On Handling Negative Transfer and Imbalanced Distributions in Multiple Source Transfer Learning

Liang Ge, Jing Gao, Hung Ngo, Kang Li, Aidong Zhang

Computer Science and Engineering Department, State University of New York at Buffalo
Buffalo, 14260, U.S.A.
{liangge, jing, hungngo, kli22, azhang}@buffalo.edu

Abstract

Transfer learning has benefited many real-world applications where labeled data are abundant in source domains but scarce in the target domain. As there are usually multiple relevant domains where knowledge can be transferred, *multiple source transfer learning* (MSTL) has recently attracted much attention. However, we are facing two major challenges when applying MSTL. First, without knowledge about the difference between source and target domains, *negative transfer* occurs when knowledge is transferred from highly irrelevant sources. Second, existence of *imbalanced distributions* in classes, where examples in one class dominate, can lead to improper judgement on the source domains' relevance to the target task. Since existing MSTL methods are usually designed to transfer from relevant sources with balanced distributions, they will fail in applications where these two challenges persist. In this paper, we propose a novel two-phase framework to effectively transfer knowledge from multiple sources even when there exist irrelevant sources and imbalanced class distributions. First, an effective Supervised Local Weight (SLW) scheme is proposed to assign a proper weight to each source domain's classifier based on its ability of predicting accurately on each local region of the target domain. The second phase then learns a classifier for the target domain by solving an optimization problem which concerns both training error minimization and consistency with weighted predictions gained from source domains. A theoretical analysis shows that as the number of source domains increases, the probability that the proposed approach has an error greater than a bound is becoming exponentially small. Extensive experiments on disease prediction, spam filtering and intrusion detection data sets demonstrate the significant improvement in classification performance gained by the proposed method over existing MSTL approaches.

1 Introduction

Transfer learning refers to the scenario that given a learning task on a target domain, knowledge is extracted from one or several related domains (source domains) to help the learning task on the target domain. It adapts knowledge from source domains to the target domain by considering unlabeled information on the target domain. Such knowledge transfer is possible when the target domain and source domains have the same set of categories or class labels. The process of transfer learning is deeply rooted from our individual experience: We always borrow knowledge from other areas to help learning in one area. Based on this simple philosophy, many methods have been proposed on transfer learning [10, 8, 2, 3, 9, 4, 6, 7] and many successful applications including document classification, WiFi localization, and sentiment classification [12] demonstrate the power of transfer learning.

There are usually multiple source domains where knowledge can be transferred, and how to take advantage of the different predictive powers of the source domains motivates the study of *Multiple Source Transfer Learning* (MSTL) [10, 8, 2, 3, 9, 4]. MSTL is especially useful when we have enough source domains who share the same task with the target domain, however, two major challenges prevent us to successfully apply MSTL methods to many applications due to the existence of irrelevant sources and imbalanced class distributions. Some example applications are discussed as follows. We could transfer knowledge from multiple other patients to help predict on the target patient in disease diagnosis; other users' information could be used to help build a better classifier for the target user in spam filtering; and anomaly detection tools designed for existing intrusions can be adapted to identify a new attack to computer networks. In all these applications, there exist source domains that are highly irrelevant to the target domain. However, which source is irrelevant is usually unknown, and incorporating such irrelevant sources will hurt the prediction performance of MSTL algorithms. Furthermore, the number of positive examples (disease, spam, intrusions) is much smaller than that of negative examples, resulting in difficulties of properly evaluating and weighing source domains according to their predictive behavior on the target domain. To illustrate the two challenges and how they affect existing MSTL methods, we focus on Cardiac Arrhythmia Detection (CAD) problem in the following discussions.

Cardiac Arrhythmia refers to a range of conditions arising from abnormal activities in the heart. Cardiac arrhythmia is commonly examined based on Electrocardiography data (ECG). The task of cardiac arrhythmia detection is to build a classifier to predict the labels (normal or abnormal) of the test samples, given a patient's test ECG data samples and a small portion of

labeled ECG samples. In the CAD problem, we notice that besides data of the target patient, ECG data from many other patients who suffered arrhythmia are also collected. Therefore, regarding each patient's ECG and associated labels as a source domain, this problem can be casted as a multiple source transfer learning problem, which inspire some new challenges that previous work on transfer learning seldom addresses.

Challenge I: Negative transfer. Negative transfer [12, 14, 13] refers to the phenomenon that, instead of improving performance, transfer learning from other domains degrades the performance on the target domain. Most previous work treats knowledge from every source domain as a valuable contribution to the task on the target domain. However, in the cardiac arrhythmia detection task, when we are trying to transfer knowledge from multiple other patients, it is over optimistic to believe that all source domains will contribute. In fact, it is highly probable that some of the source patients have drastically different distribution in their ECG data than the target patient, which indicates that transferring from this kind of sources could harm the learning on the target patient. We call such sources as *irrelevant sources*. Given multiple source domains, in the worst case, the majority of the source domains could be irrelevant. We believe that the occurrence of many *irrelevant sources* will trigger *negative transfer* if they are not handled properly. Despite the fact that how to avoid negative transfer is a very important issue, little research has been done on this perspective.

Challenge II: Imbalanced distributions. Imbalanced distributions in classes mean that one of the classes constitutes only a very small percentage of the data set. For the patients suffering cardiac arrhythmia, normal heart beats outnumber arrhythmia a lot. In such cases, accuracy is not a good evaluation measure of classification performance, but many existing transfer learning methods prefer to extract knowledge from the sources that have high accuracy. In the circumstances of *imbalanced distributions*, we can easily design a prediction, e.g., predicting every sample to be normal heart beat, that achieves extremely high accuracy. However, for this particular task, source patient that is good at predicting normal heart beats is hardly useful, because naturally we care much more about the arrhythmia cases. Although imbalanced distributions has been well studied in traditional classification [16], yet how to handle *imbalanced distributions* in source domains is a topic seldom discussed in the literature of transfer learning.

In light of these challenges, we propose a two-phase multiple source transfer framework, which can effectively downgrade the contributions of irrelevant source domains and properly evaluate the importance of source domains even when the class distributions are imbalanced. In the first phase, a novel Supervised Local Weight (SLW) scheme is proposed to assign an accurate local weight to each source domain on each region of the target domain. By utilizing label propagation from the small amount of labeled data in the target domain, the proposed scheme successfully identifies irrelevant sources for each region and alleviate the effect of imbalanced distributions on source domain weight assignment. To further ensure that reasonable performance is achieved even when all the source domains are irrelevant, we develop the second phase to learn a classifier by solving an optimization problem involving both source domain transferring and target domain classification. Importantly, a theoretical analysis is presented to show the error bound of the proposed method. As the number of source domains increases, the probability that the proposed approach has an error greater than a bound is becoming exponentially small. We compare the proposed approach with state-of-the-art MTSL methods on cardiac arrhythmia detection, email spam filtering and network intrusion detection data sets, and the results demonstrate that the proposed method gain significant improvement on the classification performance.

The major contributions of this paper are:

- We explore new perspectives in transfer learning where *negative transfer* and *imbalanced distributions* pose unique challenges to the transfer learning community.
- We propose a two-phase transfer learning framework that properly addresses these two important challenges.
- We provide a theoretical analysis to show the error bound of the proposed approach.
- Extensive experimental results on three applications demonstrates that the proposed approach outperforms existing transfer learning methods with improvement up to 34.6%.

2 Problem Setting and Challenges

Assume there are k source domains. The s-th source domain is characterized by a data set $D^s = (x_i^s, y_i^s)|_{i=1}^{n_s}$, where x_i^s is the feature vector, y_i^s is the corresponding label, and n_s is the total number of samples for source domain s. The target domain has a few labeled data $D_l^T = (x_i^T, y_i^T)|_{i=1}^{n_l}$ and plenty of unlabeled data $D_u^T = x_i^T|_{i=n_l+1}^{n_l+n_u}$ where n_l and n_u are the number of labeled and unlabeled target domain samples, respectively. The goal is to develop a target classifier f^T that can predict the label of the test data in the target domain, using knowledge extracted from source domains and a few target labeled data.

Due to the imbalanced distributions, *accuracy* is not that meaningful in evaluating classification performance. Therefore, we calculate *Receiver Operating*

Characteristics (ROC) curve and assess the quality of the ROC curve by *Area Under the Curve (AUC)*. In the following, we will discuss the two challenges in detail, and explain why existing multiple source transfer learning techniques fail in these circumstances. Again, we use CAD problem as an illustrating example, but the discussions can be easily generalized to other applications whose data possess these two properties.

2.1 Negative Transfer The data sets in the CAD problem come from MIT-BIH database [11]. We randomly chose 13 patients belonging to two classes: arrhythmia and normal heart beats. Given multiple source patients, some of them are similar to the target patient but some of them are not. To show this, we mandate that one patient (ID 201) is the target patient while all the other patients are source patients. We train a classifier (in this case, Logistic Regression) from each source patient's data and use the classifier to predict the test set of patient 201's data. Table 1 shows the prediction results of the classifier trained from each source patient on patient 201.

Table 1: Negative Transfer Examples

Relevant Sources			Irrelevant Sources		
ID	AUC	Accuracy	ID	AUC	Accuracy
121	0.701	85%	100	0.619	44%
202	0.807	88%	101	0.619	55%
210	0.739	84%	103	0.517	49%
215	0.673	76%	105	0.525	53%
230	0.643	75%	109	0.597	47%
232	0.689	78%	115	0.601	52%

It is straightforward to see that the 12 source patients include *relevant sources*: 121, 202, 210, 215, 230, 232 and *irrelevant sources*: 100, 101, 103, 105, 109, 115. Without knowledge of the target test data, it is impossible to know which source is relevant. If we choose source patients from these 12 patients, the collection is very likely to be a combination of both, and in the worst case, the majority are *irrelevant sources*.

Table 2: Experimental Setup

Exp.#	Source Patient IDs	Comments
1	[121, 202, 210, 215, 230, 232]	All Relevant
2	[100, 101, 103, 105, 109, 115]	All Irrelevant
3	[121, 101, 103, 105, 109, 115]	Majority Irrelevant
4	[202, 101, 103, 105, 109, 115]	Majority Irrelevant
5	[103, 202, 210, 215, 230, 232]	Majority Relevant
6	[105, 202, 210, 215, 230, 232]	Majority Relevant
7	[121, 202, 210, 101, 103, 105]	Half Relevant
8	[215, 230, 232, 101, 103, 105]	Half Relevant
9	All Patients	All Patients

The circumstances are quite different from the various applications that transfer learning is applied [12], where all (or most of) the sources are closely related to the target. Therefore, previous transfer learning methods may induce *negative transfer* given multiple *irrelevant sources*. To show this, we design the following experiments as described in Table 2. There are 9 experiments in which each experiment takes different source patients.

We have two natural baseline methods: 1) Unweighted average of multiple sources' predictions (**Unweighted**); and 2) best single prediction among all source patients (**Best**). A good transfer learning method should yield better results than the two baseline methods. Two recent MSTL methods CRC [10] and GCM [8] are compared here. Both methods rely on the maximization of the consensus among sources. CRC [10] assumes that all sources are closely related to the target while GCM [8] implicitly assumes that the majority of sources are similar to the target.

Table 3: Results Showing Negative Transfer

Exp.#	Unweighted	Best	CRC [10]	GCM [8]
1	0.918	0.807	0.666	0.781
2	0.536	0.626	0.611	0.521
3	0.732	0.701	0.511	0.528
4	0.764	0.807	0.522	0.520
5	0.857	0.807	0.620	0.739
6	0.898	0.807	0.617	0.642
7	0.859	0.807	0.576	0.733
8	0.650	0.689	0.569	0.689
9	0.882	0.807	0.600	0.699

As shown in Table 3, in experiments 3 and 4 where the majority of the sources are *irrelevant*, CRC and GCM can't beat the best prediction among all sources and their performance are close to random guessing (AUC=0.5). Therefore, multiple sources, if not handled properly, will do more harm than good, and *negative transfer* hurts performance. In other experiments, CRC and GCM didn't work well mainly because of the *imbalanced distributions* in source patient data (as will be discussed in Section 2.2). The experiments confirm our speculation that *negative transfer* will cause troubles and how to properly handle *irrelevant sources* and avoid *negative transfer* needs to be addressed.

2.2 Imbalanced Distributions Most patients suffering cardiac arrhythmia have much more normal heart beats than arrhythmia. Such imbalanced distributions in source patients are likely to yield classifiers that are good at predicting normal heart beats and its overall accuracy is high due to the large volume of normal heart beats. Most existing transfer learning methods assign high weights to such sources as their weighting scheme is purely based on accuracy. However, predicting arrhythmia accurately is much more important, and transfer learning methods should take this factor into account.

Two recently proposed MSTL methods MDA [2] and LWE [9] both weigh each source domain based on the smoothness assumption that a source will gain a high weight if its predictions are smooth among data samples that are close in the feature space. MDA [2] computes a single weight for each source while LWE [9] computes a weight of each source on each sample. Both methods are vulnerable to imbalanced distributions. For example, suppose we have a classifier which predicts every sample

Table 4: Summary of Related Work

Alg.	Negative Transfer	Imbal-anced	Methodology
CRC [10]	no	no	maximization of consensus
GCM [8]	no	no	graph-based consensus
MDA [2]	yes	no	global weight smoothness assumption
LWE [9]	yes	no	local weight smoothness assumption
SLW	**yes**	**yes**	**local weight supervised manifold**

to be a normal heart beat. Such a classifier will be assigned the highest weight by MDA and LWE because its predictions are smooth everywhere.

Let's take experiment 4 as an example. It is composed of source patients [202, 101, 103, 105, 109, 115]. The weights computed by MDA for patients 202 and 105 are 0.234 and 0.662, respectively. Different from MDA, LWE produces a weight distribution for each source. In this experiment, the weight distributions for patients 105 and 202 are almost the same. From Table 1, we know that patient 105 is an *irrelevant source* while patient 202 is a *relevant source*, yet patient 105 gains a higher weight by MDA than patient 202, and the weights of patient 105 and 202 by LWE are almost the same. As can be seen, both MDA and LWE assign high weights to patient 105 because it has smooth predictions, but such predictions make little use in the CAD problem. For the previous two MSTL methods CRC [10] and GCM [8], they are also vulnerable to imbalanced distributions. Given multiple source patients, it is likely to have many classifiers that are good at predicting normal heart beats. CRC and GCM will produce results that are consistent with the majority of predictions from sources, thus incurring less satisfactory performance. Table 4 summarizes the key features of related worked discussed in this section. From the above analysis, we believe that it is crucial to develop a solution that can handle both *negative transfer* and *imbalanced distribution*, which is achieved by a novel approach called SLW. The details will be elaborated in the following section.

3 Methodology

In this section, a two-phase approach, consisting of a Supervised Local Weight (SLW) scheme (Section 3.1) and a combined classifier learning step (Section 3.2), is presented to handle *negative transfer* and *imbalanced distributions* in multiple source transfer learning.

3.1 Supervised Local Weight Scheme We want to assign a weight to each source domain which represents its predictive power on the target domain. The weight should be local because each source may be good in some regions but bad in some other regions. To achieve this, we first propose a Supervised Local Weight (SLW) scheme based on the following assumption:

- *Supervised Manifold Assumption*: If predictions from a particular source domain are smooth and consistent with true labels on a manifold, the source domain will be assigned a high weight on this manifold.

Let's use a toy example to illustrate the *Supervised Manifold Assumption*. Figure 1 shows the target label distribution while Figure 2 shows the predictions from a particular source domain s. In the figures, square and triangle represent two different classes. As we can see, the target data contain four manifolds/clusters. The predictions from the source domain are smooth in manifolds $R1$ and $R4$ and also consistent with true labels. Therefore we will assign high weights to domain s for samples in manifolds $R1$ and $R4$. In Region $R2$, the predictions are not smooth, a low weight will be assigned to domain s in this manifold. An important observation arises that the predictions are smooth in Region $R3$ but they are opposite to the true labels in this manifold. Thus, we should assign a low weight to domain s in Region $R3$. This assumption considers local weights for each source, which is different from the global weight assumption held by CRC [10], MDA [2] and GCM [8]. Although LWE [9] calculates local weights, it only considers unsupervised manifolds, which may lead to wrong predictions in negative transfer and imbalanced situations.

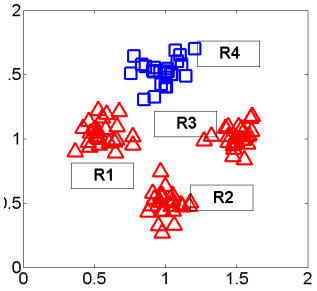

Figure 1: Target Domain

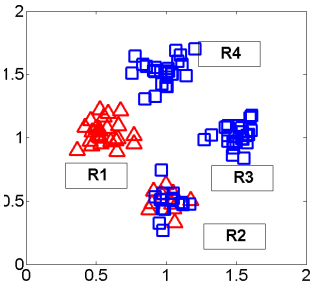

Figure 2: Source Domain

The proposed method is as follows. We first use the spectral clustering [17] algorithm to partition the target data into c clusters, which minimizes:

$$\min_{\{f_i\}_{i=1}^{n^T}} \frac{1}{2} \sum_{i,j=1}^{n^T} W_{ij} \left(\frac{f_i}{\sqrt{D_{ii}}} - \frac{f_j}{\sqrt{D_{jj}}} \right)^2, \tag{3.1}$$

where W_{ij} is the similarity between two samples in the target data, D is a diagonal matrix with its (i,i) entry equal to the sum of the i-th row of W and f_i is the cluster id. Secondly, we propose to approximate groundtruth of the target labels by label propagation [18]. Given a small training set with labels in the target

domain, we can obtain the approximated label of each sample by minimizing the following:

(3.2)
$$\min_{\{F_i\}_{i=1}^{n^T}} \frac{1}{2}\left(\sum_{i,j=1}^{n^T} W_{ij}||\frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}}||^2 + \mu\sum_{i=1}^{n^T}||F_i - H_i||^2\right),$$

where F_i is a $1 \times c$ vector indicating the class membership of a data object on the target domain to be computed. H_i denotes the initial label where $H_{ij} = 1$ if sample i is labeled class j. H_i represents the training data on the target domain. The intuition behind Eq. 3.2 is to propagate the known labels based on the smoothness assumption, which encourages label smoothness over all data points in that similar examples tend to have similar labels. Note that Eq. 3.2 has a closed-form solution as follows:

$$F^* = (I - \delta L)^{-1}H, \tag{3.3}$$

where $\delta = \frac{1}{1+\mu}$ and$L = D^{-1/2}WD^{-1/2}$.

For a given manifold/cluster C_i and a source domain's predictions p, we have p_{C_i} denote the predictions of p on the manifold C_i, and F_{C_i} denote the approximated labels on the manifold C_i drawn from Eq. 3.2. Given the approximated labels, the local weight w_{p,C_i} is defined as:

$$w_{p,C_i} = \frac{\sum_{v1\in p_{C_i}}\sum_{v2\in F_{C_i}} 1\{v1=v2\}}{|C_i|}, \tag{3.4}$$

where w_{p,C_i} denotes the percentage of label matches between predictions made by a source domain and those made by label propagation.

The only difference between Eq. 3.1 and Eq. 3.2 is the regularization term on the small amount of labels from target domain. Therefore, we expect the approximated labels from Eq. 3.2 are consistent with clusters from Eq. 3.1. Eq. 3.4 assigns weights to each domain based on the accuracy of the alignment in each local region in the target domain. Based on Eq. 3.4, w_{p,C_i} is high if the predictions are smooth on the manifold C_i and consistent with approximated labels. w_{p,C_i} is low if the predictions are not smooth or not consistent with approximated labels on the manifold C_i. In this way, we implement *Supervised Manifold Assumption* by computing w_{p,C_i}. The pseudo code in Algorithm 1 summarizes how to compute the supervised local weights.

3.2 Learning the Target Classifier The second phase considers both the weighted predictions from all sources and the target training data to learn a classifier f^T. It ensures that even in the worst case where most of source domains are irrelevant, the performance of the proposed method is no worse than the prediction using target training data alone. In addition, we can generate a "single" classifier which has the behavior similar to the ensemble classifier, which leads to easy interpretation and usage on future predictions.

Algorithm 1 Supervised Local Weight Computation
Input: predictions from each source domain $p_1, ..., p_k$, target training set D_l^T, target testing set D_u^T, number of clusters c, the parameter μ
Output: a $n_u \times k$ weight matrix P_w

1: Partition the testing set D_u^T into c clusters based on Eq. 3.1
2: Compute approximated labels F^* using Eq. 3.3
3: For each cluster C_i, given source prediction p_k, compute the predictions on C_i from p_k: p_{C_i}
4: For each cluster C_i, given approximated labels F^*, compute approximated labels on C_i: F_{C_i}
5: Compute each entry of weight matrix P_w using Eq. 3.4
6: Normalize each row of P_w so that their sum is 1

Given the supervised local weights, the predicted label for the i-th sample of the target domain combining multiple sources is

$$\hat{h}_i = \sum_{s=1}^{k} w_{s,i} f_i^s, \tag{3.5}$$

where $w_{s,i}$ is the local weight of source s on the i-th sample and f_i^s is the predicted label from source s. Then we combine the information of both weighted predictions from source domains and the training data of the target domain by learning a classifier f^T to minimize the following objective function:

(3.6)
$$\min_{f^T} \frac{1}{n_l}\sum_{i=1}^{n_l}(f_i^T - h_i^T)^2 + \gamma||f^T||_K^2 + \frac{\beta}{2}\sum_{j=n_l+1}^{n_T}||f_j^T - \hat{h}_j||^2.$$

The implications of Eq. 3.6 are three-fold: 1) We want to minimize the training error; 2) we want the test data close to the predicted labels from source domains and β is the confidence of such belief; and 3) we want to control the complexity of f^T which is governed by γ.

We mandate that f^T comes from a Reproducing Kernel Hilbert Space that is induced by a Kernel function K. By the Representer theorem [15], we can find an optimal solution for the objective function in Eq. 3.6, which is a linear expansion of the kernel function K as follows:

$$f^T(x) = \sum_{i=1}^{n_l+n_u} \alpha_i K(x_i, x). \tag{3.7}$$

Taking Eq. 3.7 back into Eq. 3.6, we obtain the optimal α^* by solving the optimization problem in Eq. 3.6 and we have the following solution for α^*:

$$\alpha^* = (JK + \gamma(n_l + \beta n_u)I)^{-1}JH. \tag{3.8}$$

H is the label vector where $H_i = h_i$ if sample i belongs to training set, and $H_i = \hat{h}_i$ if sample i belongs to testing set. J is a diagonal matrix of size $(n_l + n_u) \times (n_l + n_u)$ where J=$diag$(1,...,1,β,...,β) with the first n_l diagonal entries as 1 and the rest as β.

3.3 Summary Since the Supervised Local Weight scheme plays a more important role in the two-phase framework, we name the overall approach as **SLW**. SLW is able to handle the two challenges i.e., negative transfer and imbalanced distribution for multiple source transfer learning. The formal error analysis can be found in the next section.

- *Negative Transfer*: SLW is able to handle negative transfer in that it prevents low quality predictions from irrelevant sources to have a high weight. Even in the worst case scenario when all the sources are irrelevant, SLW minimizes a combined loss function involving both target training sets and weighted predictions of source domains.
- *Imbalanced Distribution*: SLW handles imbalanced distribution in source domains in that it prevents predictions which have high accuracy to have high weights in the manifold representing the minority class. If predictions are smooth but opposite to the true labels in a manifold, it will be assigned low weights in this manifold.

4 Error Bound Analysis

In this section, we present a theoretical analysis on the performance of the proposed approach. In Section 4.1, we derive the equation for the error made by SLW. In Section 4.2, we present the error bound.

4.1 Error Formulation Here we focus on binary classification problems. Let Y be the random variable that represents the class label of each sample in the target domain's data. Let $Y = 1$ denote a positive case and $Y = 0$ denote a negative case.

Now we formulate the *Supervised Manifold Assumption*. Suppose there are two natural clusters in the target set and each of them is associated with a label $Y_c = 1$ or 0. We assume that cluster labels align with class labels, i.e., $Y_c = 1$ indicates a positive case and $Y_c = 0$ indicates a negative case. Y_c is in fact an approximated label on each target object. Suppose $P(Y_c = 1) = t$, i.e., the probability that a data sample falls into the cluster denoting positive cases. Similarly we have $P(Y_c = 0) = 1 - t$. However, we don't assume that objects in each cluster always belong to the same class. Instead, we assume that the chance of being a positive case in the target object given a negative output by the approximated label (false negative) is $P(Y = 1|Y_c = 0) = q$, and consequently $P(Y = 0|Y_c = 0) = 1 - q$. Similarly, we define $P(Y = 0|Y_c = 1) = p$ (false positive), and $P(Y = 1|Y_c = 1) = 1 - p$.

Next we link classification models derived from source domains to the target domain. Suppose there are k classifiers learnt from k source domains, and let Y_s denote the prediction made by the s-th model on a target object. Again, $Y_s = 1$ indicates that the s-th model predicts positive and $Y_s = 0$ denotes a negative case. Suppose each model follows the *Supervised Manifold Assumption* in general but with a probability of flipping the cluster label to the other class. The chance of the s-th model making a different prediction from the approximated label is $P(Y_s = 0|Y_c = 1) = p_s$, and $P(Y_s = 1|Y_c = 0) = q_s$, respectively.

In this analysis, we focus on the first phase of the proposed method. SLW combines the output of k classification models trained from the data of source domains using weights derived from the manifold structure of the target domain data. Let Y_m denote the prediction made by SLW. The following lemma presents the probability of making false negative error ($P(Y_m = 0|Y = 1)$) based on t, p, q, and $\{p_s, q_s\}_{s=1}^k$. Note that all these variables are probabilities, and thus they are all between 0 and 1.

LEMMA 4.1.

$$P(Y_m = 0|Y = 1) = \frac{a\sum_{s=1}^k (1-q_s)^2 + b\sum_{s=1}^k p_s(1-p_s)}{a\sum_{s=1}^k (1-q_s) + b\sum_{s=1}^k (1-p_s)},$$

where $a = (1-t)q$, $b = t(1-p)$.

Proof. Let's first compute $P(Y_m = 0|Y_c = 1)$. Since SLW takes a weighted combination of base model output Y_s, we have $P(Y_m = 0|Y_c = 1) = \sum_{s=1}^k P(Y_s = 0|Y_c = 1)P(M_s|Y_c = 1)$ where $P(M_s|Y_c = 1)$ indicates the weight assigned to the s-th model. In SLW, a model has a higher weight if its prediction aligns with the manifold and a lower weight otherwise. When $Y_c = 1$, i.e., the cluster manifold indicates a positive instance, we can use $P(Y_s = 1|Y_c = 1)$ to simulate the model weight $P(M_s|Y_c = 1)$. Therefore, $P(Y_m = 0|Y_c = 1) = \sum_{s=1}^k P(Y_s = 0|Y_c = 1)P(Y_s = 1|Y_c = 1) = \sum_{s=1}^k p_s(1-p_s)$. Similarly, we can derive that $P(Y_m = 0|Y_c = 0) = \sum_{s=1}^k (1-q_s)^2$.

We now show the probability of having $Y = 1$ and $Y_c = 1$ but $Y_m = 0$, i.e., the chance of SLW making a false negative error when both approximated label and the true label are positive. Due to independency assumption across models and true class labels, we have

$$\begin{aligned}
&P(Y_m = 0, Y = 1, Y_c = 1) \\
&= P(Y_m = 0|Y_c = 1)P(Y = 1|Y_c = 1)P(Y_c = 1) \\
&= t(1-p)\sum_{s=1}^k p_s(1-p_s).
\end{aligned}$$

Following the same procedure, we have $P(Y_m = 0, Y = 1, Y_c = 0) = (1-t)q\sum_{s=1}^k (1-q_s)^2$. Summing up these two probabilities, we have

$$\begin{aligned}
&P(Y_m = 0, Y = 1) = \\
&(1-t)q\sum_{s=1}^k (1-q_s)^2 + t(1-p)\sum_{s=1}^k p_s(1-p_s).
\end{aligned}$$

We can also get $P(Y_m = 1, Y = 1) = (1-t)q\sum_{s=1}^{k} q_s(1-q_s) + t(1-p)\sum_{s=1}^{k}(1-p_s)^2$. Based on total probability and Bayes theorem, we can derive the probability of SLW predicting wrong on positive instances $P(Y_m = 0|Y = 1)$ as shown in Lemma 4.1.

The false positive error of SLW $P(Y_m = 1|Y = 0)$ can be derived in a similar way. False negative error is more critical in the imbalanced classification problem, and thus we focus on the error bound analysis of false negative error.

4.2 Error Bound We have the following theorem on the error bound.

THEOREM 4.1. *Suppose p_s and q_s are i.i.d and follow uniform distribution $U(0,1)$, let $\bar{p} \geq \frac{2a+b}{3a+3b}$ and $\bar{\mu} = (1/3 - \bar{p}/2)a + (1/6 - \bar{p}/2)b$, we have*

$$Prob\left[P(Y_m = 0|Y = 1) \geq \bar{p}\right] \leq \exp\left(\frac{-2k\bar{\mu}^2}{C}\right),$$

where a and b are defined in Lemma 4.1, k is the number of source domains and C is a constant.

Proof. To simplify the representation, let $x_s = (1 - q_s)$ and $y_s = (1 - p_s)$, then we have

$$\begin{aligned} & P(Y_m = 0|Y = 1) \geq \bar{p} \\ \Leftrightarrow & \sum_{s=1}^{k}(ax_s^2 + b(1-y_s)y_s - a\bar{p}x_s - b\bar{p}y_s) \geq 0 \\ \Leftrightarrow & \sum_{s=1}^{k} Z_s \geq 0, \end{aligned}$$

$Z_s = ax_s^2 + b(1-y_s)y_s - a\bar{p}x_s - b\bar{p}y_s$ is a random variable.

Since p_s and q_s follow uniform distribution $U(0,1)$, we have x_s and y_s also follow uniform distribution $U(0,1)$. The expectation of Z_s is as follows:

$$\begin{aligned} E[Z_s] &= E[ax_s^2 + b(1-y_s)y_s - a\bar{p}x_s - b\bar{p}y_s] \\ &= a/3 + b/6 - a\bar{p}/2 - b\bar{p}/2 = \bar{\mu}. \end{aligned}$$

Let's mandate $\bar{p} \geq \frac{2a+b}{3a+3b}$, so that $\bar{\mu} \leq 0$. Since x_s, y_s, a and b are bounded in [0,1], it is easy to see that Z_s is also bounded. Let Z_s be bounded by $[m, n]$. We can set $m = 0$, $n = 2$.

The Hoeffding Inequality [5] shows that when $t \geq 0$:

$$Prob\left[\sum_{s=1}^{k} Z_s - E\left[\sum_{s=1}^{k} Z_s\right] \geq t\right] \leq \exp\left(\frac{-2t^2}{k(m-n)^2}\right).$$

Now let $t = -k\bar{\mu}$ and $C = (m-n)^2$, and we have:

$$Prob\left[\sum_{s=1}^{k} Z_s \geq 0\right] \leq exp\left(\frac{-2k\bar{\mu}^2}{C}\right).$$

That completes the proof.

Table 5: Performance Comparison (AUC)

CAD Prediction Data Set

Target. #	CRC	GCM	MDA	LWE	DAM	LP	SLW
100	0.666	0.777	0.760	0.722	0.925	0.959	**0.975**
101	0.611	0.779	0.742	0.423	0.753	0.802	**0.820**
103	0.511	0.626	0.478	0.543	0.648	0.683	**0.920**
105	0.522	0.654	0.714	0.718	0.725	0.617	**0.731**
109	0.620	0.739	0.700	0.753	0.879	0.837	**0.964**
115	0.576	0.679	0.654	0.720	**0.746**	0.503	0.713
121	0.534	0.610	0.655	0.492	0.572	0.526	**0.710**
201	0.600	0.699	0.843	0.854	0.894	0.892	**0.945**
202	0.600	0.715	0.818	0.795	0.847	0.675	**0.881**
210	0.617	0.699	0.830	0.819	0.899	0.828	**0.947**
215	0.620	0.760	0.537	0.632	0.544	0.869	**0.919**
230	0.614	0.679	0.334	0.610	0.674	0.824	**0.859**
232	0.652	0.771	0.724	0.948	0.855	0.954	**0.978**

Email Spam Filtering Data Set

User #	CRC	GCM	MDA	LWE	DAM	LP	SLW
U00	0.689	0.517	0.760	0.517	0.915	0.858	**0.939**
U01	0.789	0.837	0.535	0.905	0.853	0.818	**0.936**
U02	0.711	0.803	0.851	0.948	0.848	0.710	**0.954**
U03	0.722	0.916	0.908	0.804	0.827	0.827	**0.973**
U04	0.820	0.853	0.836	0.808	0.779	0.739	**0.866**
U05	0.786	0.806	0.702	0.839	0.776	0.735	**0.842**
U06	0.734	0.772	0.781	0.828	0.672	0.753	**0.828**
U07	0.800	0.864	0.914	0.907	0.827	0.848	**0.941**
U08	0.817	0.866	0.873	0.906	0.799	0.859	**0.919**
U09	0.824	0.872	0.850	0.927	0.744	0.856	**0.949**
U10	0.814	0.643	0.874	0.613	0.774	0.706	**0.872**
U11	0.752	0.751	0.860	0.889	0.825	0.751	**0.915**
U12	0.552	0.884	0.857	0.886	0.755	0.867	**0.909**
U13	0.762	0.573	0.752	0.678	0.845	0.715	**0.921**
U14	0.722	0.735	0.847	0.861	0.815	0.714	**0.877**

Intrusion Detection Data Set

Task #	CRC	GCM	MDA	LWE	DAM	LP	SLW
R2L	0.675	0.710	0.834	0.889	0.943	0.903	**0.983**
U2R	0.620	0.735	0.802	0.845	**0.873**	0.741	0.866
PROBE	0.631	0.703	0.851	0.928	0.948	0.803	**0.986**

Theorem 4.1 shows that given an error bound $\bar{p}$, the probability that SLW has an error greater than a bound will be exponentially decreased when the number of source domains increases. The error bound $\bar{p}$ is closely related to the probability of errors made by each source. We mandate that p_s and q_s follow i.i.d and uniform distribution because we don't have any prior knowledge about the source domains. The experimental evaluations show that SLW is able to reach a lower error bound given multiple source domains.

5 Experiments

In this part, we demonstrate the effectiveness of the proposed approach **SLW**. The algorithms are evaluated on three application domains and compared with six baseline methods.

5.1 Data sets We conduct experiments on three real-life data sets introduced as follows:

Cardiac Arrhythmia Detection. The ECG data sets in the CAD problem are from MIT-BIH database [11]. We randomly pick 13 patients' ECG data (time-series). Each patient's data consists of around 1008 to 1416 samples of 39 dimensional feature vectors, belonging to two classes: arrhythmia and normal heart beats. When learning on one patient, we transfer knowledge from all the other patients.

Spam Email Filtering. The email spam data set was

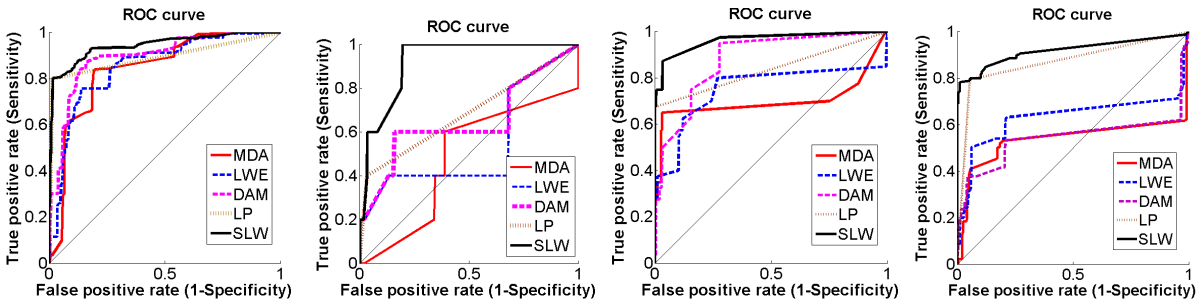

Figure 3: ROC curves for Patient 201, 103, 109 and 215

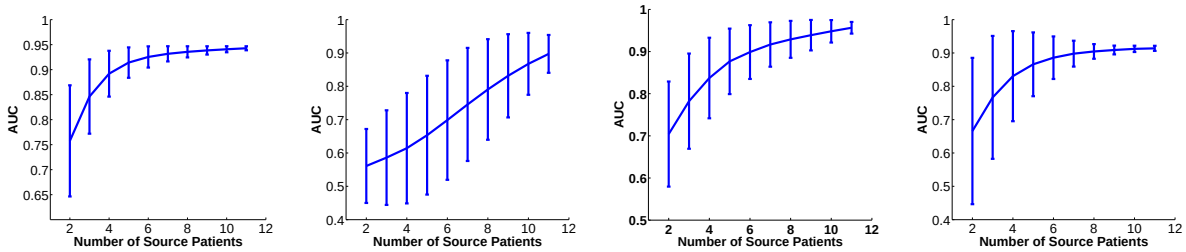

Figure 4: Impact of Number of Sources on Targets: Patient 201, 103, 109 and 215

released by ECML/PKDD 2006 discovery challenge[1]. Its task B contains 15 different users' email box, each of which has different word distributions. The task is to build a spam email filter for each individual user by transferring knowledge from all the other users (sources). The number of normal emails outnumbers that of spam emails in that the percentage of spam emails is roughly 25%.

Intrusion Detection. The KDD cup 99 data set[2] consists of a series of TCP connection records for a local area network. Each example in the data set corresponds to a network connection, which is labeled as either normal or an attack. Attacks fall into four main categories: DOS, R2L, U2R, and Probing. We create three data sets, each contains a large set of randomly chosen normal examples and a set of attacks from one category. The transfer learning scenario is to learn a classifier on the target task domain by transferring knowledge from the other task domains.

5.2 Baseline Methods To properly evaluate the performance of the proposed approach, we compare **SLW** with the following baseline methods: CRC [10], GCM [8], MDA [2] and LWE [9], which we've discussed about in detail in Section 2. In addition, we also include another recent MSTL method DAM [4] as a baseline method. DAM computes the weight of each source by computing the Maximal Mean Discrepancy (MMD) [1] between source samples and target samples. Moreover, to show the benefits of transfer learning, we include Label Propagation [18] as another baseline method, which does not utilize source domain information but only rely on the predictions made by propagation from the small amount of labeled data in the target domain.

5.3 Performance Study In this set of experiments, we use $\gamma = 0.1$, $\beta = 0.3$ and the target training set is 5% of the target data. Table 5 summarizes the performance of all baseline methods and SLW on three data sets. We first notice that CRC generally does not perform well, and sometimes its performance is worse than that of label propagation, which indicates that it suffers from negative transfer. The reason is that CRC tries to output a solution that represents consensus, however, in the cases with many irrelevant sources, consensus will give a wrong solution. GCM works better than CRC when the majority of source domains are relevant, for example, on the spam filtering data set with U03 as the target user. Unfortunately, when many sources are irrelevant, GCM cannot work either, for example, on CAD and intrusion detection problems. On the other hand, MDA and LWE is vulnerable to imbalanced distributions. We observe that on the tasks with highly imbalanced distributions, such as CAD problems, MDA and LWE cannot beat label propagation, but they perform relatively well on the data sets where the imbalanced distribution problem is less severe, such as the spam filtering data sets. In general, DAM has pretty stable performance because it relies on similarity in feature vectors between source and target domains, and thus it is less vulnerable to these two challenges.

Comparing with all these baseline methods, SLW

[1] http://www.ecmlpkdd2006.org/challenge.html
[2] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

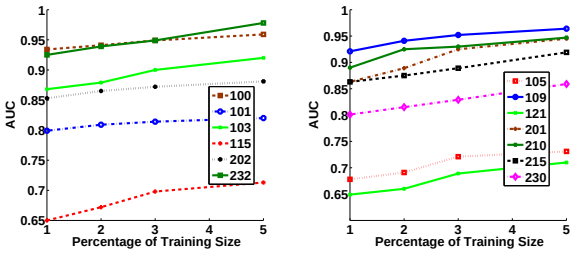

Figure 5: Impact of Training Size

achieves the best performance on almost all the experiments. By utilizing the two-phase framework, the proposed SLW approach can successfully transfer useful knowledge even when irrelevant sources and imbalanced distributions exist. The improvement in AUC can be up to 34.6% and on average 12.3% compared with the best baseline. The advantages of the proposed SLW approach can be observed in more details in Figure 3 where we show the ROC curves of different methods for patients 201, 103, 109 and 215 in CAD prediction, respectively (due to space limit, we only show 4 cases).

Impact of Number of Source Domains. In our theoretical analysis, we prove that as the number of source domains increases, the probability that the proposed SLW approach has an error greater than a bound is becoming exponentially small. Figure 4 shows the impact of the number of source domains on patients 201, 103, 109, and 215 in CAD problem. On the x-axis, each number k represents C_{12}^k experiments, taking all possible k source domains. The plots show the average and variance of the performance with regard to different number of source domains. Results show that the performance of SLW increases monotonically with regard to the number of source domains. The result is consistent with our theoretical analysis.

Impact of Training Set Size. In our problem setting, we maintain that the training set should not be big because it is time-consuming to collect many labels. In the experiments, we choose the training set to be at most 5% of the target data. Figure 5 shows the impact of different training set sizes on the CAD data sets. In each experiment, we use one target patient, and all the other patients are treated as source domains. Results are averaged over 5 randomly chosen training set of the same size. It can be seen that SLW's performance improves if given more labels.

6 Conclusions

Multiple source transfer learning transfers knowledge from multiple source domains to a target domain where labeled data are hard to get. Existing MSTL approaches suffer from *negative transfer* and *imbalanced distributions*. To tackle these challenges, we propose an effective two-phase approach to transfer useful knowledge from multiple source domains, and thus derive accurate and robust predictions on the unlabeled target examples. We propose to first compute a supervised local weight to approximate how likely each source domain will help make the correct predictions. To further guarantee reasonable performance in the worst case scenario when all the sources are irrelevant, we try to minimize a combined loss function involving both target training sets and weighted predictions of source domains. The proposed approach avoids the influence of negative transfer and imbalanced distributions. We present a theoretical analysis to show the error bound of SLW. Experiments on three applications comparing with six baseline methods demonstrate the effectiveness of SLW in multiple source transfer learning, in which it outperforms existing MSTL methods with AUC improvement up to 34.6%.

References

[1] K. Borgwardt, A. Gretton, M. Rasch, H. Kriegel, B. Scholkopf, and A. Smola. Integrating Structured Biological Data by Kernel Maximum Mean Discrepancy. In *Bioinformatics*, 2006.

[2] R. Chattopadhyay, J. Ye, S. Panchanathan, W. Fan, and I. Davidson. Multi-Source Domain Adaptation and Its application to Early Detection of Fatigue. In *Proc. of KDD*, 2011.

[3] Q. Sun, R. Chattopadhyay, S. Panchanathan, J. Ye. A Two-StageWeighting Framework for Multi-Source Domain Adaptation. in *Proc. of NIPS*, 2011

[4] L. Duan, I. Tsang, D. Xu, and T. Chua. Domain Adaptation from Multiple Sources via Auxiliary Classifiers. In *Proc. of ICML*, 2009.

[5] W. Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. In *Journal of the American Statistical Association*, 1963.

[6] M. Bahadori, Y. Liu and D. Zhang. Learning with Minimum Supervision: A General Framework for Transductive Transfer. Learning In *ICDM*, 2011

[7] M. Long, J. Wang, G. Ding, W. Cheng, X. Zhang and W. Wang. Dual Transfer Learning. In *SDM*, 2012

[8] J. Gao, F. Liang, W. Fan, Y. Sun, and J. Han. Graph-based Consensus Maximization among Multiple Supervised and Unsupervised Models. In *Proc. of NIPS*, 2009.

[9] J. Gao, W. Fan, J. Jiang, and J. Han. Knowledge Transfer via Multiple Model Local Structure Mapping. In *Proc. of KDD*, 2008.

[10] P. Luo, F. Zhuang, H. Xiong, Y. Xiong, and Q. He. Transfer Learning From Multiple Source Domains via Consensus Regularization. In *Proc. of CIKM*, 2008.

[11] G. Moody and R. Mark. The Impact of the MIT-BIH Arrhythmia Database. In *Proc. of IEEE Engineering in Medicine and Biology Society*, 2001.

[12] S. Pan and Q. Yang. A Survey on Transfer Learning. In *TKDE*, 2010.

[13] B. Cao, S. Pan, Y. Zhang, D. Yeung and Q. Yang. Adaptive Transfer Learning. In *AAAI*, 2010

[14] M. Rosenstein, Z. Marx, and L. Kaelbling. To transfer or Not to Transfer. In *NIPS 2005 Workshop on Inductive Transfer: 10 Years Later*, 2005.

[15] B. Scholkopf and A. Smola. Learning with Kernels Support Vector Machines, Regularization, Optimization and Beyond. *The MIT Press*, 2002.

[16] N.V. Chawla, N. Japkowicz, and A. Kotcz. Editorial: special issue on learning from imbalanced data sets *SIGKDD Explor. Newsl.*, 2004

[17] J. Shi. and J. Malik. Normalized Cuts and Image Segmentation. In *PAMI*, 2000.

[18] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with Local and Global Consistency. In *Proc. of NIPS*, 2003.