consider classifier/model-based transfer by choosing the weights of the source classifiers (among many source classifiers with different bias parameters) via cluster assumption that exists in the target unlabeled data for positive transfer.

From our survey of the literature, some of the core roots of negative transfer due to predictive distribution differences (which generally violate the assumptions of many DA methods) can be summarized as follows.

1) *Conflicting class labels between related source domains*: The domains contain sample data or clustered data with conflicting class labels. For instance, a domain with a class label which differs from the majority of related domains having a common class label, in some localized region of the vector space, is established as an outlier.
2) *Sample selection bias due to imbalanced class distribution*: The sparsity of labeled data does not serve as good representations of the general population [9], particularly for the *imbalance problem* where bias exists when estimating the target predictive distribution (e.g., by logistic regression (LR) or naïve Bayes classifier); this bias is generally known as the *sample selection bias*.[1] In most cases, the source domains have differing class distributions from the target domain, and these class distribution differences can easily lead to predictive distribution dissimilarities among the domains. Without sufficient label information on the target domain, the true class distribution of the target domain is generally unknown, and resampling strategies (e.g., Synthetic Minority Oversampling Technique [37]) that are designed to adapt the class distributions of the source domains to match with the target domain do not apply well in this setting. Even in the event of high similarity in the feature space, any class distribution differences between the target and source domains can still mislead the learning of the target

[1] Recently, a work [36] considers that the class distribution of the training set differs from the testing set while both sets are from the same domains.

predictive distribution due to the learning biases toward that of the source domains.

Taking these cues, in this paper, we propose a novel DA method, namely, *predictive distribution matching* (PDM), to address the challenges that arise from the predictive distribution differences. The main contributions and core ingredients of the proposed framework are outlined in what follows.

1) A criterion of positive transferability is proposed to measure the differing predictive distributions of the target domain and the related source domains. Using this criterion, a PDM regularized classifier is introduced to infer target pseudolabeled data which subsequently assists in identifying the relevant source data in a manner that the predictive distributions of both source and target data are maximally aligned.
2) To our knowledge, the proposed framework serves as the first attempt to combat negative transfer in multiclass problems, where target labeled data are unavailable or scarce. In particular, a new form of LR, established here as the PDM-LR, is proposed for handling the multiclass DA problem.

This paper extends from the preliminary work [38] and is organized as follows. Section II introduces the proposed PDM framework to match the predictive distributions of the source and target domains in the context of multiclass problems. Instantiations of the PDM framework on LR and SVM are subsequently showcased in Section III. Extensive experimental studies of the PDM framework, pitted against several state-of-the-art DA and traditional algorithms on multiclass and multidomain data sets, including the real-world Sentiment data set, are reported in Section IV. Analysis and discussion pertaining to the experimental results are then provided in Section V. In addition, a novel real-world water-molecule application is showcased in Section VI. The brief conclusions of this paper are then drawn in Section VII. In contrast to this paper, the initial work [38] of this paper focuses on only binary problems and SVM context. The core symbols used throughout the rest of this paper are listed in Table I.

II. PDM Framework

In this section, we present a detailed description of the proposed PDM framework for matching the predictive distributions of the source and target domains, with the work flow of the framework shown in Fig. 1. The work flow presents an iterative process of inferring the target unlabeled data where irrelevant source data are removed such that the target and source data are maximally aligned based on the criterion of positive transferability upon convergence (defined in Section II-A). In particular, the framework begins with the training of a PDM regularized classifier (see Section II-A). Instantiations of the PDM regularized classifier in SVM and LR are subsequently presented in Section III. The PDM regularized classifiers are used to infer the pseudolabeled data (see Section II-B) from the set of target unlabeled data. Source data that do not align with the predictive distribution of the inferred pseudolabeled data are then removed (see Section II-C). The entire process iterates until the stopping criterion is reached. Upon convergence, the SVM classifier or LR classifier is trained using the identified relevant source labeled data and the acquired pseudolabeled data of the target domain, which are then validated on the target testing data.

TABLE I
SYMBOL DEFINITION

Symbol	Definition
m	Total number of domains, the first $(m-1)$ domains represent source domains while m denotes the target domain
C	Number of Class Labels.
$\mathbf{x}_i^r$	Feature vector of ith data of rth domain
y_i^r	Class Label (1,...,C) of the data $\mathbf{x}_i^r$. When $r == m$, it refers to the ith inferred pseudo-label from the target unlabeled data.
n_r	Number of labeled data in rth source domain or the number of pseudo-labeled data in target domain. For simplicity, the index for jth iteration after inferring pseudo-labeled data and removing the irrelevant source data is not shown.
n	$\sum_{r=1}^{m} n_r$
$\mathcal{D}_L$	$\cup_{r=1}^{m-1}\{\mathbf{x}_i^r, y_i^r\}^{n_r}$, all labeled data in all source domains
$\mathcal{D}_L^j$	Remaining of source labeled data in jth iteration. Note, $\mathcal{D}_L^0 = \mathcal{D}_L$
$\mathcal{D}_U$	The set of unlabeled data in target domain
B^j	n_m number of inferred pseudo-labeled data from the target (mth domain) unlabeled data during the jth iteration, i.e., $\{\mathbf{x}_i^m, y_i^m\}_{i=1}^{n_m}$ where y_i^m is the pseudo-label of data $\mathbf{x}_i^m$
$\mathcal{D}_U^j$	$\mathcal{D}_U^{j-1} \setminus B^{j-1}$. Note, $\mathcal{D}_U^0 = \mathcal{D}_U$
η	Number of features
$P^r(\mathbf{x})$	Marginal distribution of rth domain
$P^r(y \mid \mathbf{x})$	Predictive distribution of rth domain
$I(\cdot)$	Indicator function which has a logic of 1 if the predicates hold, otherwise a logic of 0 is given
W_{ij}^{rd}	The positive transferability criterion between the ith and jth samples in rth and dth domains, respectively.

A. PDM Regularization for Multiple Source Domains

Here, our interest is on PDM across multidomains and the multiclass contexts. First, we define the notions of positive transferability and negative transferability.

Definition 1: Positive transferability is introduced as the assessment of the synergy between the source and target domains in their prediction models. In other words, it measures the constructive synergy of the source labeled data in accelerating or enhancing the learning of the prediction model for the target unlabeled data. This is highly plausible when the selected source labeled data and the set of target data share similar predictive models.

Next, the antonym of positive transferability, which is referred as negative transferability, is defined.

Definition 2: Negative transferability is introduced as a measure for the destructive synergy of the source labeled data in enhancing the learning of the prediction model for the target unlabeled data.

With the notion of positive transferability given by Definition 1, we proposed a criterion to measure the degree of positive transferability between a sample pair of different domains in terms of their predictive distributions.

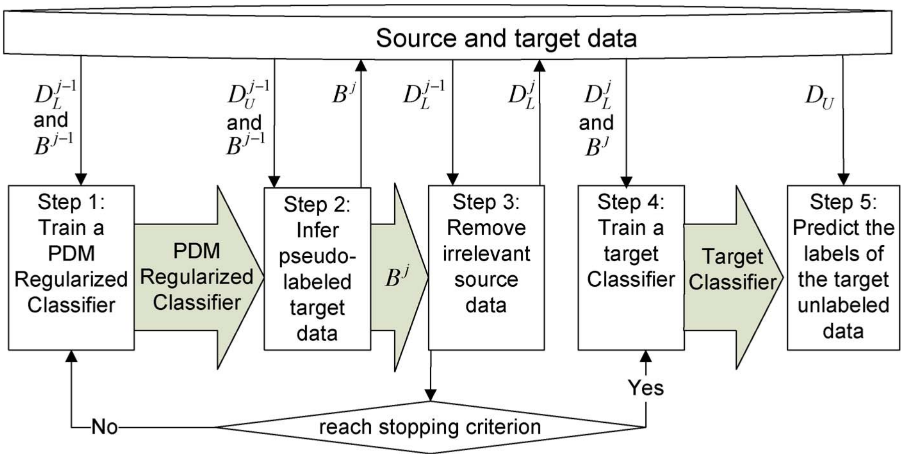

Source and target data
D_L^{j-1} and B^{j-1}
D_U^{j-1} and B^{j-1}
B^j
D_L^{j-1}
D_L^j
D_L^j and B^j
D_U
Step 1: Train a PDM Regularized Classifier
PDM Regularized Classifier
Step 2: Infer pseudo-labeled target data
B^j
Step 3: Remove irrelevant source data
Step 4: Train a target Classifier
Target Classifier
Step 5: Predict the labels of the target unlabeled data
Yes
No
reach stopping criterion

Fig. 1. Work flow of the PDM framework to maximally align the positive transferability criterion. See Table I for the notations.

Definition 3: The criterion of positive transferability between two samples from different domains is defined here as

$$\begin{aligned} W_{ij}^{rd} &= P\left(y_i^r, y_j^d | \mathbf{x}_i^r, \mathbf{x}_j^d\right) S\left(\mathbf{x}_i^r, \mathbf{x}_j^d\right) I\left[y_i^r = y_j^d\right] \\ &= P^r\left(y_i^r | \mathbf{x}_i^r\right) P^d\left(y_j^d | \mathbf{x}_j^d\right) S\left(\mathbf{x}_i^r, \mathbf{x}_j^d\right) I\left[y_i^r = y_j^d\right] \quad (2) \end{aligned}$$

where W_{ij}^{rd} is defined as the product of $S(\mathbf{x}_i^r, \mathbf{x}_j^d)$, which measures the similarity between $\mathbf{x}_i^r$ and $\mathbf{x}_j^d$, and $P(y_i^r, y_j^d | \mathbf{x}_i^r, \mathbf{x}_j^d)$ that describes the synergy between the source and target domains for Definition 1. For the efficient estimation of $P(y_i^r, y_j^d | \mathbf{x}_i^r, \mathbf{x}_j^d)$, a conditionally independent assumption is made for their class labels y_i^r and y_j^d given the inputs $\mathbf{x}_i^r$ and $\mathbf{x}_j^d$ where $P(y_i^r, y_j^d | \mathbf{x}_i^r, \mathbf{x}_j^d) = P^r(y_i^r | \mathbf{x}_i^r) P^d(y_j^d | \mathbf{x}_j^d)$. Furthermore, the similarity $S(\mathbf{x}_i^r, \mathbf{x}_j^d)$ is used to measure how the importance of $P^r(y_i^r | \mathbf{x}_i^r)$ and the importance of $P^d(y_j^d | \mathbf{x}_j^d)$ should be related. In addition, $I[y_i^r = y_i^d]$ indicates that the two samples must have the same labels.

From this definition, positive transferability thus denotes the maximal alignment between $P^r(y_i^r | \mathbf{x}_i^r)$ and $P^d(y_j^d | \mathbf{x}_j^d)$ and their similarity $S(\mathbf{x}_i^r, \mathbf{x}_j^d)$. Note that $P^r(y_i^r | \mathbf{x}_i^r)$ is the predictive distribution of the rth domain on vector $\mathbf{x}_i^r$, which is estimated, for instance, by means of LR [39] or SVM probability [40], using the labeled data in the rth domain.[2] The next challenge is how to maximally align the source and target domains when no or few target labeled data are available. To address this challenge, we propose to infer the pseudolabeled data (see Section II-B) and identify the source data (see Section II-C) via an iterative process that converges when the source and target data are maximally aligned. To infer the pseudolabeled data, we use the positive transferability in Definition 3 to define a PDM regularizer as follows.

Definition 4: A PDM regularizer is defined to minimize the predictive function $\omega(\cdot)$ of a classifier: $(1/n^2) \sum_{r,d=1}^{m} \sum_{i=1}^{n_r} \sum_{j=1}^{n_d} (\omega(\mathbf{x}_i^r) - \omega(\mathbf{x}_j^d))^2 W_{ij}^{rd} I[r \neq d]$ where W_{ij}^{rd} defines two samples drawn from different domains $(I[r \neq d])$. This regularizer enforces data that have high similarity according to the definition of positive transferability W, to share similar predictive outputs.

Definition 4 is motivated by the concept of manifold regularization [41]–[43] where two inputs, $\mathbf{x}_i^r$ and $\mathbf{x}_j^d$, are enforced by W_{ij}^{rd} to share similar predictive outputs of $\omega(\mathbf{x}_i^r)$ and $\omega(\mathbf{x}_j^d)$. Note that W_{ij}^{rd} defines the weight of an edge in a k-nearest-neighbor graph. In traditional manifold regularization, only $S(\mathbf{x}_i^r, \mathbf{x}_j^d)$ is considered, and the data are assumed to originate from a single source. In this paper, the data are from different sources, and W_{ij}^{rd} is defined based on the notation of positive transferability given in Definition 3. Furthermore, the *PDM score* for $\mathbf{x}_i^r$ and $\mathbf{x}_j^d$ is $P^r(y_i^r | \mathbf{x}_i^r) P^d(y_j^d | \mathbf{x}_j^d)$ where both $\sum_{z=1}^{C} P^r(y_z | \mathbf{x}_i^r)$ and $\sum_{z=1}^{C} P^d(y_z | \mathbf{x}_j^d)$ have the value of 1. Since we do not assume any manifold assumption on each domain, the indicator function $I[r \neq d]$ serves to enforce only data of unique domains to associate or pair up. Note that, if the target domain follows a manifold assumption, a manifold regularizer can be incorporated into our formulation seamlessly by simply excluding the $I[r \neq d]$ term. To avoid any loss of generality, here, we consider a formulation that generalizes well.

B. Inferring Target Pseudolabeled Data

In this section, we discuss how to infer pseudolabeled data from the target unlabeled data. For each iteration j, the highest confidence predicted outputs of the target unlabeled data are added to the set of acquired pseudolabeled data

$$B^j = B^{j-1} \cup \left\{ \cup_c^C B_c^j \right\}. \quad (3)$$

The acquired pseudolabeled data of the jth iteration for class c is then

$$B_c^j = \left\{ (\mathbf{x}_i, y_i) \in T_c^j | 1 \leq i \leq \mathfrak{p}_c^j \right\} \quad (4)$$

where $\mathfrak{p}_c^j = \min(\sigma, |T_c^j|)$, $|T_c^j|$ is the cardinality of T_c^j and σ is a relaxation parameter.[3] T_c^j denotes the unlabeled data with class labels c inferred using the PDM regularized classifier f^j

[2] For each source domain ($r < m$), the predictive distributions in (2) are required to compute only once before the iterative process.

[3] A higher value of σ would speed up the process, but at the expense of including less confident pseudolabeled data. Since the efficiency of the process is not a major concern, σ is configured to 1 in our study.

and is sorted in a decreasing order in terms of $P(y = c|\mathbf{x})$ as follows:

$$T_c^j = \Big\{(\mathbf{x}_i, c)|\mathbf{x}_i \in \mathcal{D}_U^j, h(\mathbf{x}_i) = c, P(y = c|\mathbf{x}_i) \geq P(y = c|\mathbf{x}_{i+1})\Big\} \quad (5)$$

where $\mathcal{D}_U^j = \mathcal{D}_U^{j-1} \setminus B^{j-1}$ and $P(y = c|\mathbf{x}_i)$ is the predictive distribution for class c given $\mathbf{x}_i$ that is estimated from both B^{j-1} and $\mathcal{D}_L^{j-1}$ (which is the identified source data in the $(j-1)$th iteration and is presented in the next Section II-C), while the predicted class $h(.)$ is defined as follows:

$$h(\mathbf{x}) = \arg\max_{c \in C} \omega_c(\mathbf{x}) \quad (6)$$

with $\omega_c(\mathbf{x})$ denoting the predictive output of PDM regularized classifier f^j for class c. Hence, B_c^j in (4) (the acquired p_c^j number of pseudolabeled data) represents the data with the highest predictive distribution values in T_c^j. After the new set of pseudolabeled data is formed in (3), the PDM framework reestimates $P^m(y|\mathbf{x})$, for instance, by means of LR. The inferred pseudolabeled data formed in (3) can then be used to compute the positive transferability on (2) for the next iteration.

C. *Removing Irrelevant Source Data*

In practice, some source data may not align with the predictive distribution of the inferred target pseudolabeled data. Hence, in this section, we discuss how these irrelevant source data are removed.

Without loss of generalities, the remaining source labeled data at the jth iteration can be defined as

$$\mathcal{D}_L^j = \mathcal{D}_L^{j-1} \setminus \left\{\cup_c^C \mathcal{D}_c\right\} \quad (7)$$

where $\mathcal{D}_L^0$ is the initial set of source labeled data $(\mathcal{D}_L)$ and $\mathcal{D}_c$ is the set of data grouped according to their true class label y_i. Each grouped set is then sorted according to their estimated predictive distribution values in ascending order as follows:

$$\mathcal{D}_c = \Big\{(\mathbf{x}_i, y_i)|\mathbf{x}_i \in \mathcal{D}_L^{j-1}, y_i = c, P^m(y = c|\mathbf{x}_i) \leq P^m(y = c|\mathbf{x}_{i+1}) \leq \pi, 1 \leq i \leq \gamma\Big\} \quad (8)$$

where π denotes the minimum level of confidence for any data vectors to be retained and P^m is estimated by using the pseudolabel data B^j. The removed source data for each class $\mathcal{D}_c$ are the lowest γ consistence data points[4] with respect to the pseudolabeled data. After convergence is reached, all source labeled data with $P^m(y|\mathbf{x}, f_t^j) \leq \pi$ are removed by simply setting γ to ∞.

Since the inferring process is designed to iteratively select the highly confident pseudolabeled data, it is natural to end the PDM process when the inferred labels of the pseudolabeled data fail to measure up to the given confidence level. In this paper, parameter δ is used to control the level of confidence in the pseudolabeled data as

$$\min_{\mathbf{x}_i \in B^j} P^m(y_i|\mathbf{x}_i) \leq \delta. \quad (9)$$

III. PDM Regularized Classifier Instantiations

In this section, instantiations of the PDM framework with LR (PDM-LR) and SVM (PDM-SVM) are presented as the PDM regularized classifier.

A. *PDM LR Classifier (**PDM-LR**)*

LR is primarily popular in the context of text classification. On multiclass classification problems, the predictive distribution $P(y = c|\mathbf{x})$ of a class c is defined as follows:

$$\omega_c(\mathbf{x}) = P(y = c|\mathbf{x}) = \frac{e^{\boldsymbol{\beta}'_c \mathbf{x}}}{\sum_{z=1}^{C} e^{\boldsymbol{\beta}'_z \mathbf{x}}} \quad (10)$$

where $\boldsymbol{\beta}_c$ is the weight vector for class c and $\omega_c(\mathbf{x})$ is the predictive output for class c. In LR, both $P(y = c|\mathbf{x})$ and $\omega_c(\mathbf{x})$ have the same value, and $\sum_c^C \omega_c(\mathbf{x}) = 1$ can be regarded as a form of probability measure. On multiclass LR, minimizing the negative log likelihood of (10) becomes

$$g_1(\boldsymbol{\beta}) = -\sum_{r=1}^{m} \sum_{i=1}^{n_r} w_i^r \boldsymbol{\beta}'_{y_i^r} \mathbf{x}_i^r + \sum_{r=1}^{m} \sum_{i=1}^{n_r} w_i^r \log \sum_{z=1}^{C} e^{\boldsymbol{\beta}'_z \mathbf{x}_i^r} \quad (11)$$

where w_i^r is the weight[5] of the ith sample in the rth domain and $\boldsymbol{\beta} = [\boldsymbol{\beta}_{\mathbf{1}}, \ldots, \boldsymbol{\beta}_{\boldsymbol{C}}]$. To prevent overfitting, a regularizer with parameter C_1 is typically incorporated

$$g_2(\boldsymbol{\beta}) = \frac{1}{2} C_1 \|\boldsymbol{\beta}\|_2^2. \quad (12)$$

Hence, a regularized multiclass LR is defined as

$$\arg\min_{\boldsymbol{\beta}} g_1(\boldsymbol{\beta}) + g_2(\boldsymbol{\beta}). \quad (13)$$

Next, with the $\omega(\mathbf{x})$ of the PDM regularizer in Definition 4 given by $\boldsymbol{\beta}'_y \mathbf{x}$, the resultant PDM regularizer is then formulated as follows:

$$g_3(\boldsymbol{\beta}) = \frac{C_2}{2n^2} \sum_{r,d=1}^{m} \sum_{i=1}^{n_r} \sum_{j=1}^{n_d} \left(\boldsymbol{\beta}'_{y_i^r} \mathbf{x}_i^r - \boldsymbol{\beta}'_{y_j^d} \mathbf{x}_j^d\right)^2 W_{ij}^{rd} I[r \neq d] \quad (14)$$

where C_2 denotes the parameter that regulates the importance of PDM. Hence, combining the PDM regularizer with (13), the proposed PDM multiclass LR or **PDM-LR** in short becomes

$$\arg\min\nolimits_{\boldsymbol{\beta}} G(\boldsymbol{\beta}) = \arg\min\nolimits_{\boldsymbol{\beta}} g_1(\boldsymbol{\beta}) + g_2(\boldsymbol{\beta}) + g_3(\boldsymbol{\beta}). \quad (15)$$

[4] Note that a higher γ value in (7) speeds up the process, but it will remove high confidence source labeled data that are relevant to the target domain. As speeding up the process is not our main objective here, γ is configured as 1 in our study.

[5] This weight can be adjusted to define the importance of each data vector sample to deal with the marginal distribution differences among domains, often known as the reweighting method [18]. Since the focus here is on predictive distribution differences, we treat all data samples equally in this paper, i.e., assigning each w_i^r to $1/n$.

1) Convergence Analysis: In what follows, the details toward solving (15) and the properties of convergence are presented. Note that the PDM regularizer in Definition 4 represents a special case of [41] which is positive semidefinite, and LR is strongly convex [39]. Hence, the resultant PDM-LR is also strongly convex. Thus, the problem can be solved using convex optimization technique, such as the coordinate descent method due to its simplicity and efficiency, since the computation of the entire Hessian matrix is not required. The coordinate descent method is composed of an outer and an inner loop. The inner loop conducts the Newton descent search on a dimension while the outer loop checks for convergence. An outline of the coordinate descent method is depicted in Algorithm 1.

Algorithm 1 Coordinate descent method

```
repeat
    for $p = 1$ & $c = 1$ TO $\eta$ & $C$, respectively, do
        Solve $min_{\beta_{pc}} G(\boldsymbol{\beta})$, by means of approximation, to obtain $z$.
        $\beta_{pc} = \beta_{pc} + z$
    end for
until $\boldsymbol{\beta}$ is optimal
return $\boldsymbol{\beta}$
```

To attain the gradient and Hessian information of each coordinate, the derivations for LR, (13), follow that of [39]. The gradient and Hessian formulations for the PDM regularizer in (14), on the other hand, are derived in what follows. The first derivation of (14) can be derived as

$$\begin{aligned}\frac{\partial g_3}{\partial \beta_{pc}} &= \frac{C_2}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\big(W_{ij}\left(\boldsymbol{\beta}_c'\mathbf{x}_i - \boldsymbol{\beta}_c'\mathbf{x}_j\right) \\ &\qquad \times \left(I[y_i = c]x_{ip} - I[y_j = c]x_{jp}\right)\big) \\ &= \frac{C_2}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(W_{ij} I[y_i = c]\left(\boldsymbol{\beta}_c'\mathbf{x}_i - \boldsymbol{\beta}_c'\mathbf{x}_j\right)\right. \\ &\qquad \left.\times (x_{ip} - x_{jp})\right) \end{aligned} \tag{16}$$

since W_{ij} consists of $I[y_i = y_j]$ where the pc of β_{pc} is the pth dimension and the class c of $\boldsymbol{\beta}$. Note that, for the sake of conciseness in (16), the notation for the PDM regularizer in (14) is simplified with the removal of domain indexes, i.e., r and d, since the indicator, $I[r \neq d]$, is a precomputed value and will implicitly inherit the domain indexes. Without loss of generality, the PDM regularizer in (14) simplifies to

$$g_3(\boldsymbol{\beta}) = \frac{C_2}{2n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\boldsymbol{\beta}_{y_i}'\mathbf{x}_i - \boldsymbol{\beta}_{y_i}'\mathbf{x}_j\right)^2 W_{ij}. \tag{17}$$

In addition, the second-order information, i.e., the Hessian of (17), is then derived as

$$\frac{\partial^2 g_3}{\partial^2 \beta_{pc}} = \frac{C_2}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} W_{ij} I[y_i = c](x_{ip} - x_{jp})^2. \tag{18}$$

In what follows, the details of updating β_{pc} using a scaling factor z that meets the sufficient decrease condition will be discussed. $\mathbf{E}(p, c) \in \Re^{\eta \times C}$ defines the direction of dimension p and class c for updating the value β_{pc} as

$$E_{ij}(p, c) = \begin{cases} 1, & i = p, j = c \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Updating of new β_{pc}^{new} as $\beta_{pc}^{new} = \beta_{pc}^{old} + z$ is equivalent to performing $\boldsymbol{\beta} + z\mathbf{E}(p, c)$. With this update, the value of PDM-LR function G in (15) decreases as follows:

$$D\left(z\mathbf{E}(p, c)\right) = G\left(\boldsymbol{\beta} + z\mathbf{E}(p, c)\right) - G(\boldsymbol{\beta}). \tag{20}$$

The sufficient decrease condition [39] of $z\mathbf{E}(p, c)$ is given by

$$D\left(z\mathbf{E}(p, c)\right) \leq \sigma z G'(\beta_{pc}) \tag{21}$$

where $z = \lambda d$ and $d = -(G'(\beta_{pc})/G''(\beta_{pc}))$ denotes the Newton direction. From [39, Theorem 4], there exists the parameter $\lambda = 1, 0.5, 0.5^2, 0.5^3, \ldots$ that satisfies the condition in (21) with $\sigma \in (0, 0.5)$. Here, our search begins with $\lambda = 1$, followed by a check on the sufficient decrease condition of (21). If the condition is violated, λ is reduced by half repeatedly, until the inequality of (21) is satisfied. The $D(z\mathbf{E}(p, c))$ of each component in (15) can be formulated as $D(z\mathbf{E}(p, c)) = D_1(z\mathbf{E}(p, c)) + D_2(z\mathbf{E}(p, c)) + D_3(z\mathbf{E}(p, c))$, where D_1, D_2, and D_3 denote the reduction values of the function g_1, g_2, and g_3 in (15), respectively. While the derivations of D_1 and D_2 follow that of [39], D_3, on the other hand, is derived as

$$\begin{aligned} &D_3\left(z\mathbf{E}(p, c)\right) \\ &\quad = g_3\left(\boldsymbol{\beta} + z\mathbf{E}(p, c)\right) - g_3(\boldsymbol{\beta}) \\ &\quad = z\frac{C_2}{n^2}\sum_{i=1,j=1}^{n} 2W_{ij} I[y_i = c] \\ &\qquad \times \left((x_{ip} - x_{jp})\boldsymbol{\beta}_c'\mathbf{x}_i + (x_{jp} - x_{ip})\boldsymbol{\beta}_c'\mathbf{x}_j\right) \\ &\qquad + z^2\frac{C_2}{n^2}\sum_{i=1,j=1}^{n} \\ &\qquad \times \left(W_{ij} I[y_i = c]\left(x_{ip}^2 + x_{jp}^2 - 2x_{ip}x_{jp}\right)\right). \end{aligned} \tag{22}$$

The stopping criterion of the outer loop is defined as $\|G'\|_2^2 < \epsilon$, where ϵ is a predefined parameter (see G' in (21)).

2) Computational Complexity of Algorithm 1: From [39, Theorem 5], it can be derived that the gradient, Hessian, and reduction functions of g_1 and g_2 in (15) have a total computational complexity of $O(n)$. In PDM, (16), (18), and (22) are computed. It is worth noting that the matrix W of PDM is sparse, with at most k nonzero values that denote the k nearest neighbor of each sample [41]. Since matrix W has only at most kn nonzero values, the computations of (16), (18), and (22) equate to $O(kn)$ per β_{pc}. Since (18) and the second component of (22) are independent of $\boldsymbol{\beta}$, both need to be computed only once and can be cached for subsequent reuse throughout the coordinate descent process. From the aforementioned computational analysis, the computational complexity of each inner loop in Algorithm 1 totals to $O(knC\eta)$. This, however, can be reduced to $O(kn\eta)$ due to the existence of the $I[y_i = c]$ term in

TABLE II
INVESTIGATED TRADITIONAL SUPERVISED, SEMISUPERVISED, AND DA STATE-OF-THE-ART ALGORITHMS

	Algorithm	Descriptions	Training-set		Multi-
			$\mathcal{D}_L$	$\mathcal{D}_U$	class
Traditional Methods	Support Vector Machine (*SVM*)	Defined in (23) which maximizes the margin of separation while minimizing the loss function on the training set.	✓		✓
	Logistic Regression (*LR*)	Defined in (13), which maximizes the log likelihood on the training set.	✓		✓
	Laplacian Support Vector Machine (*LapSVM*)	[44] is a primal solver for [41] that considers the manifold structure of the entire training set.	✓	✓	
DA Methods	Feature Augmentation on LR (*FA-LR*)	[3] uses the LR classifier in (13) and augments the original feature space.	✓		✓
	Kernel Mean Matching (*KMM*)	Kernel Mean Matching re-weight each of the source samples based on minimizing the Maximum Mean Discrepancy (MMD) criterion on between the source and target data [18]. Then, a weighted SVM is trained on the weighted source samples.	✓	✓	✓
	Domain Adaptation SVM (*DASVM*)	[9] iteratively reconstructs the margin of separation by removing source labeled data from the training set, and at the meanwhile, incorporates pseudo-labeled data into the training set. At each iteration, the training set is trained using SVM. Eventually, the training set will only consist pseudo-labeled data.	✓	✓	
	Predictive Distribution Matching LR (*PDM-LR*)	As described earlier in Section III-A and depicted in Figure 1.	✓	✓	✓

(16), (18), and (22). Moreover, since k is a constant and usually small, it can be ignored to finally arrive at $O(n\eta)$. Furthermore, it is worth noting that the outer loop in Algorithm 1 typically takes only very few iterations to reach convergence, i.e., ten iterations, as will be shown and discussed later in the experimental study (see Section V-C). Hence, the computational complexity of Algorithm 1 can be arrived as $O(n\eta)$.

B. PDM SVM Classifier (**PDM-SVM**)

When a classification problem is governed by a nonlinear function, SVM fits in nicely with its properties of integrating nonlinear kernel for nonlinear and binary classification problems with ease. The SVM takes the form of

$$\arg\min_{\omega} \sum_{r=1}^{m} \sum_{i=1}^{n_r} w_i^r \ell\left(y_i^r, \omega\left(\mathbf{x}_i^r\right)\right) + \frac{C_1}{2}\|\omega\|^2 \tag{23}$$

where w_i^r denotes the weight[6] of the ith sample in the rth domain, $\ell(\cdot)$ defines the hinge loss function, i.e., $max(0, 1 - y_i^r\omega(\mathbf{x}_i^r))$, $y_i^r \in \{-1, 1\}$, and C_1 is the parameter that defines the tradeoff between classification errors on the labeled samples and model complexity. Note that, in SVM, $h(\mathbf{x})$ in (6) is defined by $sign(\omega(\mathbf{x}))$. The decision function of SVM takes the form of

$$\omega(\mathbf{x}) = \sum_{r=1}^{m} \sum_{i=1}^{n_r} \alpha_i^r K\left(\mathbf{x}, \mathbf{x}_i^r\right). \tag{24}$$

Incorporating the PDM regularizer within SVM, which we denote here as the **PDM-SVM**, the resultant formulation becomes

$$\min_{\omega} \sum_{r=1}^{m} \sum_{i=1}^{n_r} w_i^r \ell\left(y_i^r, \omega\left(\mathbf{x}_i^r\right)\right) + \frac{C_1}{2}\|\omega\|^2 + \frac{C_2}{2n^2} \sum_{r,d=1}^{m} \sum_{i=1}^{n_r} \sum_{j=1}^{n_d} \left(\omega\left(\mathbf{x}_i^r\right) - \omega\left(\mathbf{x}_j^d\right)\right)^2 W_{ij}^{rd} I[r \neq d] \tag{25}$$

where C_2 is the parameter that regulates the importance toward PDM. Note that (25) can be solved using the Laplacian SVM algorithm described in [44], which has a computation complexity of $O(n^2)$. The predictive distribution for SVM is then estimated by

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{A\omega(\mathbf{x})+B}} \tag{26}$$

where both A and B are determined by means of maximizing the log likelihood on the training data [40]. From (26), $P(y = -1|\mathbf{x}) = 1 - P(y = 1|\mathbf{x})$.

IV. EXPERIMENTAL SETUP

In this section, an experiment study of the proposed PDM framework, i.e., PDM-LR, is carried out on synthetic multiclass and multidomain data sets and a real-world Sentiment problem.

A. State-of-the-Art Algorithms

In this paper, a plethora of supervised, semisupervised, and DA state-of-the-art algorithms are considered for comparison as summarized in Table II, where $\mathcal{D}_L$ denotes the data of all the available source domains and $\mathcal{D}_U$ is the target unlabeled data. Furthermore, Table II indicates which algorithms can be directly applied to address the multiclass problem. Since the proposed PDM is integrated with LR for multiclass problems, the variants of LR algorithms considered include LR, FA-LR, and PDM-LR in the experimental study.

The parameters of all the methods are configured by means of k-fold source-domain cross-validation, which represents an extension of the k-fold cross-validation for DA suggested in [45]. Specifically, each partition is a source domain in the k-fold source-domain cross-validation. In addition, a linear kernel is used in the SVM considered in this section, due to its popularity in the text classification domain.

[6] In this paper, our focus is on predictive distribution differences, so each w_i^r is treated equally by assigning them as $1/n$.

TABLE III
NUMBER OF DATA FOR EACH CLASS IN EACH DOMAIN ON SENTIMENT DATA SET

Domain Star Rating	1	2	4	5
Book	1386	1400	1381	1334
DVDs	1321	1244	1273	1280
Electronics	1468	1466	1483	1484
Kitchen Appliances	1300	1313	1262	1274

TABLE IV
NUMBER OF DATA FOR EACH CATEGORY IN EACH DOMAIN ON REUTERS DATA SET

Domain	Category	♯ data
Reuter-1	excs.nyse	76
	orgs.ec	318
	peo.reagan	127
Reuter-2	excs.nasdaq	63
	orgs.imf	96
	peo.james-baker	97
Reuter-3	excs.amex	53
	orgs.opec	87
	peo.nakasone	53
Reuter-4	excs.cbt	30
	orgs.worldbank	61
	peo.volcker	40
Reuter 5	excs.tse	17
	orgs.gatt	63
	peo.lawson	34

B. Data Sets

Here, the commonly used data sets, including Sentiment, Reuters-21578, and Newsgroup-20, are considered. Sentiment is a popular multidomain benchmark data set defined in [46]. It is typically used in the context of DA and is used here to synthesize the presence of diverse class distributions in the source and target domains, for the purpose of investigating on the robustness of traditional machine learning and DA methods. On the one hand, Reuters data set allows one to study the efficacy of the methods in the presence of uneven class distribution in each domain. On the other hand, Newsgroup-20 enables the study on the existence of similar class distribution in the domain. The brief descriptions of the Sentiment, Reuters-21578, and Newsgroup-20 data sets are summarized in Tables III–V, respectively. In the experimental study, the Sentiment, Reuters-21578, and Newsgroup-20 data sets are preprocessed by extracting only the single terms, removing all stop words, performing stemming, and normalizing each feature. Each feature of the review is then represented by its respective *term frequency-inverse document frequency* value.

1) Multidomain Sentiment Data Set: The multidomain Sentiment data set is generated from *Amazon.com* and comprises four categories of product reviews: *Book*, *DVDs*, *Electronics*, and *Kitchen Appliances*. The data set consists of five-star rating for each review, but the third-star rating is removed to avoid ambiguity [46] in the binary classification problem. Hence, only ratings 1, 2, 4, and 5 are considered. For each task, one category forms the target domain while the rest are treated as related source domains. In the target domain, all available samples form the unlabeled data. In each source domain, 200 samples are randomly chosen to form the labeled data. Each task is repeated ten times, and the average performances are reported.

TABLE V
NUMBER OF DATA FOR EACH CATEGORY IN EACH DOMAIN ON NEWSGROUP DATA SET

Domain	Category	♯ data
Newsgroup-1	comp.graphics	970
	rec.autos	983
	sci.crypt	989
Newsgroup-2	comp.windows.x	983
	rec.motorcycles	991
	sci.electronics	981
Newsgroup-3	comp.sys.mac.hardware	953
	rec.sport.hockey	990
	sci.space	984
Newsgroup-4	comp.sys.ibm.pc.hardware	974
	rec.sport.baseball	984
	sci.med	975

In the binary problem, star ratings 1 and 2 form the positive data, while 4 and 5 form the negative data. To study the mismatch of predictive distributions between the source and target domains, five different positive class ratio (PCR) settings are generated here for investigations. The five PCR settings are chosen from 0.3 to 0.7 at an incremental step size of 0.1. Note that the PCR value defines the percentage of positive data in the source domains. A PCR setting of 0.3, for example, indicates that 60 data vectors have positive class labels while the rest have a negative label, out of the 200 selected data in each source domain.

In the multiclass problem, each star rating is equivalent to a class. Coincidently, the data set consists of an even class number; hence, it becomes possible to study the mismatch in predictive distributions between the source and target domains based on binary problem settings. The same five PCR settings are also generated for the source domains, and the PCR value denotes the total percentage of the reviews of star ratings 1 and 2 in the source domain. In addition, the number of star-rating reviews are chosen to be equal for 1 and 2 and 4 and 5. For a PCR value of 0.3, for example, out of the 200 selected data in each source domain, star ratings 1 and 2 each have 30 reviews while star ratings 4 and 5 each have 70 reviews.

2) Multidomain Reuters Data Set: Three out of four main categories of the Reuters data set, namely, *People(Peo)*, *Organizations(Orgs)*, and *Exchanges(Excs)*, are considered in this paper. The *Places* category is not considered in this paper due to the vast instances belonging to this category that would overwhelm all other categories, thus making the study fruitless. In each task, the kth largest subcategory of a main category is considered as the kth domain. The groupings of the domains are detailed in Table IV. Mainly, the largest subcategory is used as the target domain (Reuter-1) while the remaining four largest subcategories form the related source domains (Reuter-2 to Reuter-5).

In the binary context, three resultant tasks are investigated in total: *Peo* versus *Orgs*, *Peo* versus *Excs*, and *Orgs* versus *Excs*. For each task, the mth domain of a category is labeled as positive, while the mth domain in another category forms the negative data. Note that this experimental setting is consistent to the other works [47]–[49] that also considered the Reuters data set.

In the multiclass context, each category is treated as an individual class. All data in the source domains are used as labeled data, and for the target domain, the entire data set is used as unlabeled data. Note that this data set contains imbalance positive and negative samples in each subcategory; hence, the class distribution of the target domain is imbalanced, and the predictive distributions of the source domains are quite diverse with respect to one another.

3) Multidomain Newsgroup Data Set: The three main categories of the data set are *comp*, *rec*, and *sci*. We considered four subcategories in each main category and grouped these four subcategories into four domains as described in Table V.

In the multiclass context, each category is treated as an individual class. Since each domain has a significant number of data to be used as the target domain, we generated four tasks from these groupings. Task m uses Newsgroup-m as the target domain, and the rest forms the source domains. All data in the source domains form the labeled data while the entire data set of the target domain serves as the unlabeled data.

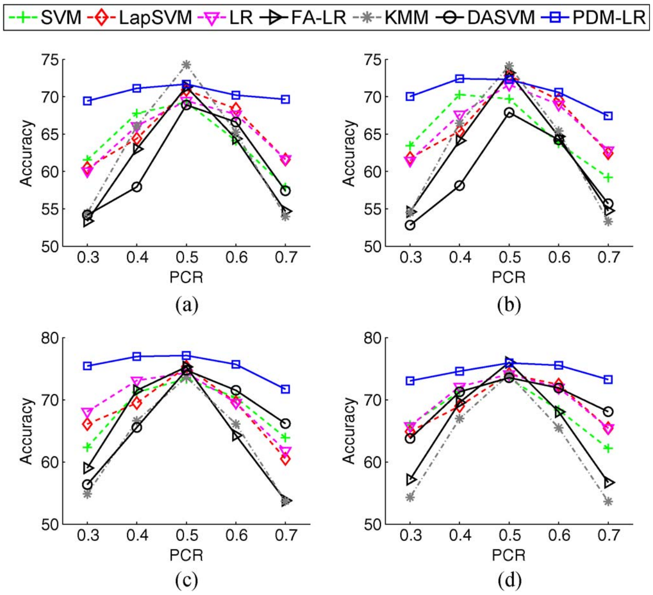
V. Experimental Study

In this section, we present an empirical study of the PDM framework, particularly PDM-LR, on several commonly used DA benchmark text classification data sets.

A. Binary Classification DA

We begin our study on the performances of various classifiers for each of the four domains in the Sentiment data set, i.e., Book, DVDs, Electronics, and Kitchen Appliances, when used independently as the target domain, are summarized in Fig. 2(a)–(d), respectively. For each of the subfigures, it depicts the testing accuracies obtained for five different PCR settings of the source domains at 0.3 to 0.7. On the other hand, the PCR of the target unlabeled data set is configured at 0.5. Hence, when the PCR of the source domains is also in the region of 0.5, the predictive distributions of the source domains are likely to match that of the target unlabeled data set. Any other PCR settings, on the other hand, would likely result in mismatch of the predictive distributions between the source and target domains.

Note that each of the subgraphs exhibits similar performance trends, where all of the classifiers considered in the study perform optimally at a PCR value of 0.5, while displaying sharp declining accuracies when the PCR is skewed toward either extreme ends, except for the proposed PDM-LR which is designed to handle data sets with unbalanced class labels. It is notable that the larger the discrepancies in predictive distributions between the source and target domains, the greater is the bias found in the target prediction accuracies by the classifiers, which are geared toward the source domains. At both extreme ends of the PCR settings, LapSVM which represents an extension of the traditional SVM is noted to generally fare better than SVM since the former acts to evolve the predictive distribution of the labeled data toward that of the unlabeled data. However, just using the unlabeled data alone in LapSVM does not resolve the issues pertaining to differing predictive distributions between the source and target domains (as denoted by the values of PCR $\neq 0.5$) since LapSVM is observed to underperform PDM-LR in Fig. 2.



(a) (b)

(c) (d)

Fig. 2. Binary-class Sentiment data set: Testing accuracies for varying PCR in source domains. Subfigures (a)–(d) represent the target domain of Book, DVDs, Electronics, and Kitchen Appliances, respectively.

At a PCR of 0.5, the target domain shares similar predictive distribution to the source domain. Thus, it is natural to expect the DA algorithms to exhibit similar performances with the traditional classifiers. Nevertheless, FA-LR is observed to obtain improved accuracy over LR in all the subgraphs. In addition, KMM and DASVM also outperform SVM in Fig. 2(a)–(c), respectively. Recall that FA-LR, KMM, and DASVM are the DA versions of the LR and SVM, respectively. The observed improvements thus suggest that FA-LR, KMM, and DASVM can only be beneficial when the predictive distributions among domains match. In either extreme ends of the PCR values, the FA-LR, KMM, and DASVM adaptation methods have reported poorer accuracy compared to their respective non-DA counterparts. Hence, as a summary, it is notable that, when the predictive distributions among domains do not match, DA algorithms generally report lower accuracies than their respective non-DA counterparts; thus, DA algorithms are prone to the effects of negative transfer.

In contrast to existing DA approaches, which suffer from performance degradation due to the effects of negative transfer, PDM-LR effectively unearths the useful knowledge that lies inherent within the multisource domains by means of prediction distribution matching, to arrive at the robust prediction performance observed on the Sentiment data target testing set. In particular, while DA methods in either extreme ends of the PCR values displayed poor prediction accuracies of around 55%, PDM-LR, on the other hand, shows an impressive gain of 15% improvements at an accuracy of 70%. Furthermore, PDM-LR is able to deliver stable results with accuracies that do not deviate over 5% across all the PCR settings. This demonstrates the robustness and reliability of the PDM-LR under different PCR settings by benefiting from the positive transferability facilitated in the proposed framework.

We further experimented the classifiers on the Reuters data set. The results obtained are summarized in Fig. 3. Overall,

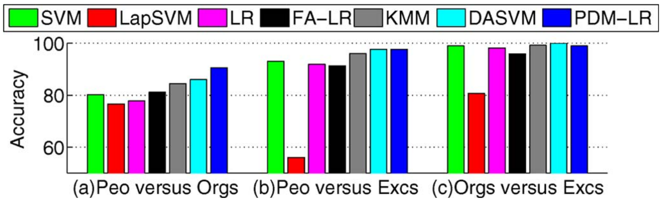


Fig. 3. Testing accuracies on binary-class Reuters data set.

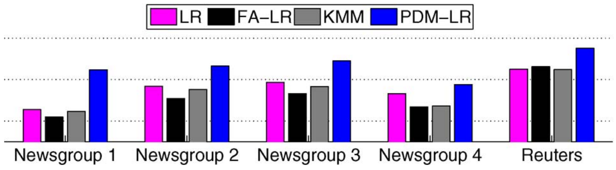
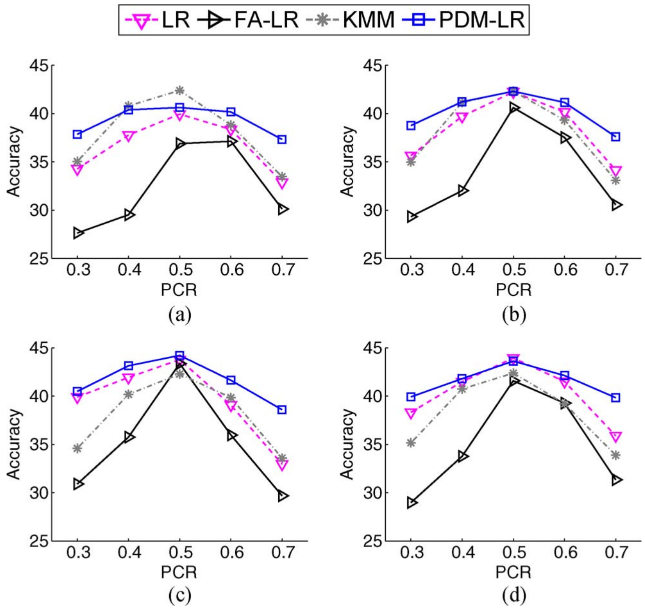


Fig. 5. Testing accuracies on the three-class Newsgroup and Reuters data sets. Compared multiclass problem on various types of LR methods.



(a)

(b)

(c)

(d)

Fig. 4. Four-class Sentiment data set: Testing accuracies for varying PCR in source domains. Subfigures (a)–(d) represent the target domain of Book, DVDs, Electronics, and Kitchen Appliances, respectively. Compared multiclass problem on various types of LR methods.

it is observed that PDM-LR, KMM, and DASVM outperform all other classifiers considered on the Reuters data set. On the other hand, FA-LR which can also be considered as a multitask learning method for learning a shared parameter model under the unique predictive distribution of each domain can lead to negative transfer since FA-LR underperforms LR in Fig. 3(b) and (c). Notably, Fig. 3(a) shows significant accuracy improvements of PDM-LR over KMM, DASVM, and all others. In Fig. 3(b) and (c), PDM-LR is also shown as competitive to KMM and DASVM. Note that SVM is observed to reach a near full score of 100% accuracy in Fig. 3(c). This suggests the high similarities in predictive distributions among the source and target domains. It is thus reasonable for PDM-LR to perform close to KMM and DASVM. Nevertheless, PDM-LR is generally superior to LR.

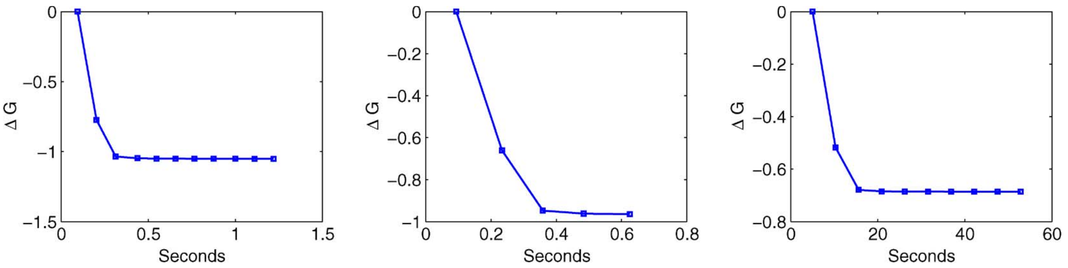
B. Multiclass Classification DA

The prediction trends of the DA and LR classifiers in the multiclass setting of the Sentiment problem in Fig. 4 are observed to be in agreement with those obtained on the two-class setting (as shown in Fig. 2). Fig. 4 shows that PDM-LR generally outperforms the other counterpart algorithms in almost all the different PCR configurations on the multiclass Sentiment data set. Furthermore, PDM-LR delivers stable results with accuracies that do not deviate over 5% across all the PCR settings. In the multiclass setting, the negative transfer phenomenon of DA is more adverse as shown in the results of FA-LR. This is because the likelihood for differing predictive distributions among domains is likely to happen since more class distributions are considered. Hence, PDM becomes ever more challenging in the multiclass classification context.

The multiclass Newsgroup and Reuters experimental results are next summarized in Fig. 5. Although the class distributions of the Newsgroups are similar across domains, there are signs of the DA methods suffering from negative transfer since FA-LR is observed with lower accuracies than the traditional LR. In addition, KMM also has poorer accuracies than LR. In particular, the causes of low prediction accuracy on the testing set for FA-LR and KMM are likely to be a result of the differing predictive distributions between the training and testing sets and also likely the reason for the lack of robustness in the performances of KMM and FA-LR when compared to LR across the range of PCR values in both the Sentiment and Newsgroup data sets (see Figs. 2, 4, and 5). It is also worth noting in Fig. 5 that LR fares generally better in prediction accuracy on the Reuters than the Newsgroup data sets. Taking this cue, we infer the predictive distributions of the domains in the Reuters data set to bare greater similarities. In comparison to all algorithms, PDM-LR overall exhibits at least 5% improvements in accuracy on all the data sets while attaining at least 10% accuracy enhancements on four out of the five data sets considered. Furthermore, an impressive improvement of up to 20% is observed on Newsgroup-1.

C. Computational Complexity of PDM-LR Regularized Classifier

In this section, we analyze the computational complexity of the PDM-LR regularized classifier via empirical study. Fig. 6 summarizes the computational effort involved in the training of a PDM-LR regularized classifier on the various data sets considered. It is observed that the coordinate descent method as described previously in Section III-A2 takes around ten iterations in the outer loop to solve the PDM-LR regularized classifier. The experimental results thus confirm our theoretical complexity analysis of PDM-LR as $O(n\eta)$. In addition, the time taken by the classifier in each algorithm is shown in Fig. 7. In particular, the PDM-LR regularized classifier takes an addition of 0.4 s or twice the time for training compared to LR where the additional time is mainly for computing the regularizer term of PDM-LR.



(a) (b) (c)

Fig. 6. Computational time effort incurred to train the PDM-LR regularized classifier on multiclass settings with Kitchen Appliances with PCR $= 0.5$, Reuters, and Newsgroup-1 as the target domains. Each point in the subplots represents the ith iteration for the outer loop of the coordinate descent method in PDM-LR. The x-axis denotes the time taken in seconds while the y-axis denotes the $\Delta G = G(\boldsymbol{\beta}^i) - G(\boldsymbol{\beta}^0)$ at the ith iteration. The algorithm terminates when $\|G'\|_2^2 < 0.00001$.

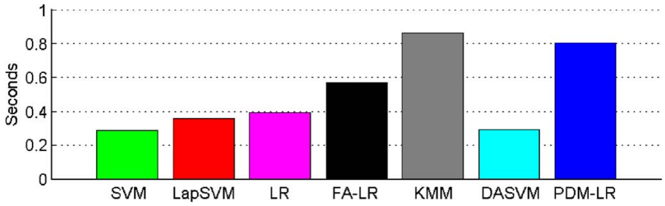


Fig. 7. Time taken for training a classifier in each algorithm under binary-class Kitchen Appliances with PCR $= 0.5$.

VI. Real-World Complex Problem: Water Molecules

In this section, we present the application of the proposed approach to the water isomer discovery problem [50]. Water clusters are crucial for understanding the enigmatic properties of water. They are analyzed in biology to study hydrophobic and hydrophilic interactions and elucidate water's role in biochemical processes which include ligand docking and protein folding [50]. Water clusters are also investigated in physical chemistry to discover the fundamental molecular interactions and collective effects of the condensed phase (liquid and ice) [51]. The identification of water isomers, which are low-energy stable and metastable molecular structures of pure water clusters, is important to study the key properties of the structures.

Obtaining the true computational design of water isomers using mechanical calculation, such as B3LYP [52], is often computationally intractable without the availability of some supercomputing facilities, particularly on large-scale water clusters. To overcome the issue of computational intractability, cost-effective empirical models, including OSS2 [53] and TTM2.1-F [54], have been developed and employed as alternatives to their computationally expensive counterparts. Using the sample sets of isomers collected from past sampling on the different models, we aim to predict the true water isomers in B3LYP, thus reducing the time effort that would otherwise be spent on exhaustive sampling using the expensive mechanical calculations. Here, four source water isomer data sets have been collected from past sampling processes on the different potential energy models, which are summarized in Table VI. Data Sources 1 and 2 were obtained from OSS2 while Sources 3 and 4 were obtained from TTM2.1-F. Here, the sparse data set archived from the past computational design of water isomers via B3LYP is then referred to the target domain of interest. In each domain, all water isomers are denoted as positive labeled data while the unstable water-molecule structures are assigned with negative labels.

TABLE VI
Water-Molecule Data Sets

Potential energy model	Domain	♯ positive data	♯ negative data	Positive Class Ratio(PCR)
OSS2	Source 1	1000	1000	0.5
	Source 2	1000	800	0.556
TTM2.1-F	Source 3	900	900	0.5
	Source 4	900	900	0.5
B3LYP	Target	200	100	0.667

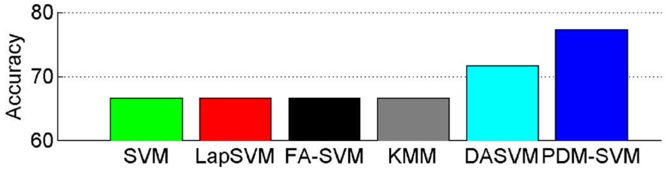
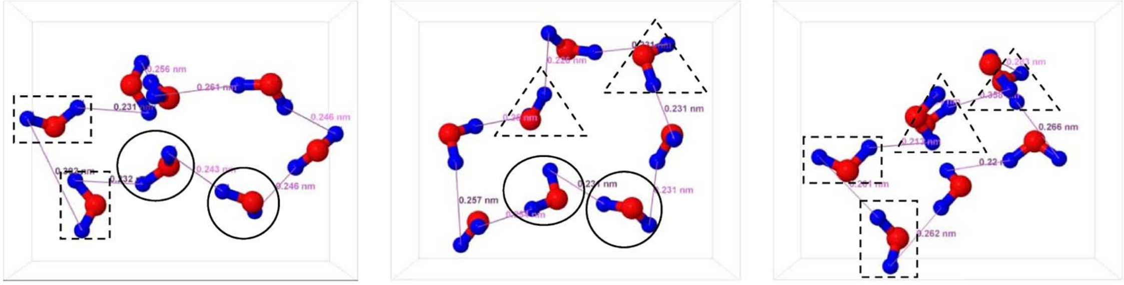


Fig. 8. Testing accuracies on binary-class water-molecule data set. Compared nonlinear problem on various types of SVM methods.

Aside from PDM-SVM, here, the traditional methods, SVM and LapSVM, and DA methods, KMM and DASVM, as described in Table II, are also used to address the water isomer prediction problem. In addition, FA on the SVM classifier (23), which is denoted in this study as FA-SVM, is further considered. In all the methods, the Gaussian kernel is employed.

Fig. 8 summarizes the accuracies obtained by PDM-SVM and all other algorithms considered for predicting the water isomers. Overall, PDM-SVM showcased superiority over all the methods considered, with rewarding performance of at least 5% accuracy improvements. Analysis shows that the improvements are attained due to class distribution differences between the source and target domains, as denoted in the last column of Table VI, and also the conflicting class labels among domains, which is caused by the low fidelity of the empirical models. In addition, PDM-SVM is shown to identify suitable source samples as depicted by the circle-enclosed substructures in Fig. 9(a) and (b) that would lead to positive transferability. The dotted-rectangle-enclosed substructures in Fig. 9(a) and the dotted-triangle-enclosed substructures in Fig. 9(b) denote the positive information transferred from source domains in the inference of the target B3LYP molecule of Fig. 9(c). This learning process allows the target structure to be inferred while avoiding source samples that conflict with the target domain.



(a) (b) (c)

Fig. 9. Example of eight water molecules in different domains. PDM-SVM uses OSS2 and TTM2.1 water isomers to infer a B3LYP as water isomer. The circle-enclosed substructures in subfigures (a) and (b) depict the structure similarities between the OSS2 and TTM2.1 samples. Similarly, the dotted-rectangle-enclosed substructures in subfigure (a) and the dotted-triangle-enclosed substructures in subfigure (b) denote the positive information transferred from source domains in the inference of the target B3LYP molecule of subfigure (c).

Eventually, PDM-SVM learns the predictive distribution of the target domain to select positive transferability data samples for enhanced prediction.

VII. Summary

In practice, the true predictive distributions of the source and the target domains often differ. When the predictive distributions among domains do not match well, DA algorithms that attempt to match the marginal distributions generally fail to function well due to the phenomenon of negative transfer. The causes of predictive distribution differences among related domains are mainly due to the differing class distributions and conflicting class labels among domains in specific regions of the vector space. In addition, the challenges pertaining to the differing predictive distributions among domains are known to increase in the multiclass context since more class distributions need to be considered.

To address the issues of predictive distribution differences among domains, we first present a criterion of positive transferability, which measures the similarity between two samples from different domains in terms of their predictive distributions. With this criterion, a *PDM* regularizer is proposed to enforce data that are similar according to the notion of positive transferability to have similar predictive outputs. To achieve this, an iterative construction of a k-nearest-neighbor graph that models the regions of relevant source labeled data with predictive distributions that maximally align with data in the target domain of interest has been presented. Finally, we incorporate our PDM regularizer into two common regularized risk frameworks. Namely, LR and SVM are considered as instantiations of the PDM to attain at PDM-LR and PDM-SVM classifiers that are robust to negative transfer caused by differing predictive distributions across domains. Extensive experimental study on the PDM framework has been shown to facilitate positive transferability with significant accuracy improvements attained on both binary and multiclass contexts, thus verifying the success of the proposed PDM regularizer in the identification of relevant data and transfer of useful knowledge across source domains.

References

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

[2] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proc. EMNLP*, 2006, pp. 120–128.

[3] H. Daumé, III, "Frustratingly easy domain adaptation," in *Proc. ACL*, 2007, pp. 256–263.

[4] S. J. Pan, I. Tsang, J. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," in *Proc. IJCAI*, 2009, pp. 1187–1192.

[5] L. Duan, I. W. Tsang, D. Xu, and T. S. Chua, "Domain adaptation from multiple sources via auxiliary classifiers," in *Proc. ICML*, 2009, pp. 289–296.

[6] W. Bian, D. Tao, and Y. Rui, "Cross-domain human action recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 298–307, Apr. 2012.

[7] X. Tian, D. Tao, and Y. Rui, "Sparse transfer learning for interactive video search reranking," *J. ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 8, no. 3, pp. 26–43, Jul. 2012.

[8] J. Yu, J. Cheng, and D. Tao, "Interactive cartoon reusing by transfer learning," *Signal Process.*, vol. 92, no. 9, pp. 2147–2158, Sep. 2012.

[9] L. Bruzzone and M. Marconcini, "Domain adaptation problems: A DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.

[10] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? Cross-domain collaborative filtering for sparsity reduction," in *Proc. IJCAI*, 2009, pp. 2052–2057.

[11] Z. Zhu, Y.-S. Ong, and J. M. Zurada, "Identification of full and partial class relevant genes," *IEEE/ACM Trans. Comput. Biol. Bioinfo.*, vol. 7, no. 2, pp. 263–277, Apr. 2010.

[12] W. Tjhi, K. K. Lee, T. Hung, Y. S. Ong, I. W. Tsang, V. Racine, and F. Bard, "Clustering-based methodology with minimal user supervision for displaying cell-phenotype signatures in image-based screening," in *Proc. IEEE BIBMW*, 2010, pp. 252–257.

[13] D. Lim, Y. Jin, Y.-S. Ong, and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 329–355, Jun. 2010.

[14] K. Crammer, M. Kearns, and J. Wortman, "Learning from multiple sources," *J. Mach. Learn. Res.*, vol. 9, pp. 1757–1774, Aug. 2008.

[15] P. Luo, F. Zhuang, H. Xiong, Y. Xiong, and Q. He, "Transfer learning from multiple source domains via consensus regularization," in *Proc. CIKM*, 2008, pp. 103–112.

[16] G. Schweikert, C. Widmer, B. Schölkopf, and G. Rätsch, "An empirical analysis of domain adaptation algorithm for genomic sequence analysis," in *Proc. NIPS*, 2009, pp. 1–8.

[17] M. M. Mahmud and S. R. Ray, "Transfer learning using Kolmogorov complexity: Basic theory and empirical evaluations," in *Proc. NIPS*, 2007, pp. 1–8.

[18] J. Huang, A. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proc. NIPS*, 2006, pp. 601–608.

[19] S. Bickel, C. Sawade, and T. Scheffer, "Transfer learning by distribution matching for targeted advertising," in *Proc. NIPS*, 2008, pp. 145–152.

[20] S. Bickel, M. Brückner, and T. Scheffer, "Discriminative learning under covariate shift," *J. Mach. Learn. Res.*, vol. 10, no. 9, pp. 2137–2155, Sep. 2009.

[21] A. Gretton, K. Borgwardt, M. Rasch, B. Scholkopf, and A. Smola, "A kernel method for the two-sample-problem," in *NIPS*, 2007, pp. 513–520.

[22] S.-I. Lee, V. Chatalbashev, D. Vickrey, and D. Koller, "Learning a meta-level prior for feature relevance from multiple related tasks," in *Proc. ICML*, 2007, pp. 489–496.

[23] S. Satpal and S. Sarawagi, "Domain adaptation of conditional probability models via feature subsetting," in *Proc. ECML/PKDD*, 2007, pp. 224–235.

[24] S. Si, D. Tao, and B. Geng, "Bregman divergence-based regularization for transfer subspace learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 929–942, Jul. 2010.

[25] S. Si, W. Liu, D. Tao, and K.-P. Chan, "Distribution calibration in Riemannian symmetric space," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 4, pp. 921–930, Aug. 2011.

[26] X. Zhu, "Cross-domain semi-supervised learning using feature formulation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 6, pp. 1627–1638, Dec. 2011.

[27] S. Si, D. Tao, and K.-P. Chan, "Evolutionary cross-domain discriminative Hessian eigenmaps," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1075–1086, Apr. 2010.

[28] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram, "Multi-task learning for classification with Dirichlet process priors," *J. Mach. Learn. Res.*, vol. 8, no. 1, pp. 35–63, Jan. 2007.

[29] J. Chen, J. Zhou, and J. Ye, "Integrating low-rank and group-sparse structures for robust multi-task learning," in *Proc. SIGKDD*, 2011, pp. 42–50.

[30] S. Ben-David and R. S. Borbely, "A notion of task relatedness yielding provable multiple-task learning guarantees," *Mach. Learn.*, vol. 73, no. 3, pp. 273–287, Dec. 2008.

[31] M. T. Rosenstein, Z. Marx, and L. P. Kaelbling, "To transfer or not to transfer," in *Proc. NIPS Workshop Inductive Transfer: 10 Years Later*, 2005, pp. 1–4.

[32] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[33] C.-W. Seah, I.-T. Tsang, and Y.-S. Ong, "Healing sample selection bias by source classifier selection," in *Proc. IEEE ICDM*, 2011, pp. 577–586.

[34] C.-W. Seah, I.-T. Tsang, Y.-S. Ong, and M. Qi, "Learning target predictive function without target labels," in *Proc. IEEE ICDM*, 2012, pp. 427–439.

[35] Y. Chen, G. Wang, and S. Dong, "Learning with progressive transductive support vector machine," in *Proc. IEEE ICDM*, 2002, pp. 67–74.

[36] M. C. Du Plessis and M. Sugiyama, "Semi-supervised learning of class balance under class-prior change by distribution matching," in *Proc. ICML*, 2012, pp. 823–830.

[37] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, Jan. 2002.

[38] C.-W. Seah, I. W. Tsang, Y.-S. Ong, and K.-K. Lee, "Predictive distribution matching SVM for multi-domain learning," in *Proc. ECML/PKDD*, 2010, pp. 231–247.

[39] F.-L. Huang, C.-J. Hsieh, K.-W. Chan, and C.-J. Lin, "Iterative scaling and coordinate descent methods for maximum entropy models," *J. Mach. Learn. Res.*, vol. 11, pp. 815–848, Feb. 2010.

[40] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Proc. Adv. Large Margin Classifiers*, 1999, pp. 61–74.

[41] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, no. 11, pp. 2399–2434, Nov. 2006.

[42] W. Bian and D. Tao, "Manifold regularization for SIR with rate root-n convergence," in *Proc. NIPS*, 2009, pp. 117–125.

[43] B. Geng, D. Tao, C. Xu, L. Yang, and X.-S. Hua, "Ensemble manifold regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1227–1233, Jun. 2012.

[44] S. Melacci and M. Belkin, "Laplacian support vector machines trained in the primal," *J. Mach. Learn. Res.*, vol. 12, no. 3, pp. 1149–1184, Mar. 2011.

[45] J. Jiang and C. Zhai, "A two-stage approach to domain adaptation for statistical classifiers," in *Proc. CIKM*, 2007, pp. 401–410.

[46] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification," in *Proc. ACL*, 2007, pp. 440–447.

[47] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. ICML*, 2007, pp. 193–200.

[48] J. Gao, W. Fan, J. Jiang, and J. Han, "Knowledge transfer via multiple model local structure mapping," in *Proc. ACM SIGKDD*, 2008, pp. 283–291.

[49] X. Ling, W. Dai, G. R. Xue, Q. Yang, and Y. Yu, "Spectral domain-transfer learning," in *Proc. KDD*, 2008, pp. 488–496.

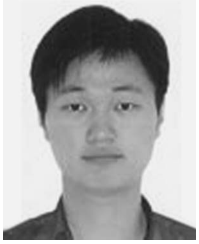
[50] H. Soh, Y.-S. Ong, Q. C. Nguyen, Q. H. Nguyen, M. Habibullah, T. Hung, and J.-L. Kuo, "Discovering unique, low-energy pure water isomers: Memetic exploration, optimization, and landscape analysis," *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 419–437, Jun. 2010.

[51] M. Mella, J.-L. Kuo, D. C. Clary, and M. L. Klein, "Nuclear quantum effects on the structure and energetics of (h2o)6h+," *Phys. Chem. Chem. Phys.*, vol. 7, no. 11, pp. 2324–2332, Jun. 2005.

[52] C. Lee, W. Yang, and R. G. Parr, "Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density," *Phys. Rev. B, Condens. Matter*, vol. 37, no. 2, pp. 785–789, Jan. 1988.

[53] L. Ojamae, "Potential models for simulations of the solvated proton in water," *J. Phys. Chem.*, vol. 109, no. 13, pp. 5547–5564, Oct. 1998.

[54] G. S. Fanourgakis and S. S. Xantheas, "The flexible, polarizable, thole-type interaction potential for water (TTM2-F) revisited," *J. Phys. Chem. A*, vol. 110, no. 11, pp. 4100–4106, Mar. 2006.



Chun-Wei Seah received the B.Eng. degree (first-class honors) in computer science from Nanyang Technological University (NTU), Singapore, in 2009, where he is currently working toward the Ph.D. degree in the field of machine learning in the School of Computer Engineering. He is also a student at the Center for Computational Intelligence, NTU.

His current research interests include transductive learning, transfer learning, and sentiment prediction.

Mr. Seah was a recipient of the Nanyang President's Graduate Scholarship in 2009.

Yew-Soon Ong received the B.S. and M.S. degrees in electrical and electronics engineering from Nanyang Technological University (NTU), Singapore, in 1998 and 1999, respectively, and the Ph.D. degree on artificial intelligence in complex design from the Computational Engineering and Design Center, University of Southampton, Southampton, U.K., in 2002.

He is currently an Associate Professor and the Director of the Center for Computational Intelligence at the School of Computer Engineering, NTU. His research interest in computational intelligence spans across memetic computing, evolutionary design, machine learning, agent-based systems, and cloud computing.

Dr. Ong is the Founding Technical Editor-in-Chief of the Memetic Computing Journal, the Chief Editor of the Springer book series on studies in adaptation, learning, and optimization, and an Associate Editor of the IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE, IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS—PART B, Soft Computing, Information Sciences, International Journal of System Sciences, and many others. He also chairs the IEEE Computational Intelligence Society Emergent Technology Technical Committee and has served as the Guest Editor of several journals.

Ivor W. Tsang received the Ph.D. degree in computer science from the Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2007.

He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. He is the Deputy Director of the Center for Computational Intelligence, NTU.

Dr. Tsang received the prestigious IEEE Transactions on Neural Networks Outstanding 2004 Paper Award in 2006 and the 2008 National Natural Science Award (Class II), China, in 2009. His coauthored papers also received the Best Student Paper Award at the 23rd IEEE Conference on Computer Vision and Pattern Recognition in 2010, the Best Paper Award at the 23rd IEEE International Conference on Tools with Artificial Intelligence in 2011, the 2011 Best Student Paper Award from Pattern Recognition and Machine Intelligence Association, Singapore, in 2012, and the Best Paper Award from the IEEE Hong Kong Chapter of Signal Processing Postgraduate Forum in 2006. He was also conferred with the Microsoft Fellowship in 2005.