Journal of Machine Learning Research 20 (2019) 1-31 Submitted 12/13; Revised 4/17; Published 4/19

Multi-class Heterogeneous Domain Adaptation

Joey Tianyi Zhou ZHOUTY@IHPC.A-STAR.EDU.SG
Institute of High Performance Computing (IHPC)
*Agency for Science, Technology and Research (A*STAR),*
Singapore 138632

Ivor W. Tsang IVOR.TSANG@UTS.EDU.AU
Centre for Artificial Intelligence
University of Technology Sydney
Australia

Sinno Jialin Pan SINNOPAN@NTU.EDU.SG
School of Computer Science and Engineering
Nanyang Technological University,
Singapore 639798

Mingkui Tan MINGKUITAN@SCUT.EDU.CN
School of Software Engineering
South China University of Technology,
China

Editor: Shie Mannor

Abstract

A crucial issue in heterogeneous domain adaptation (HDA) is the ability to learn a feature mapping between different types of features across domains. Inspired by language translation, a word translated from one language corresponds to only a few words in another language, we present an efficient method named *Sparse Heterogeneous Feature Representation (SHFR)* in this paper for multi-class HDA to learn a sparse feature transformation between domains with multiple classes. Specifically, we formulate the problem of learning the feature transformation as a compressed sensing problem by building multiple binary classifiers in the target domain as various measurement sensors, which are decomposed from the target multi-class classification problem. We show that the estimation error of the learned transformation decreases with the increasing number of binary classifiers. In other words, for adaptation across heterogeneous domains to be successful, it is necessary to construct a sufficient number of incoherent binary classifiers from the original multi-class classification problem. To achieve this, we propose to apply the error correcting output correcting (ECOC) scheme to generate incoherent classifiers. To speed up the learning of the feature transformation across domains, we apply an efficient batch-mode algorithm to solve the resultant nonnegative sparse recovery problem. Theoretically, we present a generalization error bound of our proposed HDA method under a multi-class setting. Lastly, we conduct extensive experiments on both synthetic and real-world datasets to demonstrate the superiority of our proposed method over existing state-of-the-art HDA methods in terms of prediction accuracy and training efficiency.

Keywords: Heterogeneous domain adaptation, multi-class classification, compressed sensing.

1. Introduction

In many applications, it is often expensive or time consuming to annotate sufficient labeled training data to build machine learning-based information systems. There may therefore be only a few labeled training data in the domain of interest but many labeled training data in another domain that is referred to as a source domain. Because of the differences between domains, a model trained on the labeled data in the source domain cannot be applied directly to the target domain. To address this problem, domain adaptation that aims to adapt a model from a source domain to a target domain with little or no additional human supervision has been proposed (Blitzer et al., 2006; Jiang and Zhai, 2007; Huang et al., 2007; Pan et al., 2008, 2011). Apart from thorough theoretical studies (Ben-David et al., 2006; Ben-David et al., 2010), domain adaptation techniques have been successfully applied to various applications, such as WiFi localization (Pan et al., 2008, 2011), Natural Language Processing applications (Daumé III, 2007; Jiang and Zhai, 2007), sentiment analysis (Blitzer et al., 2007b; Pan et al., 2010; Seah et al., 2011; Zhou et al., 2014, 2016a), object categorization (Duan et al., 2009; Gong et al., 2012; Hoffman et al., 2013), object detection (Donahue et al., 2013; Hoffman et al., 2014) and information retrieval (Zhou et al., 2016b, 2018).

Most existing domain adaptation works are based on the assumption that different domain data can be represented by the same feature space of the same dimensionality (Pan and Yang, 2010). However, this assumption may not hold in many real-world scenarios. For instance, in a cross-language (e.g., English/Spanish) text classification task (Prettenhofer and Stein, 2010), the feature spaces are referred to as vocabularies in different languages, and the features across domains are neither in the same feature space nor of the same dimensionality. To relax this assumption, there has been an increased focus on domain adaptation across heterogeneous feature spaces, which is referred to as Heterogeneous Domain Adaptation (HDA) (Dai et al., 2008; Yang et al., 2009). Besides cross-language text classification, many other real-world applications can be formulated as an HDA problem, such as an image classification task using auxiliary text data (Zhu et al., 2011), where the heterogeneous features are referred to as pixels and bags of textual words, respectively, image classification by using auxiliary images with different sets of features (Saenko et al., 2010; Kulis et al., 2011), where the heterogeneous features are referred to as different computer vision features, and so on.

Existing HDA approaches can be classified into two main categories. In the first category, two dense feature mappings $\mathbf{P}$ and $\mathbf{Q}$ can be learned to transform the source domain data $\mathbf{X}_S$ and target domain data $\mathbf{X}_T$, respectively, to a new latent common feature space such that the difference between the mapped domain data $\mathbf{P}\mathbf{X}_S$ and $\mathbf{Q}\mathbf{X}_T$ is reduced (Shi et al., 2010; Prettenhofer and Stein, 2010; Wang and Mahadevan, 2011; Duan et al., 2012). For example, Shi et al. (2010) proposed a heterogeneous spectral mapping (HeMap) method to learn dense orthogonal mappings based on spectral embedding without using any label information. The resultant optimization problem is a standard eigen-decomposition problem. However, this approach is known to suffer from the scalability issue. Wang and Mahadevan (2011) proposed a manifold alignment method known as DAMA to align heterogeneous features into a latent space based on manifold regularization. Unfortunately, DAMA only works on data that has a manifold structure. This limits its transferability to data on which the manifold assumption is satisfied. Furthermore, the use of DAMA results in a generalized eigen-decomposition problem of a series of matrices whose sizes depend on the dimensionality of the data. Duan et al. (2012) proposed a Heterogeneous Feature Augmentation (HFA) method to augment heterogeneous features with homogeneous common features learned using a maximum-

margin approach from both source and target domains. However, the proposed model results in an expensive semidefinite program (SDP) problem.

In the second category, a dense feature mapping $\mathbf{G}$ can be learned to transform heterogeneous data from one domain to the other domain directly so that the difference between $\mathbf{G}\mathbf{X}_S$ and $\mathbf{X}_T$ can be minimized or the alignment between $\mathbf{G}\mathbf{X}_S$ and $\mathbf{X}_T$ can be maximized. Harel and Mannor (2011) proposed a method named MOMAP to learn dense rotation matrices to match data distributions between the source and target domains. The resultant optimization problem is solved through singular value decomposition (SVD) for each class in an independent way. Kulis et al. (2011) proposed an asymmetric regularized cross-domain transformation (ARC-t) method to learn asymmetric transformation across domains based on metric learning. Similar to DAMA, ARC-t also utilizes the label information to construct the similarity and dissimilarity constraints between instances from the source and target domains, respectively. However, the computational complexities of ARC-t and its kernelized version depend quadratically on the feature dimensions and the data size, respectively.

Though most existing HDA methods have shown promising results, they still suffer from the following three major limitations.

1. **Dense Feature Mappings.** Existing methods tend to recover dense feature mappings, which is not tenable without sufficient constraints. As we will show, dense feature mappings may inevitably lose interpretability and contain significant amount of noise, which may affect the classification performance.

2. **Multi-class problem.** To address multi-class classification problems, most existing HDA methods (Dai et al., 2008; Wang and Mahadevan, 2011; Duan et al., 2012) simply adopt the one-vs-all strategy to learn multiple binary classifiers independently. Unfortunately, this strategy fails to fully explore the underlying structure among multiple classes. Consequently, this heuristic scheme is unable to guarantee good performance for multi-class HDA problems.

3. **High Computational Cost.** Since the size of the feature mapping $\mathbf{G}$ scales with the product of the dimensionality of the source and target domains, the computational cost to estimate the mapping is extremely high, especially for high-dimensional source and target data, e.g. MOMAP, HeMap, DAMA, ARC-t. To address this computational issue, Duan et al. (2012) and Kulis et al. (2011) proposed kernelized versions to learn the feature mapping. However, the kernelized methods still suffer from high computational cost on large-scale data in terms of the number of data instances.

2. Motivations and Contributions

In this work, we propose a **Sparse Heterogeneous Feature Representation** (SHFR) approach to address the above issues under following assumptions.

2.1 Motivations

1. **Sparse feature representation:** The feature mapping $\mathbf{G}$ between two domains is row-sparse, i.e., each target domain feature can be represented by a small subset of the source domain features.

2. **Class-invariance transformation:** Instances belonging to different classes share the same feature mapping $\mathbf{G}$.

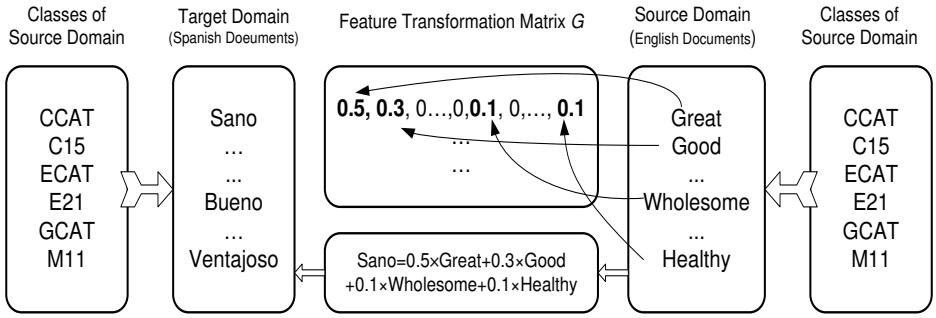


Figure 1: Illustration of the sparse feature representation matrix.

The above two assumptions are common in real-world multilingual text categorization applications. Recall that the sparsity assumption of the feature mapping across domains implies that each feature in one domain can only be represented by a small subset of features in another domain. Here, we still use the multi-language (i.e., English/Spanish) text classification problem as a motivating example. Typically, the word "Sano" in Spanish has a similar meaning to the words "Great", "Good", "Wholesome", and "healthy" in English, but not to all English words. Therefore, by assuming that the feature mapping across domains is linear, a feature or word in the Spanish domain can be represented by a linear combination of only several features or words in the English domain. As illustrated in Figure 2.1, the sparse matrix $\mathbf{G}$ denotes the feature mapping from the English domain to the Spanish domain. Based on the sparse matrix $\mathbf{G}$, the word "Sano" in Spanish can be represented sparsely by only four words in English as "Sano$''$ $=$ $0.5\times$ "Great$''+0.3\times$ "Good$''+0.1\times$ "Wholesome$''+0.1\times$ "Healthy$''$. This sparsity characteristic, which also facilitates a significant reduction in computational cost on very high-dimensional data, has not been explored in existing HDA methods. The feature mapping of the word "Sano" is invariant for different classes, for example, "CCAT" and "C15". To encode the *class-invariance* property into the feature mapping, we propose to learn a common $\mathbf{G}$ underlying all the classes which has a similar spirit to multi-task feature learning (Argyriou et al., 2007) and domain-invariant feature learning (Gong et al., 2013). We first decompose a multi-class classification problem into multiple binary classification tasks, and then jointly optimize all the binary classification tasks and the feature mapping $\mathbf{G}$, such that $\mathbf{G}$ is class-invariant to all the binary classification tasks.

2.2 Our Contributions

Based on these observations, we propose a new approach to learning a feature mapping for HDA with application to cross-lingual multi-class text categorization. To estimate this a feature mapping, we leverage the weight vectors of the binary classifiers learned in the source and target domains. We then formulate the problem of learning the feature mapping between domains as a Compressed Sensing (CS) problem, which aims to recover a sparse signal from a small number of measurements with respect to an underdetermined linear system by taking advantage of the sparseness or compressibility of the signal (Donoho, 2006; Candès et al., 2006). The main contributions of this paper are as follows:

1. We propose a **Sparse Heterogeneous Feature Representation** (SHFR) approach to learn a sparse transformation for HDA by exploring the common underlying structures between multiple classes.

2. Our major finding is that, based on the CS theory, a sparse feature mapping can be learned if and only if a sufficient number of classifiers are provided. Based on this analysis, we propose to generate sufficient classifiers under the **Error Correcting Output Correcting** (ECOC) scheme to accurately estimate the sparse feature mapping.

3. To reduce the high computational cost, we propose a **batch-mode** pursuit algorithm to efficiently solve the resultant nonnegative lasso problem of SHFR. The computational cost of SHFR is independent of the data size and the dimensionality of the source domain data, and scales linearly only with size of the dimensionality of the target domain data.

4. This paper is the first to present a **generalization error bound** for multi-class HDA which is dependent on the number of tasks as well as the degree of sparsity of the feature mapping $\mathbf{G}$.

The remainder of the paper is organized as follows. In Section 3, we first present how to formulate an HDA problem as a compressed sensing problem, and describe the detail of the proposed method, SHFR. We then theoretically analyze a generalization error bound for SHFR in Section 4, and develop a batch-mode algorithm in Section 5 to efficiently solve the resultant compressed sensing problem. In Section 6, we conduct extensive experiments on both toy and real-world datasets to demonstrate the effectiveness and efficiency of SHFR. We conclude this paper and discuss future work in Section 7.

3. Sparse Heterogeneous Feature Representation

We study the HDA problem with one source domain and one target domain under the multi-class setting. Let $\{(\mathbf{x}_{S_i}, y_{S_i})\}_{i=1}^{n_S}$ denote a set of labeled training instances of the source domain, where $\mathbf{x}_{S_i} \in \mathbb{R}^{d_S}$ denotes the i-th instance, and $y_{S_i} \in \{1, 2, \cdots, c\}$ denotes its label. Let $\{(\mathbf{x}_{T_i}, y_{T_i})\}_{i=1}^{n_T}$ be a set of labeled training instances of the target domain, where $n_T \ll n_S$, $\mathbf{x}_{T_i} \in \mathbb{R}^{d_T}$, and $y_{T_i} \in \{1, 2, \cdots, c\}$.

Given a binary task $t \in \{1, 2, \cdots, n_c\}$, we assume that the classifiers for both source and target domain are linear, which can be written as $f^t(\mathbf{x}) = \mathbf{w}^{t\top}\mathbf{x}$, where $\mathbf{w}^t$ is the weight vector of the t-th classifier. Since there are sufficient labeled data in the source domain, i.e., $\{(\mathbf{x}_{S_i}, y_{S_i})\}_{i=1}^{n_S}$, we can learn a robust set of weight vectors $\{\mathbf{w}_S^t\}_{t=1}^{n_c}$ for the source classifiers of the binary tasks decomposed from the multi-class classification problem. Similarly, we can build a corresponding set of weight vectors $\{\mathbf{w}_T^t\}_{t=1}^{n_c}$ with limited labeled data $\{(\mathbf{x}_{T_i}, y_{T_i})\}_{i=1}^{n_T}$ for the target domain.

3.1 Feature Mapping for HDA

Recall that, for HDA problems, the feature dimensions of the source and target domains may not be equal, i.e., $d_S \neq d_T$. To make effective learning across heterogeneous domains possible, a transformation matrix $\mathbf{G} \in \mathbb{R}^{d_T \times d_S}$ was introduced in ARC-t (Kulis et al., 2011; Saenko et al., 2010) to learn the similarity $\mathbf{x}_{T_i}^\top \mathbf{G} \mathbf{x}_{S_i}$ between a source instance $\mathbf{x}_{S_i} \in \mathbb{R}^{d_S}$ and a target instance $\mathbf{x}_{T_i} \in \mathbb{R}^{d_T}$. Data points can be transformed from the source feature space to the target feature

space via $\mathbf{x}_{T_i} = \mathbf{G}\mathbf{x}_{S_i}$, or equivalently, the target domain data can be mapped into the source domain via $\mathbf{G}^\top$ (Saenko et al., 2010). To learn the transformation matrix $\mathbf{G}$, corresponding pairs across domains are usually supposed to be given in advance (Qi et al., 2011) or generated based on the label information in both the source and target domains (Saenko et al., 2010). However, in practice, instance-correspondences between domains are usually missing, or the constructed instance correspondences using label information are not precise. Therefore, instead of learning the transformation by using the input raw data directly, we adapt an idea from a multi-task learning method (Ando and Zhang, 2005)[1], and propose to learn the feature mapping across heterogeneous features based on the source and target predictive structures, i.e., $\{\mathbf{w}_S^t\}$'s and $\{\mathbf{w}_T^t\}$'s.

We can either learn the transformation matrix $\mathbf{G} \in \mathbb{R}^{d_T \times d_S}$ by maximizing the dependency between the transformed weight vectors of source classifiers and the weight vectors of target classifiers via

$$\max_{\mathbf{G}} \mathbf{w}_T^{t\top} \mathbf{G} \mathbf{w}_S^t,$$

or by minimizing the distance between these two vectors via

$$\min_{\mathbf{G}} \|\mathbf{w}_T^t - \mathbf{G}\mathbf{w}_S^t\|.$$

For simplicity in theoretical analysis, we adopt the latter objective to learn the transformation $\mathbf{G}$. Given a binary task $t \in \{1, \cdots, n_c\}$ and a transformation $\mathbf{G} \in \mathbb{R}^{d_T \times d_S}$, the relationship between the weight vectors of the source and target classifiers can be modeled as follows,

$$\mathbf{w}_T^t - \mathbf{G}\mathbf{w}_S^t = \mathbf{w}_\Delta^t,$$

where $\mathbf{w}_\Delta^t$ is referred to as a "delta" weight vector, and its ℓ_2-norm $\|\mathbf{w}_\Delta^t\|_2$ is used to measure the difference between the target weight vector $\mathbf{w}_T^t$ and the transformed source weight vector $\mathbf{G}\mathbf{w}_S^t$. Note that the relationship is linear. This is because, in this paper, we focus on the application of HDA on cross-language text categorization, where using a linear form to model the relationship of predictive structures between domains is more suitable for high-dimensional text data. For modeling a nonlinear relationship between the source and target predictive structures, which may be more useful in other application areas, e.g., image classification or speech recognition, it is possible to apply an explicit nonlinear feature mapping $\phi(\mathbf{x})$ to both the source and target domain data before training the weight vectors for each domain.

3.2 Proposed Formulation

In order to use the robust source domain weight vector $\mathbf{w}_S^t$ to make predictions on the target domain data, the difference between the source and target domains after transformation should be minimized. We therefore propose to learn the transformation $\mathbf{G}$ by minimizing $\|\mathbf{w}_\Delta^t\|_2$. As mentioned in Section 2, the feature mapping $\mathbf{G}$ should be class-invariant and sparse, and for multi-language text classification and many other real-world applications, the feature mapping $\mathbf{G}$ between feature spaces should be nonnegative. The motivation of adding the nonnegative constraint on the feature mapping $\mathbf{G}$ is similar to that of applying nonnegative matrix factorization (NMF) (Lee and Seung,

1. Multi-task learning aims to simultaneously learn the classifiers for multiple tasks by sharing parameters between all the tasks, where each task has only limited labeled data. In contrast, the goal of HDA is to learn a classifier only for a target domain with limited target label data by leveraging the models learned from the source domain.

2001) to text mining. Here, we aim to approximate "input vectors", i.e., target features, by a nonnegative linear combination of a set of nonnegative "basis vectors", i.e., source features, through the mapping $\mathbf{G}$. This is also similar to multiple kernel learning (MKL) (Lanckriet et al., 2004; Gönen and Alpaydın, 2011), where the target kernel is represented by the nonnegative linear combination of base kernels. In the context of multi-lingual text classification, the nonnegative entries in $\mathbf{G}$ can be deemed as explicit "soft" global co-occurrences between the source and target wordbooks, while without the nonnegative constraint on $\mathbf{G}$, it is difficult to interpret the physical or semantic meaning behind $\mathbf{G}$.

Let $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_{d_T}]^\top$, by imposing the ℓ_1-regularization and the nonnegative constraint on $\mathbf{g}_i$, the learning of $\mathbf{G}$ can be formulated as the following nonnegative LASSO problem (Yuan and Lin, 2007; Slawski and Hein, 2011),

$$\begin{aligned} \min_{\mathbf{G}} \quad & \frac{1}{n_c}\sum_{t=1}^{n_c} \|\mathbf{w}_T^t - \mathbf{G}\mathbf{w}_S^t\|_2^2 + \sum_i^{d_T} \lambda_i \|\mathbf{g}_i\|_1, \\ s.t. \quad & \mathbf{g}_i \succeq \mathbf{0}, \end{aligned} \tag{1}$$

where $\lambda_i > 0$ is the trade-off parameter for the regularization term on each $\mathbf{g}_i$. In (1), the first term in the objective aims to minimize the difference between $\mathbf{w}_T^t$ and $\mathbf{G}\mathbf{w}_S^t$ over all the n_c tasks, the second term in the objective is to enforce the sparsity on each row of $\mathbf{G}$, and the constraints are used to preserve nonnegative linear relations between the source and target predictive structures.

Note that in practice, once the source domain classifiers in terms of the weight vectors $\{\mathbf{w}_S^t\}_{t=1}^{n_c}$ are learned offline, the source domain training data can be discarded. For a new domain of interest, i.e., the target domain, the weight vectors $\{\mathbf{w}_T^t\}_{t=1}^{n_c}$ can first be learned with a few labeled training data, and then $\mathbf{G}$ can be learned with $\{\mathbf{w}_S^t\}_{t=1}^{n_c}$ instead of the original source domain training data, which significantly reduces the learning complexity. This learning scheme is typically different from most of the existing HDA methods that require the original source domain training data to be available to learn the feature mapping across domains. More discussion on the complexity issue can be found in Section 5.1. As $\mathbf{G}$ is learned through all the binary classification tasks, knowledge can be transferred from the easy tasks to the difficult ones. In Section 4.2, we theoretically show that as long as a significant number of base binary tasks are not badly designed, the generalization error of the target classification model is guaranteed to be small.

To solve the optimization problem (1), we can first rewrite it in the following equivalent form,

$$\begin{aligned} \min_{\mathbf{g}_i} \quad & \frac{1}{n_c}\sum_{t=1}^{n_c}\sum_{i=1}^{d_T} (w_{T_i}^t - \mathbf{w}_S^{t\top}\mathbf{g}_i)^2 + \sum_i^{d_T} \lambda_i \|\mathbf{g}_i\|_1, \\ s.t. \quad & \mathbf{g}_i \succeq \mathbf{0}, \end{aligned} \tag{2}$$

where $w_{T_i}^t$ is the i-th element of the vector $\mathbf{w}_T^t$. If we exchange the summation sequences in the first term, the above optimization problem can be reformulated as the following CS problem,

$$\begin{aligned} \min_{\mathbf{g}_i} \quad & \sum_i^{d_T} \left(\frac{1}{n_c}\|\mathbf{b}_i - \mathbf{D}\mathbf{g}_i\|_2^2 + \lambda_i \|\mathbf{g}_i\|_1 \right), \\ s.t. \quad & \mathbf{g}_i \succeq \mathbf{0}, \end{aligned} \tag{3}$$

where $\mathbf{b}_i$ is the concatenated row vector containing $\{w_{T_i}^t\}_{t=1}^{n_c}$, and $\mathbf{D} = [\mathbf{w}_S^1\ \mathbf{w}_S^2\ \cdots\ \mathbf{w}_S^{n_c}]^\top \in \mathbb{R}^{n_c \times d_S}$.

This type of CS problem has been well studied, and a number of algorithms have been proposed to solve it, such as nonnegative least squares (NNLS) (Cantarella and Piatek, 2004), accelerated proximal gradient algorithm (APG) (Toh and Yun, 2010), and orthogonal matching pursuit (OMP) (Zhang, 2009). In this work, we propose an efficient batch-mode algorithm to solve it, which is introduced in detail in Section 5. After the sparse transformation $\mathbf{G}$ has been learned, we can reuse the source domain classifiers to predict its label for any unseen test data $\mathbf{x}_T^*$ from the target domain via

$$y_T^* = F(\{(\mathbf{G}\mathbf{w}_S^t)^\top \mathbf{x}_T^*\}_{t=1}^{n_c}),$$

where $F(\cdot)$ is a decision function that combines the predictive results of all the n_c source classifiers to make a final prediction.

4. Generalization Error Bound for HDA

In the past decade, theoretical analysis for domain adaptation has been widely studied (Blitzer et al., 2007a; Mansour et al., 2009; Ben-David et al., 2010), and has focused on understanding conditions and assumptions of domain adaptation. However, most previous theoretical works were restricted by the assumption that the data from the source and target domains are represented in the same feature space and therefore cannot be applied directly to analyze the generalization error of our proposed method, SHFR, under the HDA setting. In this section, we analyze the theoretical guarantee of SHFR in detail. In contrast to existing generalization error analysis for homogeneous domain adaptation, we derive the generalization error bound for SHFR mainly from the perspectives of ECOC and compressed sensing.

4.1 Estimation Error Bound for Sparse Feature Mapping

Recall that the optimization problem (3) is composed of d_T lasso problems. In general, we have $n_c < d_S$ for relatively high-dimensional data, therefore (3) is an underdetermined linear system (Donoho, 2006). However, based on the compressed sensing theory, if $\mathbf{g}_i$ is sparse, it is possible to obtain a solution with sufficient measurements and a matrix $\mathbf{D}$ that satisfies certain conditions (Donoho, 2006; Candès et al., 2006). In particular, one such condition is the sparse Riesz condition or RIP condition, which requires that any two columns of $\mathbf{D}$ should be as perfectly incoherent as possible (Donoho, 2006; Zhang and Huang, 2008; Candès et al., 2006).

For simplicity in presentation, let k_i denote the number of non-sparse entries of $\mathbf{g}_i$ (or non-sparsity degree), and $\widehat{\mathbf{g}}_i$ denote an estimator of $\mathbf{g}_i$. According to Theorem 3 and Remark 4 in (Zhang and Huang, 2008), under some restricted conditions, the estimation error $||\mathbf{g}_i - \widehat{\mathbf{g}}_i||_2$ for each independent subproblem can be bounded by $O\left(\sqrt{\frac{k_i \log d_S}{n_c}}\right)$, where d_S denotes the dimensionality of the source domain data and n_c is the number of tasks. Hence, we can obtain the following lemma.

Lemma 1 *Under the sparse Riesz condition, the estimation error $\|\Delta\mathbf{G}_{CS}\|_F = \|\mathbf{G} - \widehat{\mathbf{G}}\|_F$ in (1) is bounded by $O\left(d_T\sqrt{\frac{k \log d_S}{n_c}}\right)$, where k, d_S, d_T, n_c, $\widehat{\mathbf{G}}$ denote the largest row non-sparsity degree of $\mathbf{G}$, the dimensionality of the source and target domain data, the number of tasks, and the estimator of $\mathbf{G}$, respectively.*

Note that besides the sparse Riesz condition, Lemma 1 also holds under some other restricted conditions, such as the RIP condition (Zhang and Huang, 2008; Candès et al., 2006). Both the sparse Riesz condition and the RIP condition require that any two columns of $\mathbf{D}$ should be as incoherent as possible (Donoho, 2006; Zhang and Huang, 2008; Candès et al., 2006). However, although the RIP condition for randomly designed matrices has been thoroughly investigated (Baraniuk et al., 2008), it is more difficult to investigate it for deterministically designed matrices in general. In contrast, the sparse Riesz condition for a deterministically designed matrix holds with high probability if the following condition is satisfied (see Proposition 1 in (Zhang and Huang, 2008)):

$$\max_{|A|=k} \inf_{\alpha \geq 1} \left\{ \sum_{j \in A} \left(\sum_{i \in A, i \neq j} |\rho_{ji}|^{\alpha/(\alpha-1)} \right)^{\alpha-1} \right\}^{1/\alpha} \leq \delta < 1,$$

where k denotes the non-sparsity degree and $\rho_{ji} = \mathbf{d}_j' \mathbf{d}_i / n_c$.

Remark 2 *Note that ρ_{ji} can be deemed as the correlation between the columns $\mathbf{d}_j$ and $\mathbf{d}_i$. Therefore, the sparse Riesz condition for a deterministically designed matrix can be evaluated by the correlations between the columns of the dictionary matrix. The smaller the summary of correlations $\sum |\rho_{ji}|$ (or δ) is, the more likely it is that the sparse Riesz condition for $\mathbf{D}$ will hold.*

According to Lemma 1 and Remark 2, to reduce the reconstruction error of the feature mapping $\mathbf{G}$, it is necessary to 1) construct as many classifiers as possible to increase the number of measurements, i.e., n_c needs to be large enough, and 2) ensure the columns of $\mathbf{D}$ constructed from the classifiers are incoherent, i.e., $\mathbf{D}$ is incoherent. For a multi-class classification problem with labels $\{1, 2, \cdots, c\}$, c binary classifiers can be generated using the one-vs-all strategy (Dietterich and Bakiri, 1995). However, when c is small, this strategy is not able to generate sufficient binary classifiers. Alternatively, the one-vs-one strategy may be used to generate $c(c-1)/2$ binary classifiers. The classifiers generated in this way may have large redundancy, however, i.e., some classifiers may be highly correlated to each other. To address this issue, we propose to use the Error Correcting Output Codes (ECOC) scheme (Dietterich and Bakiri, 1995; Zhou et al.) to generate sufficient binary classifiers for estimating the feature mapping $\mathbf{G}$. As will be shown empirically in Section 6.5, greater incoherence for $\mathbf{D}$ over the one-vs-all or one-vs-one strategy can be guaranteed with a proper design of the ECOC coding matrix.

4.2 Generalization Error Analysis of HDA Based on ECOC

By using the ECOC scheme, the multi-class hypothesis h is composed of a set of binary classifiers $\{\widehat{\mathbf{G}}\mathbf{w}_S^t\}_{t=1}^{n_c}$. However, in contrast to the one-vs-all scheme, ECOC is more general and consists of two steps: encoding, i.e., to design a coding matrix M to induce binary classifiers, and decoding, i.e., to combine the results of all the binary classifiers to make predictions based on the coding matrix. In order to analyze the generalization error bound of SHFR for multi-class HDA based on ECOC, we need to first analyze the principles of ECOC for multi-class classification.

As discussed in Dietterich and Bakiri (1995); Allwein et al. (2001), the prediction performance of a multi-class classification model learned based on the ECOC scheme depends on the design of the coding matrix. Given a coding matrix $M = \{-1, 0, +1\}^{c \times n_c}$, each class is associated with a row of the matrix M, and each binary classifier is associated with a column of the matrix M. If

an entry $M(i,j) = +1$, it implies that in the classifier f^j, class i is considered as a positive class, while if $M(i,j) = -1$, it implies that in the classifier f^j, the class i is considered as a negative class, otherwise, the class i is not taken into account for training and testing. In this way, either the one-vs-all scheme or the one-vs-one scheme can be considered as a special case of the ECOC scheme.

As pointed out by Dietterich and Bakiri (1995), a coding matrix is good if each codeword, i.e., each row, is well-separated in the Hamming distance from each of the other codewords, and each bit-position function f^k, i.e., each column, is uncorrelated with the functions to be learned for the other bit positions f^j, $j \neq k$. The power of a coding matrix M for multi-class classification can be measured by the minimum Hamming distance between any pair of its codewords (Allwein et al., 2001), denoted by $\Delta_{min}(M)$. For instance, for the one-vs-all code, $\Delta_{min}(M) = 2$, and for the one-vs-one code, $\Delta_{min}(M) = \left(\left(\begin{array}{c} c \\ 2 \end{array}\right) - 1\right)/2 + 1$. For a random matrix with components chosen over $\{-1, 0, 1\}$, the expected value of $\Delta_{min}(M)$ for any distinct pair of rows is $n_c/2$.

Here, we first follow the proof procedure on error analysis for multi-class classification described in Allwein et al. (2001) to analyze the condition of a testing error by the ECOC decoding. After that, we derive the generalization error bound for the proposed SHFR with the ECOC scheme.

Given a coding matrix $M \in \{-1, 0, +1\}^{c \times n_c}$, each row corresponds to a class and each column corresponds to a binary task. For some $t \in \{1, 2, \cdots, n_c\}$, we denote $\mathbf{w}_S^t$ and h_S^t as the source domain weight vector and the hypothesis of the t-th task defined by the coding matrix M, respectively. We denote $\widehat{\mathbf{G}}\mathbf{w}_S^t$ and h_T^t as the estimated target domain weight vector and hypothesis of the t-th task, respectively. We also define the loss function,

$$L(z) = (1 - z)_+ = \max\{1 - z, 0\}, \tag{4}$$

in terms of the margin z. For example, given a target domain instance $\mathbf{x}_{T_i}$ with its label y_{T_i}, the corresponding margin $z_{T_i}^t$ for t-th task is defined as

$$z_{T_i}^t = M(y_{T_i}, t) h_T^t(\mathbf{x}_{T_i}) = M(y_{T_i}, t)((\widehat{\mathbf{G}}\mathbf{w}_S^t)^\top \mathbf{x}_{T_i}). \tag{5}$$

For instance, SVM seeks to minimize the margin-based loss under L_2 norm regularization as follows,

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|_2^2 + C \sum_{i=1}^{m} (1 - y_i \mathbf{w}^\top \mathbf{x}_i)_+, \tag{6}$$

where we denote margin $z = y_i \mathbf{w}^\top \mathbf{x}_i$. When the prediction is correct and $M(y_{T_i}, t) \neq 0$ for $\mathbf{x}_{T_i}$, then the margin $z_{T_i}^t > 0$, which results in $0 \leq L(z) < 1$; when the prediction is wrong and $M(y_{T_i}, t) \neq 0$ for $\mathbf{x}_{T_i}$, then the margin $z_{T_i}^t < 0$, which results in $L(z) > 1$; when $M(y_{T_i}, t) = 0$, then the margin $\mathbf{z}_{T_i}^t = 0$, which results in $L(z) = 1$. We define the $L(z)$ loss-based decoding scheme for prediction as follows:

$$d_L(M(r,:), h(\mathbf{x}_{T_i})) = \sum_{t=1}^{n_c} L(M(r,t) h_T^t(\mathbf{x}_{T_i})). \tag{7}$$

The predicted label $\hat{y} \in \{1, 2, \cdots, c\}$ is given by $\hat{y} = \arg\min_r d_L(M(r), h(\mathbf{x}_{T_i}))$.

To understand the error correction ability of ECOC, we first define the distance between the codes in any distinct pair of rows, $M(r_1,:)$ and $M(r_2,:)$, in the coding matrix M as

$$d(M(r_1,:),M(r_2,:)) = \sum_{s=1}^{n_c} d(M(r_1,s),M(r_2,s)), \tag{8}$$

where $d(M(r_1,s),M(r_2,s)) = \frac{1-sign(M(r_1,s)M(r_2,s))}{2}$.

We define $\rho = \min_{r_1 \neq r_2} d(M(r_1,:),M(r_2,:))$ as the minimum distance between any two rows in the coding matrix M, which is pre-calculated before the training.

Proposition 3 *Given a coding matrix $M \in \{-1,0,+1\}^{c\times n_c}$, a set of hypothesis outputs $h^1(\mathbf{x}),\cdots,h^{n_c}(\mathbf{x})$ on a test instance $\mathbf{x}$ generated by n_c base classifiers, and a convex loss function $L(z)$ in terms of the margin z, if $\mathbf{x}$ is misclassified by the loss-based ECOC decoding, then*

$$\sum_{t=1}^{n_c} L(M(y,t)h^t(\mathbf{x})) \geq \rho L(0), \tag{9}$$

where $M(y,t)$ denotes the corresponding label generated by the coding matrix M of ground truth. In other words, the loss of the vector of predictions $f(\mathbf{x})$ is greater than $\rho L(0)$.

Proof Suppose that the loss-based ECOC decoding incorrectly classifies a test instance $\mathbf{x}$ with the ground-truth label y. Then should then exist a label $r \neq y$ such that

$$d_L(M(y,:),h(\mathbf{x})) \geq d_L(M(r,:),h(\mathbf{x})).$$

By using the definitions of the loss function and margin introduced in (4) and (5), we denote

$$z^t = M(y,t)h^t(\mathbf{x}),$$
$$z'^t = M(r,t)h^t(\mathbf{x}),$$

By using the definitions of d_L in (7), we have

$$\sum_{t=1}^{n_c} L(z^t) \geq \sum_{t=1}^{n_c} L(z'^t). \tag{10}$$

Let $S_\Delta = \{t : M(r,t) \neq M(y,t) \wedge M(r,t) \neq 0 \wedge M(y,t) \neq 0\}$ be the set of columns of M whose r-th and y-th row entries are different and nonzero, and let $S_0 = \{t : M(r,t) = 0 \vee M(y,t) = 0\}$ be the set of columns whose r-th or y-th row entry is zero.

If $t \notin S_\Delta \cup S_0$, then $z^t = z'^t$, and (10) implies that

$$\sum_{t\in S_\Delta \cup S_0} L(z^t) \geq \sum_{t\in S_\Delta \cup S_0} L(z'^t),$$

which, in turn, implies that

$$\begin{aligned}\sum_{t=1}^{n_c} L(z^t) \geq & \sum_{t\in S_\Delta \cup S_0} L(z^t) \\ \geq & \frac{1}{2}\sum_{t\in S_\Delta \cup S_0} (L(z^t)+L(z'^t)) \\ = & \frac{1}{2}\sum_{t\in S_\Delta} (L(z^t)+L(z'^t)) + \frac{1}{2}\sum_{t\in S_0} (L(z^t)+L(z'^t)).\end{aligned} \tag{11}$$

If $t \in S_\Delta$, then $z'^t = -z^t$ and $(L(-z^t) + L(z^t))/2 \geq L(0)$ due to convexity. If $t \in S_0$, then either $z^t = 0$ or $z'^t = 0$. Hence, we obtain that $L(z'^t) + L(z^t) \geq L(0)$, and the second term of (11) is at least $L(0)|S_0|/2$. Therefore,

$$\begin{aligned}\sum_{t=1}^{n_c} L(z^t) \geq & \frac{1}{2}\sum_{t\in S_\Delta}(L(z^t)+L(z'^t)) + \frac{1}{2}\sum_{t\in S_0}(L(z^t)+L(z'^t)) \\ \geq & L(0)\left(|S_\Delta| + \frac{|S_0|}{2}\right) \\ = & L(0)d(M(r,:),M(y,:)) \qquad (12) \\ \geq & \rho L(0),\end{aligned}$$

where (12) is obtained from the fact that

$$\begin{aligned} d(M(r_1,s),M(r_2,s)) &= \frac{1-sign(M(r_1,s)M(r_2,s))}{2} \\ &= \begin{cases} 0 & if \;\; M(r,t) = M(y,t) \wedge M(r,t) \neq 0 \wedge M(y,t) \neq 0 \\ \frac{1}{2} & if \;\; M(r,t) = 0 \vee M(y,t) = 0 \\ 1 & if \;\; M(r,t) \neq M(y,t) \wedge M(r,t) \neq 0 \wedge M(y,t) \neq 0. \end{cases}\end{aligned}$$

This completes the proof. ■

Remark 4 *From Proposition 3, it can be seen that misclassification on a test instance $(\mathbf{x}, y)$ implies that $\sum_{t=1}^{n_c} L(M(y,t)h^t(\mathbf{x})) \geq \rho L(0)$. In other words, the prediction codes are not required to be exactly the same as ground-truth codes for all the base classifications. As long as the loss is smaller than $\rho L(0)$, ECOC can rectify the error committed by some base classifiers, and is still able to make an accurate prediction. This error-correcting ability is very important, especially when the labeled data is insufficient in the target domain. This proposition holds for any convex margin-based loss function $L(z)$. In this paper, we use hinge loss defined in (4), then $L(0) = 1$.*

Theorem 5 *Let ϵ be the averaged loss of the target domain hypotheses $h_T^1, \ldots, h_T^{n_c}$ on the target test data $\{\{(\mathbf{x}_{T_i}, M(y_{T_i}, t))\}_{i=n_T+1}^{n_{Te}}\}_{t=1}^{n_c}$ with respect to the coding matrix $M \in \{-1, 0, +1\}^{c\times n_c}$, where c is the cardinality of the label set. Let the convex loss function be $L(z) = |1-z|_+$. The multi-class HDA generalization error of SHFR using sparse random encoding and loss-based decoding is then at most $\frac{\epsilon n_c}{\rho}$.*

Proof According to Proposition 3, the loss for any misclassified testing instance $\mathbf{x}_T$ satisfies

$$\rho L(0) \leq \sum_{t=1}^{n_c} L(M(y_T, t)h_T^t(\mathbf{x}_T)).$$

Let a be the number of incorrect predictions for a test sample of size n_{Te}, then we can obtain the following inequality,

$$a\rho L(0) \leq a\sum_{t=1}^{n_c} L(M(y_T,t)h_T^t(\mathbf{x}_T)) \leq n_{Te}\sum_{t=1}^{n_c} L(M(y_T,t)h_T^t(\mathbf{x}_T)). \qquad (13)$$

Then we have

$$a \leq \frac{n_{Te} \sum_{t=1}^{n_c} L(M(y_T, t) h_T^t(\mathbf{x}_T))}{\rho L(0)} = \frac{n_{Te} \epsilon n_c}{\rho}, \tag{14}$$

where $\epsilon = \frac{\sum_{t=1}^{n_c} L(M(y_T,t) h_T^t(\mathbf{x}_T))}{n_c}$ and $L(0) = |1 - 0|_+ = 1$.
Therefore, the generalization error rate is bounded by $\frac{\epsilon n_c}{\rho L(0)}$. The proof is completed. ■

Remark 6 *The above error bound is usually small in the setting studied in this paper for the following reasons. When sufficient classifiers are generated using ECOC and* $\mathbf{G}$ *satisfies the sparsity assumption, the reconstruction error is small and bounded according to the compressive sensing theory. On the other hand, the loss* $L(M(y_T, t) h_T^t(\mathbf{x}_T))$ *for some* $\{h_T^t\}$*s, which rely on* $\mathbf{G}$*, may be large due to there being limited target domain data. The data-independent term* n_c *is usually predefined and the minimum distance* ρ *is determined correspondingly, as* ρ *monotonically increases with* n_c *according to (8). Given the predefined ECOC coding matrix, as long as the averaged loss* ϵ *is small, according to Theorem 5, target classifier is still able to make a precise prediction. In other words, SHFR is only unable to achieve satisfactory performance in terms of classification accuracy if most or all of the binary classifiers generated using ECOC trained on the target domain data are poorly designed.*

In summary, we show that the multi-class HDA generalization error of SHFR based on the ECOC scheme has a linear relationship with the average loss over all generated binary classifiers when using random encoding and loss-based decoding methods. In addition to theoretical analysis, García-Pedrajas and Ortiz-Boyer (2011) conducted comprehensive experiments and concluded that ECOC and one-vs-one are superior to other multi-class classification schemes, and are the best choices for either powerful learners or simple learners. Furthermore, Montazer et al. (2012) claimed that, of several multi-class classification schemes, sparse ECOC is better than dense ECOC, while dense ECOC is better than one-vs-one, and one-vs-all is the worst. Ghani (2000) proved a theoretical bound that a randomly-construct binary matrix is not well row-separated with probability at most $1/c^4$, where c is the number of classes. Therefore, in our experiments, we adopt the sparse and random ECOC scheme for multi-class HDA problems.

4.3 Robust Transformation Learning using ECOC

The robustness of ECOC is another important motivation for using ECOC in SHFR. Recall that the error-correcting codes can be viewed as a compact form of voting, and a certain number of incorrect votes can be corrected through the corrected votes (Dietterich and Bakiri, 1995). Given a total number of T classifiers, the voting-based methods guarantee to make a correct decision as long as there are $\lfloor \frac{T}{2} + 1 \rfloor$ correct classifiers (Dietterich, 2000), where $\lfloor\ \rfloor$ is the *floor* function that returns the largest integer not greater than the input of the function. In other words, even though there are misclassifications due to incorrect base classifiers, good performance can still be achieved in terms of the final classification accuracy by using the ECOC scheme. It has been shown in previous research that although bit error is unavoidable in real-world applications, it is still possible to make correct decisions using the error-correcting codes with a sufficient number of good learners (Ghani, 2000).

This property is particularly important for the proposed multi-class HDA method, SHFR, since some of the learned binary classifiers in the target domain may not be accurate due to insufficient label information. SHFR with ECOC is able to learn a robust transformation matrix $\mathbf{G}$ even when some target binary classifiers are not precise. The robustness of SHFR with ECOC will be verified empirically in Section 6, where we demonstrate that even with inaccurate target classifiers, the class-invariant $\mathbf{G}$ learned by SHFR with the ECOC scheme can still dramatically enhance prediction accuracy for the target domain. By combining all the above theoretical analyses, we can conclude that the generalization error of our proposed method SHFR for multi-class HDA problems is small and bounded with a proper ECOC coding matrix design.

5. Efficient Batch Decoding for SHFR

In this section, we present an algorithm to solve the proposed optimization problem (3) of SHFR in detail. Recall that problem (3) contains d_T LASSO problems with nonnegative constraints, which can be solved by adapting existing algorithms like NNLS solvers (Cantarella and Piatek, 2004; Slawski and Hein, 2011) or LASSO solvers, e.g., APG (Toh and Yun, 2010) and OMP (Zhang, 2009). However, the time complexity of these methods is $O(n_c d_S)$ for each LASSO problem, which makes problem (3) intractable when the source domain data is of very high dimensionality and the number of LASSO problems d_T is very large.

Recall that each of the optimization problems (3) can be cast as the following problem

$$\begin{aligned} \min_{\mathbf{g}_i} \quad & \frac{1}{n_c}\|\mathbf{b}_i - \mathbf{D}\mathbf{g}_i\|_2^2 + \lambda_i\|\mathbf{g}_i\|_1, \\ s.t. \quad & \mathbf{g}_i \succeq \mathbf{0}, \end{aligned} \tag{15}$$

where $\mathbf{b}_i \in \mathbb{R}^{n_c \times 1}$ is the concatenated row vector containing $w_{T_i}^t$ for all the n_c tasks, and $\mathbf{D} = [\mathbf{w}_S^1\ \mathbf{w}_S^2\ \cdots\ \mathbf{w}_S^{n_c}]^\top \in \mathbb{R}^{n_c \times d_S}$. For simplicity and without loss of clarity, we drop the subscript i from $\mathbf{g}_i$ and $\mathbf{b}_i$ in our discussion. Given any $\mathbf{g}$, let

$$\boldsymbol{\xi} = \mathbf{b} - \mathbf{D}\mathbf{g} \tag{16}$$

be the residual. Problem (15) can then be equivalently reformulated as the following problem

$$\begin{aligned} \min_{\mathbf{g}\succeq\mathbf{0},\boldsymbol{\xi}} \quad & \lambda||\mathbf{g}||_1 + \frac{1}{2}||\boldsymbol{\xi}||^2, \\ s.t. \quad & \boldsymbol{\xi} = \mathbf{b} - \mathbf{D}\mathbf{g}. \end{aligned} \tag{17}$$

Solving this problem directly can be very expensive when the source domain data is of very high dimensionality. Following Tan et al. (2015a), we propose to solve problem (15) via a matching pursuit method, called matching pursuit LASSO (MPL), as shown in Algorithm 1.

In Algorithm 1, the residual $\boldsymbol{\xi}^0$ is initialized by $\boldsymbol{\xi}^0 = \mathbf{b}$ since $\mathbf{g}^0 = \mathbf{0}$. At the t-th step, we first choose a set of B components in $\mathbf{g}$ with the B largest values in $\mathbf{D}^\top\boldsymbol{\xi}_t$, and merge their indices $\mathcal{J}_t$ into the active set $\mathcal{I}_t$. We then solve a reduced problem whose scale is only $O(n_c|\mathcal{I}_t|)$:

$$\begin{aligned} \min_{\mathbf{g}\succeq\mathbf{0},\boldsymbol{\xi}} \quad & \lambda||\mathbf{g}||_1 + \frac{1}{2}||\boldsymbol{\xi}||^2, \\ s.t. \quad & \boldsymbol{\xi} = \mathbf{b} - \mathbf{D}\mathbf{g}, \mathbf{g}_{\mathcal{I}_t^c} = \mathbf{0}, \end{aligned} \tag{18}$$

Algorithm 1 Matching pursuit LASSO for solving the nonnegative LASSO problem.

Initialize the residual $\boldsymbol{\xi}^0 = \mathbf{b}$, $\mathcal{I}_0 = \emptyset$, and let $t = 1$.
1: Compute $\mathbf{q} = \mathbf{D}^\top \boldsymbol{\xi}^{t-1}$, choose the B largest q_j, and record their indices by $\mathcal{J}_t$.
2: Let $\mathcal{I}_t = \mathcal{I}_{t-1} \cup \mathcal{J}_t$.
3: Let $\mathbf{g}_{\mathcal{I}_t^c} = \mathbf{0}$, and solve the subproblem (18) to update $\mathbf{g}_{\mathcal{I}_t}$.
4: Compute $\boldsymbol{\xi}^t = \mathbf{b} - \mathbf{D}\mathbf{g}_{\mathcal{I}_t}$.
5: Terminate if the stopping condition is achieved. Otherwise, let $t = t + 1$ and go to step 1.

where $\mathcal{I}_t$ denotes the index set of the selected columns from $\mathbf{D}$ and $\mathcal{I}_t^c$ is the corresponding complementary set of $\mathcal{I}_t$. This subproblem can be solved efficiently using a projected proximal gradient (PG) method. Note that the number B in Step 1 is a small positive integer, e.g., $B = 1$. Due to the nonnegative constraints $\mathbf{g} \succeq 0$, we find the elements with the B largest value in $\mathbf{D}^\top \boldsymbol{\xi}$ rather than $|\mathbf{D}^\top \boldsymbol{\xi}|$.

Batch-mode MPL. In Algorithm 1, solving each subproblem has a complexity of $O(kn_c)$, where k denotes the degree of the sparsity, namely $k = |\mathcal{I}_t|$, while computing $\mathbf{D}^\top \boldsymbol{\xi}$ has a complexity of $O(d_S n_c)$. Note that problem (3) has d_T sub-problems as in (15). Therefore, computing $\mathbf{D}^\top \boldsymbol{\xi}$ will dominate the whole complexity of the decoding if both n_c and d_T are very large.

To address the above computational burden imposed by $\mathbf{D}^\top \boldsymbol{\xi}$, we propose a batch-mode MPL (BMPL) algorithm. Note that $\mathbf{D}^\top \boldsymbol{\xi} = \mathbf{D}^\top(\mathbf{b} - \mathbf{D}\mathbf{g}) = \mathbf{D}^\top \mathbf{b} - \mathbf{D}^\top \mathbf{D}\mathbf{g}$. We can pre-compute $\boldsymbol{\beta} = \mathbf{D}^\top \mathbf{b}$ and $\mathbf{Q} = \mathbf{D}^\top \mathbf{D}$, and store them in the memory. Since $\mathbf{g}_{\mathcal{I}_t^c} = \mathbf{0}$ for the t-th iteration, we have $\mathbf{D}^\top \boldsymbol{\xi}^t = \mathbf{D}^\top \mathbf{b} - [\mathbf{D}^\top \mathbf{D}_{\mathcal{I}_t}]\mathbf{g}_{\mathcal{I}_t} = \boldsymbol{\beta} - \mathbf{Q}_{\mathcal{I}_t}\mathbf{g}_{\mathcal{I}_t}$, where $\mathbf{Q}_{\mathcal{I}_t}$ denotes the columns of $\mathbf{Q}$ indexed by $\mathcal{I}_t$. The computation cost for $\mathbf{D}^\top \boldsymbol{\xi}$ is thus reduced to $O(kn_c)$, which saves considerable computation cost when $k \ll d_S$. Note that $\boldsymbol{\beta}$ and $\mathbf{Q}$ are shared by all the LASSO tasks, thus they only need to be calculated once. With this strategy, the overall computational cost of solving (3) when d_S is large is significantly reduced to $O(d_T n_c k)$ (Tan et al., 2015a,b).

5.1 Complexity Comparison

We use linear SVMs to build the base classifiers for the source and target domains, which are obtained using the Liblinear solver (Fan et al., 2008) and pre-trained offline. The learning of $\mathbf{G}$ is not related to the number of training instances. According to the analysis in Section 5, the computational cost of SHFR is $O(d_T n_c k)$, where the sparsity degree $k \ll d_S$. Compared to state-of-the-art HDA methods, our proposed method is much more efficient. The MOMAP method transforms the problem into c separate singular value decomposition (SVD) problems resulting in complexity of $O(4d_S^2 d_T + 8d_S d_T^2 + 9d_T^3)$. The HeMap method is based on standard eigen-decomposition, whose complexity is $O(n_S + n_T)^3$. The ARC-t method solves an optimization problem that contains $n_S n_T$ constraints by applying an alternating projection method (e.g., Bregman's algorithm (Censor and Zenios, 1997)), resulting in complexity of $d_S d_T n_S n_T$. The HFA method adopts an alternating projection method to solve a semidefinite program (SDP), where the transformation matrix to be learned is in $\mathbb{R}^{(n_S+n_T)\times(n_S+n_T)}$, resulting in time complexity bounded by $O(n_S + n_T)^3$. Therefore, ARC-t and HFA perform inefficiently when the data size is large. DAMA first constructs a series of combinatorial Laplacian matrices in $\mathbb{R}^{(d_S+d_T)\times(d_S+d_T)}$, and then solves a generalized eigen-decomposition problem of time complexity bounded by $O(d_S + d_T)^3$. DAMA is therefore

Table 1: Complexity comparison of HDA methods.

Methods	Complexity
MOMAP (Harel and Mannor, 2011)	$O(4d_S^2 d_T + 8d_S d_T^2 + 9d_T^3)$
HeMap (Shi et al., 2010)	$O(n_S + n_T)^3$
DAMA (Wang and Mahadevan, 2011)	$O(d_S + d_T)^3$
ARC-t (Kulis et al., 2011)	$O(d_S d_T n_S n_T)$
HFA (Duan et al., 2012)	$O(n_S + n_T)^3$
SHFR (our proposed method)	$O(d_T n_c k)$

computationally very expensive when the data dimensionality is high. The comparison in terms of time complexity between different methods is summarized in Table 1.

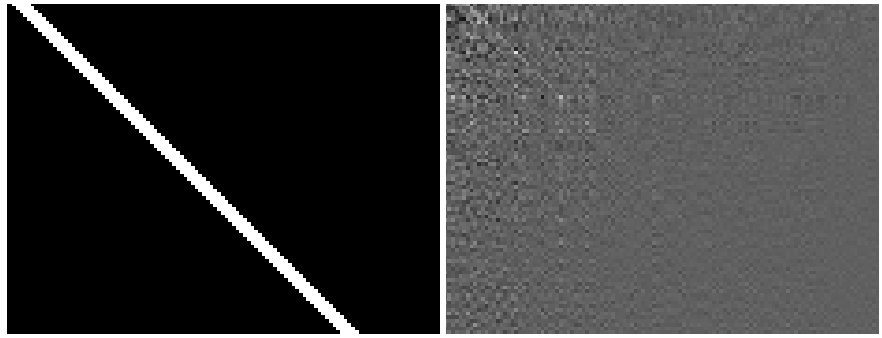
6. Experiments

In this section, we conduct experiments on both toy and real-world datasets to verify the effectiveness and efficiency of our proposed method SHFR for multi-class HDA. The parameter settings of our proposed method and other baseline methods are as follows. For ARC-t and HFA, which require the use of kernel functions to measure data similarity, we use the RBF kernel for learning the transformation. Parameter tuning is still an open research issue, as cross-validation is not applicable to HDA problems due to the limited size of labeled data in the target domain. We therefore tune the parameters of the comparison methods on a predefined range and report their best results on the test data. We use linear SVMs as the base classifiers, the regularization parameter C of which is cross-validated from the range of $\{0.01, 0.1, 1, 10, 100\}$ on the source domain data, and we use it for all the comparison methods. For our proposed method SHFR, we empirically set the maximum number of iterations to 20, $\lambda = 0.01$ and $B = 2$ in the BGMP algorithm, and generate the ECOC as long as possible for each dataset.

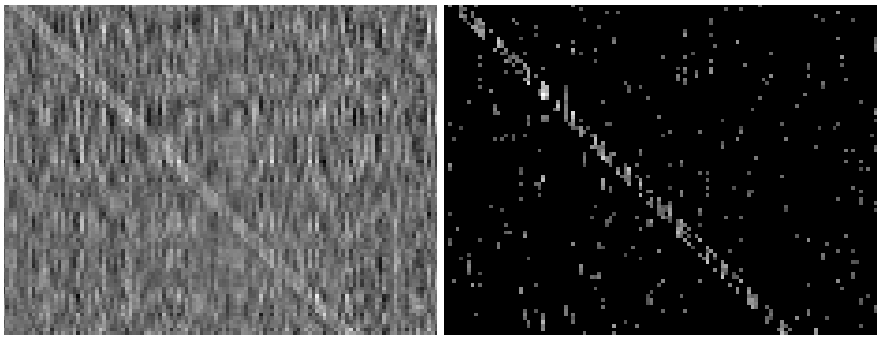
6.1 Experiments on Synthetic Datasets

We first compare the performance of different HDA methods in terms of recovering a ground-truth feature mapping $\mathbf{G}$ on a 20-class toy dataset. To generate the toy dataset, we first randomly generate 150 instances of 150 features for each class from different Gaussian distributions to form a source domain $\mathbf{X}_S \in \mathbb{R}^{150\times 3{,}000}$. We then construct the ground-truth sparse feature mapping $\mathbf{G} \in \mathbb{R}^{100\times 150}$ using the following method: for each row i, we set $\mathbf{G}_{ij} = 1/5$, where $j = i, i+1, \cdots, i+5$, and $\mathbf{G}_{ij} = 0$ otherwise. This generation of $\mathbf{G}$ implies that each target domain feature is represented by five source domain features. The ground-truth feature mapping is displayed in Figure 2(a), where the dark area represents the zero entries and the bright area denotes nonzero values of $\mathbf{G}$. Lastly, we construct the target domain data $\mathbf{X}_T \in \mathbb{R}^{100\times 3{,}000}$ by using $\mathbf{X}_T = \mathbf{G}\mathbf{X}_S$. When conducting the experiment, we randomly select five instances per class from the target domain data $\mathbf{X}_T$ as the labeled training data, and apply different HDA methods on them together with all 3,000 source domain labeled data to recover the feature mapping $\mathbf{G}$.

In this experiment, the HDA methods, DAMA and ARC-t, are adopted as the baselines. For ease of comparison, we present the recovered matrix $\mathbf{G}$ for each of the three methods in Figures 2(b)-2(d). From Figure 2(b), we observe that DAMA fails to recover the structure of $\mathbf{G}$, while ARC-t



(a) Ground Truth (b) DAMA



(c) ARC-t (d) SHFR

Figure 2: Illustration of the recovered feature mappings using different methods on the toy data. Black represents 0 and white represents 1.

shows better performance as can be seen in Figure 2(c). However, the $\mathbf{G}$s recovered by these two methods are not sparse. In contrast, SHFR can perfectly recover the sparse $\mathbf{G}$ with little noise as shown in Figure 2(d). These experimental results demonstrate that by explicitly adding sparsity constraints, SHFR is able to recover the feature mapping $\mathbf{G}$ more accurately.

6.2 Experiments on Real-world Datasets

We conduct experiments on three real-world datasets, Multilingual Reuters Collection, BBC Collection, and Cross-lingual Sentiment Dataset, to verify the effectiveness and efficiency of SHFR. The reported results are averaged over 10 independent data-split procedures.

6.2.1 DATASETS AND EXPERIMENTAL SETUP

Multilingual Reuters Collection[2] is a text dataset with over 11,000 news articles from six categories in five languages, i.e., English, French, German, Italian and Spanish, which are represented by a bag-of-words weighted by TF-IDF. Following the setting in Duan et al. (2012), we use Spanish as the target domain and the other four languages as source domains, which results in four multi-class HDA problems. For each class, we randomly select 100 instances from the source domain

2. http://multilingreuters.iit.nrc.ca/ReutersMultiLingualMultiView.htm

and 10 instances from the target domain for training. We randomly select 10,000 instances from the target domain as the test data. Note that the original data is of very high dimensionality, and the baseline methods cannot handle such high-dimensional features. To conduct comparison experiments, we first perform PCA with 60% energy preserved on the TF-IDF features. We then obtain 1,131 features for the the English documents, 1,230 features for the French documents, 1,417 features for the German documents, 1,041 features for the Italian documents, and 807 features for the Spanish documents. In contrast to the baseline methods, we use original features for SHFR since it can efficiently handle high-dimensional data.

BBC Collection[3] was collected for multi-view learning, and each instance is represented by three views. It was constructed from a single-view BBC corpora by splitting news articles into related "views" of text. We consider **View 3** as the target domain, and **View 1** and **View 2** as source domains. Similar to the pre-processing on the Reuters dataset, we perform PCA on the original data to reduce dimensions such that other baselines can be applied. The reduced dimensions for **View 1**, **View 2** and **View 3** are 203, 205 and 418, respectively. We randomly select 70% source domain instances and 10 target domain instances for each class for training. The remaining target domain instances are considered to be the test data.

Cross-lingual Sentiment Dataset[4] consists of Amazon product reviews of three product categories: books, DVDs and music. These reviews are written in four languages: English, German, French, and Japanese. We treat English reviews as the source domain data and the other language reviews as the target data domain data. After performing PCA, the reviews are of 715, 929, 964, and 874 features for English, German, French and Japanese, respectively. We randomly select 1,500 source domain instances and 10 target domain instances per class for training, and use the rest 5,970 target domain instances for testing.

6.2.2 OVERALL COMPARISON RESULTS

Comparison results between SHFR and other baselines on the three real-world datasets are reported in Tables 2-4. From the tables, we observe that the SVMs conducted on a small number of target domain data only, denoted by SVM-T, using either one-vs-one or one-vs-all strategy, perform the worst on average. Moreover, the results of SVM-T using one-vs-one and one-vs-all, respectively, are not consistent. For instance, on the BBC dataset in Table 3, SVM-T using the one-vs-all strategy performs much better than SVM-T using the one-vs-one strategy, while on the sentiment dataset in Table 4, SVM-T using the one-vs-one strategy performs much better than SVM-T using the one-vs-all strategy. The reason is that the size of the labeled training data is too limited to train a precise and stable classifier in the target domain. Compared to one-vs-all and one-vs-one, SVM-T with ECOC shows much more stable performance and superior results on most problems, which is because of the error correcting ability of ECOC when the labeled data is limited. The performance in terms of classification accuracy of the HDA baseline methods, DAMA, ARC-t and HFA, are comparable on the three datasets except for the BBC dataset, where DAMA performs much worse than the other two methods. This may be because the performance of DAMA is sensitive to the intrinsic manifold structure of the data. If the manifold assumption does not hold on the data, the performance of DAMA drops significantly. Our proposed SHFR method using either the one-vs-one or ECOC scheme performs best on the three datasets. SHFR can further improve the performance

3. http://mlg.ucd.ie/datasets/segment.html
4. http://www.uni-weimar.de/cms/medien/webis/research/corpora/corpus-webis-cls-10.html

Table 2: Multilingual Reuters Collection: comparison results in terms of classification acc (%). Results of SHFR are significantly better than the other baselines, judged by the t-test with a significance level at 0.05.

Source Domain	SVM-T (1 vs all)	SVM-T (1 vs 1)	SVM-T (ECOC)	DAMA	ARC-t	HFA	SHFR (1vs1)	SHFR (ECOC)
English	65.40±4.45	65.21±7.85	68.01±2.48	63.42±2.62	66.72±2.27	68.82±1.68	69.45±1.56	**72.82±1.08**
French				65.12±1.28	67.67±2.05	68.14±1.71	70.77±1.57	**74.01±1.25**
German				66.98±2.45	68.15±1.72	68.42±1.57	71.62±1.69	**74.15±1.14**
Italian				67.56±2.23	67.47±2.32	69.59±2.51	70.75±1.54	**73.35±1.31**

Table 3: BBC Collection: comparison results in terms of classification acc (%). Results of SHFR are significantly better than the other baselines, judged by the t-test with a significance level at 0.05.

Source Domain	SVM-T (1 vs all)	SVM-T (1 vs 1)	SVM-T (ECOC)	DAMA	ARC-t	HFA	SHFR (1vs1)	SHFR (ECOC)
View 1	73.35±4.98	68.78±11.7	76.45±5.47	67.42±2.25	75.78±2.69	72.45±10.65	89.81±1.20	**90.57±1.49**
View 2				66.87±1.73	74.53±2.48	73.75±8.06	88.92±1.67	**91.85±0.96**

Table 4: Cross-lingual Sentiment Dataset: comparison results in terms of classification acc. (%). Results of SHFR are significantly better than the other baselines, judged by the t-test with a significance level at 0.05.

Target Domain	SVM-T (1 vs all)	SVM-T (1 vs 1)	SVM-T (ECOC)	DAMA	ARC-t	HFA	SHFR (1vs1)	SHFR (ECOC)
French	48.40±3.45	58.30±5.01	57.45±3.17	55.18±3.46	53.46±5.18	56.46±3.42	60.80±3.08	**62.12±2.61**
German	49.78±5.12	61.62±6.31	62.18±4.51	56.60±3.72	57.29±2.17	55.14±3.15	63.85±3.46	**65.57±2.32**
Japanese	48.25±6.34	56.13±4.81	57.27±2.61	53.56±2.67	55.75±3.02	55.02±4.67	59.67±3.75	**62.58±2.86**

in terms of classification accuracy, compared to one-vs-one, using the ECOC scheme. The superior performance benefits from both the global transformation of $\mathbf{G}$ and the error correcting ability of ECOC. As discussed in Section 4.1, the recovered feature mapping $\mathbf{G}$ tends to be more accurate with more constructed binary tasks.

6.3 Impact on Training Sample Size of the Target Domain

We verify the impact of the size of labeled training sample in the target domain to the overall HDA performance in terms of classification accuracy. We vary the number of target domain training instances from five to 20. In this experiment, we only report the results on the Reuters dataset, where we use English as the source domain and Spanish as the target domain. The experimental results are shown in Figure 3. From the figure, we observe that SHFR consistently outperforms the baseline methods under different numbers of labeled training instance in the target domain. In particular, SHFR shows significantly better performance than the baseline methods when the size of the target domain labeled data is smaller than 10.

Based on Theorem 5, the overall generalization error of SHFR depends on the accuracy of the binary classifiers generated based on ECOC. To empirically demonstrate this point, we report the

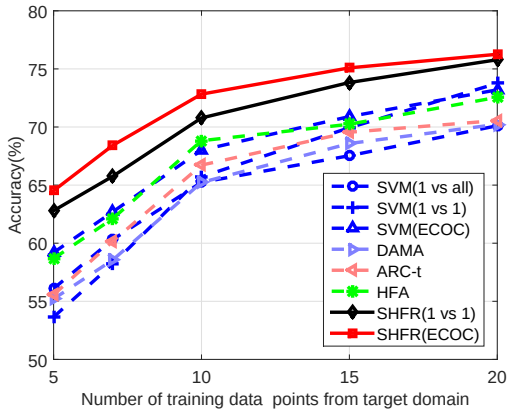
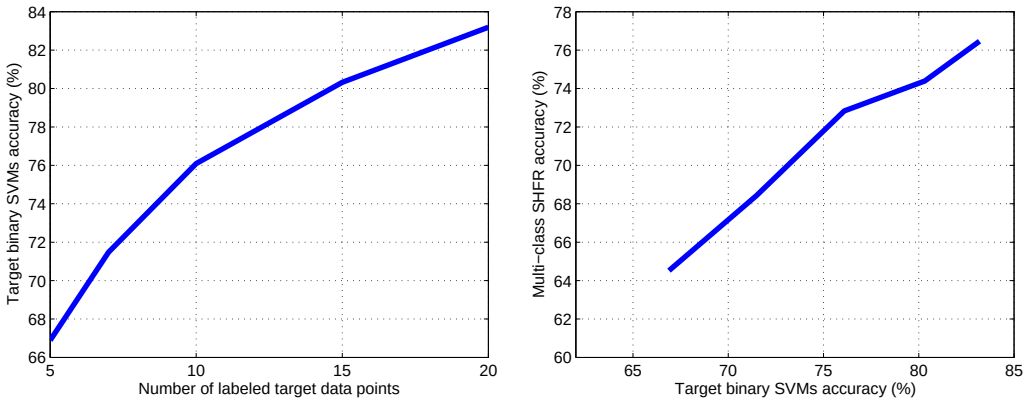


Figure 3: Comparison of different HDA methods under varying size of target labeled data.



(a) Averaged binary accuracy v.s. target data size. (b) Multiclass accuracy v.s. averaged binary accuracy.

Figure 4: Relationships between binary base classifiers and multi-class SHFR (ECOC).

averaged accuracy of all binary classifiers generated based on ECOC on the Reuters dataset under varying numbers of labeled training instances in the target domain in Figure 4(a). We also report the overall multi-class prediction accuracy versus the averaged accuracy of all binary classifiers in Figure 4(b). From the figures, we see that more labeled target data leads to more precise target binary classifiers, and thus leads to improvement in multi-class HDA classification in terms of accuracy. In particular, we observe from Figure 4(b) that the multi-class classification accuracy is proportional to the averaged binary task accuracy. The reason is that more precise target binary classifiers generally induce better alignment between the target and source classifiers.

6.4 Impact on Dimensionality of the Target Domain

In this experiment, we aim to compare the performance of different HDA methods under varying dimensionality of the target domain data. Here, we only report experimental results on the English-Spanish HDA task on the Reuters dataset. We first select 5,000 of the most-frequent features in the English domain as the source domain. We vary the dimensionality of the Spanish domain, i.e.,

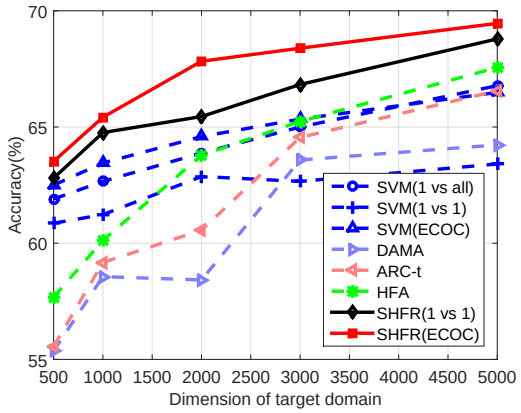


Figure 5: Comparison of different HDA methods under varying dimensions of target labeled data.

Table 5: Comparison in correlation $\sum_{ji} |\rho_{ji}|$.

Coding Scheme	English	French	German	Italian
1 vs all	0.8532 ± 0.1358	0.7557 ± 0.1289	0.2738 ± 0.0245	0.9331 ± 0.5701
1 vs 1	0.3374 ± 0.0290	0.2509 ± 0.0576	0.1168 ± 0.0110	0.3322 ± 0.1679
ECOC	0.1349 ± 0.0142	0.1478 ± 0.0225	0.0732 ± 0.0145	0.1725 ± 0.1047

the target domain, in the range of $\{500, 1000, 2000, 3000, 5000\}$ by selecting the most-frequent features of this domain. The results are shown in the Figure 5, from which we observe that SHFR consistently outperforms other baselines, especially when the dimensionality of the target domain data is low.

6.5 Incoherent Dictionary $\mathbf{D}$ Construction

In this experiment, we adopt the summation of absolute value of all pairs of columns in the dictionary $\mathbf{D}$ to evaluate $\mathbf{D}$'s coherence degree, i.e., $\sum_{ji} |\rho_{ji}|$. In other words, the smaller the value of $\sum_{ji} |\rho_{ji}|$ is, the more incoherent the dictionary is (Zhang and Huang, 2008). The results of different coding schemes on the Reuters dataset are summarized in Table 5. From the table, we observe that ECOC can be used to significantly reduce the coherence degree to enable better construction of the dictionary for the formulated compressed sensing problem.

6.6 Error Correction through Learning a Global $\mathbf{G}$

As discussed in Section 4, the weight vectors of the binary classifiers constructed in the target domain, i.e., $\{\mathbf{w}_T\}$s, may be unreliable due to the lack of target labeled data, which may affect the estimation of $\mathbf{G}$. To verify that SHFR can correct the bias of some binary classifiers, we conduct comparison experiments between SVM-T and SHFR in terms of classification accuracy on each binary task on the Reuters dataset. The results are shown in Table 6, where each column corresponds to a binary task, indexed by $k \in \{1, \cdots, 15\}$. From the table, we observe that the predictions of SVM-T on some binary tasks are inaccurate due to the limited number of labeled data. The accuracy is even below 50% (numbers in boldface). However, by learning the transformation $\mathbf{G}$ jointly among all the binary tasks, we are able to reduce the bias of the weak binary classifiers,

Table 6: Comparison results on each binary task in terms of classification accuracy (%).

Binary Classifiers	1	2	3	4	5	6	7	8
SVM-T	74.42	67.98	85.67	82.53	**43.32**	75.95	66.30	73.92
SHFR	65.39	84.81	82.45	94.28	80.76	76.75	79.74	84.63
Difference	-9.03	16.82	-3.22	11.75	37.44	0.80	13.44	10.70
Binary Classifiers	9	10	11	12	13	14	15	
SVM-T	**42.78**	76.56	78.32	65.43	72.74	**46.56**	**49.71**	
SHFR	84.45	70.07	90.35	81.35	83.97	80.75	86.43	
Difference	41.67	-6.49	12.03	15.92	11.23	34.20	36.72	

and increase accuracy by more than 30% on average. Experiments on the other two datasets return similar results, where the performance in terms of accuracy of the binary classifiers obtained through $\mathbf{G}$ is increased by 26.53% on the BBC dataset and 3.01% on the Sentiment dataset on average, compared to the results generated by SVM-T.

The experimental results in Table 6 also verify why SHFR outperforms ARC-t in the experiments shown in Tables 2-4 and Figure 3. ARC-t aims to align the target data with the source data via a transformation $\mathbf{G}$ that is learned from similarity and dissimilarity constraints. The constraints are constructed from a large number of source domain labeled data and a few target domain labeled data. In the multi-class setting, the number of similarity constraints to be constructed when the number of classes is large is much smaller than the number of dissimilarity constraints due to the limited number of target domain labeled data. In contrast to ARC-t, SHFR tries to align the target classifiers with the source classifiers using transformation $\mathbf{G}$, which is shared by all the induced binary tasks or classifiers. In other words, $\mathbf{G}$ is estimated through all classifiers. By borrowing the idea from multi-task feature learning which jointly optimizes all classifiers to learn a global feature transformation $\mathbf{G}$, is still possible to estimate a stable and precise $\mathbf{G}$ for multi-class HDA. Furthermore, as discussed in Section 4.3, ECOC with a well designed coding matrix is powerful for correcting error, as shown in Table 6.

6.7 Impact of Dichotomizer Size on Classification Error

As proven in Section 4, the generalization error of SHFR for multi-class HDA depends on the averaged loss over all the tasks, which relies on two factors: the number of classifiers (or measurements) and the degree of sparsity on $\mathbf{G}$. When $\mathbf{G}$ is sparse and the constructed binary classifiers are sufficient, it is possible to recover a precise $\mathbf{G}$ by using the dictionary constructed by $\mathbf{w}_S$. To demonstrate how the generalization error of SHFR changes under varying numbers of classifiers or dichotomizers, we conduct experiments on the Reuters dataset.

Experimental results are showed in Figure 6. From this figure we observe that when more dichotomizers are constructed, the predictions are more accurate in the target domain. This verifies that having more dichotomizers provides more information to recover the feature mapping $\mathbf{G}$. Furthermore, the standard deviation decreases with the increasing number of dichotomizers; however, as observed from the figure, multi-class accuracy no longer increases when the number of classifiers reaches 31. There are two reasons for this observation: 1) the redundancy in information among the constructed binary classifiers prevents improvement in estimating $\mathbf{G}$, and 2) according to García-Pedrajas and Ortiz-Boyer (2011), when there are many dichotomizers, the minimum distance in ECOC becomes small, which decreases the error correction ability. In general, SHFR obtains bet-

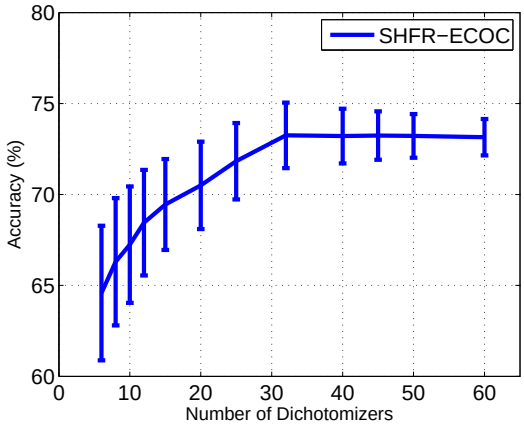


Figure 6: Accuracy vs dichotomizer (or task) size.

Table 7: Impact of Nonnegative Constraints and L_1 Norm Regularization: comparison between the results of different solvers in terms of classification accuracy (%). Results of SHFR are significantly better than the other baselines except for the cases denoted with *, judged by the t-test with a significance level at 0.05.

Dataset	Domain	SHFR-LASSO	SHFR-NNLS	SHFR
Reuters	English	67.08±1.63	71.47±1.56	**72.82±1.08**
	French	67.17±1.26	72.08±1.19	**74.01±1.25**
	German	69.56±1.28	73.62±1.75*	**74.15±1.14**
	Italian	68.19±2.51	71.98±1.78	**73.35±1.31**
BBC	View 1	84.72±2.18	88.66±1.29	**90.57±1.49**
	View 2	87.06±2.01	90.25±1.43	**91.85±0.96**
Sentiment	French	56.12±2.30	60.17±2.05	**62.12±2.61**
	German	58.83±2.24	63.47±1.49	**65.57±2.32**
	Japanese	56.72±1.81	61.51 ±1.76*	**62.58±2.86**

ter and more stable performance in terms of classification accuracy with an increasing number of dichotomizers.

6.8 Impacts of Nonnegative Constraints and ℓ_1 Norm Regularization

As discussed in Section 4, similar to the motivation of NMF, the nonnegative constraints on $\mathbf{G}$ are to approximate the "input vectors" (i.e., target features) by nonnegative linear combinations of nonnegative "basis vectors" (i.e., source features). The idea of nonnegative linear relationship can also be found in multiple kernel learning, where the target kernel is approximated by the nonnegative linear combination of base kernels. To verify the impact of the nonnegative constraints on $\mathbf{G}$, we add a baseline for comparison by moving the nonnegative constraints from the the optimization problem of SHFR, which is denoted by SHFR-LASSO. In this case, the model is reduced to a typical lasso problem.

Regarding the ℓ_1 norm regularization term in SHFR, note that some studies in the literature have shown that Non-Negative Least Squares (NNLS) can indeed achieve sparse recovery of nonnegative signals in a noiseless setting (Bruckstein et al., 2008; Donoho and Tanner, 2010; Wang and Tang, 2009; Wang et al., 2011). However, the noiseless assumption does not hold in practice. As mentioned by Slawski et al. (2013), apart from sign constraints, NNLS only consists of a fitting term, and thus is prone to overfitting, especially when the training data is high-dimensional. The ℓ_1-norm regularizer is therefore necessary to prevent over-adaptation to noisy data and enforce desired structural properties of the solution, which is similar to the idea of adding sparsity to the vector of coefficients of a predictive model (Slawski et al., 2013). In our proposed optimization problem (1), it is insufficient to achieve desirable solution of $\mathbf{G}$ without the ℓ_1-norm term in the objective since both $\mathbf{w}_S^t$ and $\mathbf{w}_T^t$ are of high dimensionality, and some of $\{\mathbf{w}_T^t\}$'s are not precise. As stated in Theorem 1 in Slawski and Hein (2011), exact sparsity, i.e., ℓ_1-norm, is not needed when the matrix satisfies the self-regularizing property, which is not applicable to our optimization problem as $\mathbf{G}$ does not have this property. Using the ℓ_1-norm ensures that a more sparse solution can be achieved by using nonnegative constraints. To verify the impact of the ℓ_1-norm on $\mathbf{G}$, we add another baseline for comparison by moving the ℓ_1-norm from the the optimization problem of SHFR, which is denoted by SHFR-NNLS. In this case, the model is reduced to an NNLS problem. In this paper, we adopt the NNLS solver proposed by Cantarella and Piatek (2004) to solve it.[5]

The results on the three multilingual datasets are summarized in Table 7. We observe from the table that SHFR-LASSO performs the worst and SHFR-NNLS enhances its performances through the nonnegative constraint. This is probably because the feature mapping $\mathbf{G}$ is learned globally and the features are naturally nonnegative for text classification. As SHFR explicitly imposes both nonnegativity and sparsity in learning the transformation $\mathbf{G}$, it can further enhance the performance compared to SHFR-NNLS by avoiding over adapting the noise in the high-dimensional data.

6.9 Experiments on Removing Negative Alignment

We have observed from Table 6 that the accuracy of SHFR for some constructed binary tasks is slightly less than the accuracy of SVM-T after learning and using the transformation $\mathbf{G}$, which can be referred to as *negative transfer* (Pan and Yang, 2010). The problem of negative transfer in SHFR mainly occurs for the following two reasons:

1. *Negative alignment* between the source and target classifiers, which happens when the classification accuracy of the corresponding target classifier is below 50%. This is because some initial target classifiers are less accurate due to the limited number of target domain labeled data.

2. The transformation $\mathbf{G}$ is class-invariant, therefore $\mathbf{G}$ is enforced to generalize well on all classifiers. In this case, $\mathbf{G}$ may correct the bias of some weak classifiers, but may also smooth the strength of good classifiers.

In this experiment, we study how the performance of SHFR is affected by removing negative alignments, and how SHFR performs if only good target classifiers whose accuracy is above 50%

5. In practice, the nonnegative lasso problem has been well studied and applied to many applications such as dimensionality reduction (Wang and Ye, 2014) and index tracking (Wu et al., 2014). Nevertheless, how to solve the nonnegative lasso problem is not the focus of our paper. In this paper, we address it by modifying the algorithm proposed by Tan et al. (2015a,b), which has demonstrated superior performance over many state-of-the-art solvers.

Table 8: Comparison results on each binary task after removing negative alignment in terms of classification accuracy (%).

Binary Classifiers	1	2	3	4	5	6	7	8
SVM-T	74.42	67.98	85.67	82.53	**43.32**	75.95	66.30	73.92
SHFR-RN	85.39	83.26	86.45	92.43	83.79	78.19	77.45	81.28
Difference	10.97	15.28	0.87	9.90	40.47	2.24	11.15	7.36
Binary Classifiers	9	10	11	12	13	14	15	
SVM-T	**42.78**	76.56	78.32	65.43	72.74	**46.56**	**49.71**	
SHFR-RN	84.15	78.45	92.14	82.12	84.58	82.72	88.61	
Difference	41.37	1.89	13.82	16.69	11.84	36.16	38.90	

are used. We investigate whether dropping negative alignments improves accuracy on all the binary tasks as well as the final multi-class HDA classification problem. Experimental results are shown in Table 8 and Figure 7. Compared to the results in Table 6, we discover from Table 8 that by removing the negative alignments we can improve all binary classification performances in terms of accuracy, and thus avoid negative transfers. We sort the binary tasks created from ECOC according to the accuracy on the test data and remove the poorly designed tasks. From Figure 7, we observe that when the number of dichotomizers is small (e.g., less than 30), SHFR with negative alignment removed achieves better performance in terms of multi-class classification accuracy. However, in practice, it is difficult to evaluate target classifiers due to the limit number of target domain labeled data. Interestingly, we can also find from Figure 7 that when the number of dichotomizers or tasks is sufficient, the classification accuracy of SHFR with or without removing negative alignments converges to the same value. This is because sub-class partition in ECOC often leads to higher binary accuracy when creating many dichotomizers or tasks. In this case, there are fewer binary target classifiers whose accuracy is below 50%. Therefore, we can conclude that SHFR with negative alignments removed enhances the performance in terms of classification accuracy when there are insufficient classifiers. In the multi-class HDA setting, however, the difficulty in alignment evaluation makes this post-processing impractical. Fortunately, SHFR can achieve the same performance as SHFR-RNs when there are sufficient dichotomizers or tasks, which makes SHFR more practicable for the multi-class HDA problem.

6.10 Training Time Comparison

Lastly, to verify the computational efficiency of SHFR, we conduct two experiments to compare the training time in learning the feature mapping between SHFR and other baselines under varying numbers of data dimensions and data instances. These experiments are conducted on a toy dataset, where the dimensions of the source and target data are constructed to be the same. In the first experiment, shown in Figure 8(a), we fix the number of training data (including both source and target domain data) to 3,000 and vary the data dimensions in the range of $\{100, 200, 500, 1000, 2000, 5000\}$. We observe that SHFR performs faster than all the baselines, and its training time increases slowly with the number of data dimensions. This is because the time complexity of SHFR has a linear relationship with the target domain dimensions. We can also see that the training time of DAMA increases dramatically when the number of data dimensions increases. In contrast, the training time of HFA and ARC-t does not increase much when the number of data dimensions increases. This

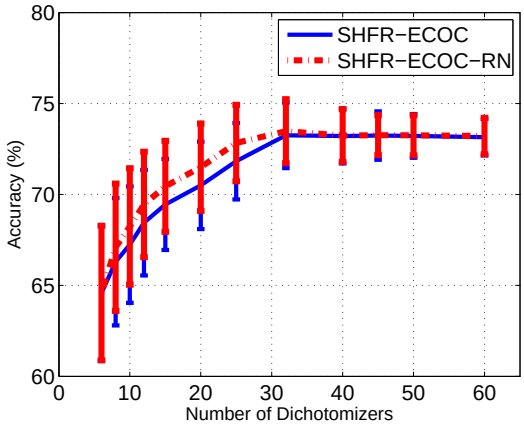
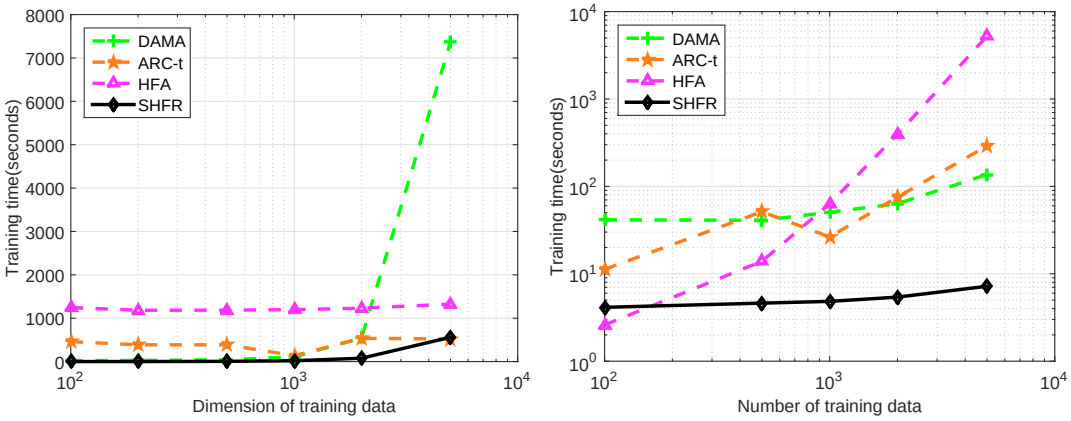


Figure 7: Accuracy v.s. dichotomizers size on removing tasks with negative alignment.



(a) Training time v.s. data dimensions

(b) Training time v.s. data size

Figure 8: Time complexity comparison

is because these two methods adopt the kernel technique such that the computational time depends on the number of data instances instead of the data dimensions. In the second experiment, shown in Figure 8(b), we fix the data dimension to 1,000 and vary the number of training instances in the range of $\{100, 500, 1000, 2000, 5000\}$. From the figure, we observe that the training time of SHFR changes slightly when the number of training instances increases. However, the training time of ARC-t and HFA increases polynomially when the number of training instances increases. These two experiments verify the superiority of SHFR in computational efficiency compared with the baseline methods.

7. Conclusion and Future Work

In this paper, we propose a **sparse heterogeneous feature representation** (SHFR) method for learning sparse transformation between heterogeneous features for HDA by exploring the common underlying structures of multiple classes between the source and target domains for multi-class

HDA. SHFR encodes the sparsity and class-invariance properties in learning the feature mapping. Learning feature mapping can be cast as a Compressed Sensing (CS) problem for SHFR. Based on the CS theory, we show that the number of binary learning tasks affects the multi-class HDA performance. In addition, the proposed method has superior scalability over other methods. Extensive experiments demonstrate the effectiveness, efficiency, and stability of SHFR.

In future, we would like to investigate how to extend the proposed method by exploring the nonlinear feature transformation between image and text feature representations.

Acknowledgments

We would like to thank the action editor Shie Mannor and anonymous reviewers for their valuable comments and constructive suggestions that greatly contributed to improving the final version of the paper.

Joey Tianyi Zhou was partially supported by programmatic grant no. A1687b0033, A18A1b0045 from the Singapore government's Research, Innovation and Enterprise 2020 plan (Advanced Manufacturing and Engineering domain). Ivor W. Tsang acknowledges the partial funding from the ARC Future Fellowship FT130100746, ARC grant LP150100671, and DP180100106. Sinno J. Pan thanks the support from NTU Singapore Nanyang Assistant Professorship (NAP) grant M4081532.020, Singapore MOE AcRF Tier-2 grant MOE2016-T2-2-060, and the Data Science & Artificial Intelligence Research Centre at NTU Singapore. Mingkui Tan was partially supported by National Natural Science Foundation of China (NSFC) 61602185, Program for Guangdong Introducing Innovative and Enterpreneurial Teams 2017ZT07X183, Guangdong Provincial Scientific and Technological Funds under Grants 2018B010107001.

References

Erin L. Allwein, Robert E. Schapire, and Yoram Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *J. Mach. Learn. Res.*, 1:113–141, September 2001.

Rie K. Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. Mach. Learn. Res.*, 6:1817–1853, 2005.

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *NIPS*, pages 41–048. MIT Press, 2007.

Richard Baraniuk, Mark Davenport, Ronald DeVore, and Michael Wakin. A simple proof of the restricted isometry property for random matrices. *Constr. Approx.*, 28(3):253–263, 2008.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *NIPS*, pages 137–144, 2006.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175, May 2010.

John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *EMNLP*, pages 120–128, 2006.

John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman. Learning bounds for domain adaptation. In *NIPS*, 2007a.

John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, 2007b.

Alfred M Bruckstein, Michael Elad, and Michael Zibulevsky. On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations. *IEEE Trans. Inform. Theory*, 54(11): 4813–4820, 2008.

Emmanuel J. Candès, Justin K. Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Comm. Pure Appl. Math.*, 59(8):1207–1223, August 2006.

Jason Cantarella and Michael Piatek. TSNNLS: A solver for large sparse least squares problems with non-negative variables. *arXiv preprint cs/0408029*, 2004.

Yair Censor and Stavros A. Zenios. *Parallel Optimization: Theory, Algorithms and Applications*. Oxford University Press, 1997.

Wenyuan Dai, Yuqiang Chen, Gui-Rong Xue, Qiang Yang, and Yong Yu. Translated learning: Transfer learning across different feature spaces. In *NIPS*, pages 353–360, 2008.

Hal Daumé III. Frustratingly easy domain adaptation. In *ACL*, pages 256–263. ACL, June 2007.

Thomas G. Dietterich. Ensemble methods in machine learning. In *MCS*, pages 1–15. Springer-Verlag, 2000.

Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *J. Artif. Intell. Res.*, 2:263–286, 1995.

Jeff Donahue, Judy Hoffman, Erik Rodner, Kate Saenko, and Trevor Darrell. Semi-supervised domain adaptation with instance constraints. In *CVPR*, 2013.

David L. Donoho. Compressed sensing. *IEEE Trans. Inform. Theory*, 52:1289–1306, 2006.

David L Donoho and Jared Tanner. Counting the faces of randomly-projected hypercubes and orthants, with applications. *Discrete & Comput. Geom.*, 43(3):522–541, 2010.

Lixin Duan, Ivor W. Tsang, Dong Xu, and Tat-Seng Chua. Domain adaptation from multiple sources via auxiliary classifiers. In *ICML*, pages 289–296, 2009.

Lixin Duan, Dong Xu, and Ivor W. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *ICML*, 2012.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008.

Nicolás García-Pedrajas and Domingo Ortiz-Boyer. An empirical study of binary classifier fusion methods for multiclass classification. *Inf. Fusion*, 12(2):111–130, April 2011.

Rayid Ghani. Using error-correcting codes for text classification. In *ICML*, pages 303–310, 2000.

Mehmet Gönen and Ethem Alpaydın. Multiple kernel learning algorithms. *J. Mach. Learn. Res.*, 12:2211–2268, 2011.

Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, pages 2066–2073. IEEE, 2012.

Boqing Gong, Kristen Grauman, and Fei Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *ICML*, pages 222–230, 2013.

Maayan Harel and Shie Mannor. Learning from multiple outlooks. In *ICML*, pages 401–408, 2011.

Judy Hoffman, Erik Rodner, Jeff Donahue, Kate Saenko, and Trevor Darrell. Efficient learning of domain-invariant image representations. In *ICLR*, 2013.

Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. LSDA: Large scale detection through adaptation. In *NIPS*, pages 3536–3544, 2014.

Jiayuan Huang, Alexander J. Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Schölkopf. Correcting sample selection bias by unlabeled data. In *NIPS*, pages 601–608, 2007.

Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in NLP. In *ACL*, pages 264–271. ACL, 2007.

Brian Kulis, Kate Saenko, and Trevor Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, pages 1785–1792, 2011.

G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *J. Mach. Learn. Res.*, 5:27–72, 2004.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2001.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *COLT*, 2009.

Gholam Ali Montazer, Sergio Escalera, et al. Error correcting output codes for multiclass classification: Application to two image vision problems. In *AISP*, pages 508–513. IEEE, 2012.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359, October 2010. ISSN 1041-4347.

Sinno Jialin Pan, James T. Kwok, and Qiang Yang. Transfer learning via dimensionality reduction. In *AAAI*, pages 677–682, July 2008.

Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Chen Zheng. Cross-domain sentiment classification via spectral feature alignment. In *WWW*, pages 751–760. ACM, 2010.

Sinno Jialin Pan, Ivor W. Tsang, James T. Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Trans. Neural Netw.*, 22(2):199–210, 2011.

Peter Prettenhofer and Benno Stein. Cross-language text classification using structural correspondence learning. In *ACL*, pages 1118–1127, 2010.

Guojun Qi, Charu C. Aggarwal, and Thomas S. Huang. Towards semantic knowledge propagation from text corpus to web images. In *WWW*, pages 297–306, 2011.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, pages 213–226. Springer-Verlag, 2010.

Chun-Wei Seah, Ivor Wai-Hung Tsang, and Yew-Soon Ong. Healing sample selection bias by source classifier selection. In *ICDM*, pages 577–586, 2011.

Xiaoxiao Shi, Qi Liu, Wei Fan, Philip S. Yu, and Ruixin Zhu. Transfer learning on heterogenous feature spaces via spectral transformation. In *ICDM*, pages 1049–1054, 2010.

Martin Slawski and Matthias Hein. Sparse recovery by thresholded non-negative least squares. In *NIPS*, pages 1926–1934, 2011.

Martin Slawski, Matthias Hein, et al. Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. *Electron. J. Stat.*, 7:3004–3056, 2013.

Mingkui Tan, Ivor W. Tsang, and Li Wang. Matching pursuit LASSO part I: Sparse recovery over big dictionary. *IEEE Trans. Signal Processing*, 63(3):727–741, 2015a.

Mingkui Tan, Ivor W. Tsang, and Li Wang. Matching pursuit LASSO part II: Applications and sparse recovery over batch signals. *IEEE Trans. Signal Processing*, 63(3):742–753, 2015b.

Kim-Chuan Toh and Sangwoon Yun. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pac. J. Optim.*, 6(615-640):15, 2010.

Chang Wang and Sridhar Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *IJCAI*, pages 1541–1546, 2011.

Jie Wang and Jieping Ye. Two-layer feature reduction for sparse-group lasso via decomposition of convex sets. In *NIPS*, pages 2132–2140, 2014.

Meng Wang and Ao Tang. Conditions for a unique non-negative solution to an underdetermined system. In *Allerton*, pages 301–307. IEEE, 2009.

Meng Wang, Weiyu Xu, and Ao Tang. A unique nonnegative solution to an underdetermined system: From vectors to matrices. *IEEE Trans. Signal Processing*, 59(3):1007–1016, 2011.

Lan Wu, Yuehan Yang, and Hanzhong Liu. Nonnegative-lasso and application in index tracking. *Comput. Stat. Data An.*, 70:116–126, 2014.

Qiang Yang, Yuqiang Chen, Gui-Rong Xue, Wenyuan Dai, and Yong Yu. Heterogeneous transfer learning for image clustering via the socialweb. In *ACL/IJCNLP*, pages 1–9, 2009.

Ming Yuan and Yi Lin. On the non-negative garrotte estimator. *J. Roy. Stat. Soc. B*, 69(2):143–161, 2007.

Cun-Hui Zhang and Jian Huang. The sparsity and bias of the LASSO selection in high-dimensional linear regression. *Ann. Statist.*, 36(4):1567–1594, 2008.

Tong Zhang. On the consistency of feature selection using greedy least squares regression. *J. Mach. Learn. Res.*, 10:555–568, 2009.

Joey Tianyi Zhou, Ivor W Tsang, Shen-Shyang Ho, and Klaus-Robert Müller. N-ary decomposition for multi-class classification. *Mach. Learn.*, pages 1–22.

Joey Tianyi Zhou, Sinno Jialin Pan, Ivor W. Tsang, and Yan Yan. Hybrid heterogeneous transfer learning through deep learning. In *AAAI*, pages 2213–2220, 2014.

Joey Tianyi Zhou, Sinno Jialin Pan, Ivor W. Tsang, and Shen-Shyang Ho. Transfer learning for cross-language text categorization through active correspondences construction. In *AAAI*, pages 2400–2406, 2016a.

Joey Tianyi Zhou, Xinxing Xu, Sinno Jialin Pan, Ivor W. Tsang, Zheng Qin, and Rick Siow Mong Goh. Transfer hashing with privileged information. In *IJCAI*, pages 2414–2420, 2016b.

Joey Tianyi Zhou, Heng Zhao, Xi Peng, Meng Fang, Zheng Qin, and Rick Siow Mong Goh. Transfer hashing: From shallow to deep. *IEEE Trans. Neural Netw. Learning Syst.*, 29(12):6191–6201, 2018.

Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. Heterogeneous transfer learning for image classification. In *AAAI*, 2011.