Joint Feature Selection and Structure Preservation for Domain Adaptation

Jingjing Li, Jidong Zhao and Ke Lu
University of Electronic Science and Technology of China, Chengdu, China
lijin117@yeah.net, {jzhao, kel}@uestc.edu.cn

Abstract
The essence of domain adaptation is to explore common latent factors shared by the involved domains. These factors can be specific features or geometric structures. Most of previous methods exploit either the shared features or the shared geometric structures separately. However, the two strategies are complementary with each other and jointly exploring them is more optimal. This paper proposes a novel approach, named joint Feature Selection and Structure Preservation (FSSP), for unsupervised domain adaptation. FSSP smoothly integrates structure preservation and feature selection into a unified optimization problem. Intensive experiments on text categorization, image classification and video event recognition demonstrate that our method performs better, even with up to 30% improvement in average, compared with the state-of-the-art methods.

1 Introduction
As poet Sándor Petőfi once wrote, “life is dear, love is dearer;” a scientist in the filed of machine learning might say, “data is dear, labeled data is dearer.” How to acquire more labeled data from existing ones has been a crucial research topic recently. Domain adaptation [Pan and Yang, 2010] proves to be effective for leveraging labeled data in the well-labeled source domain to transfer classification discriminability to the unlabeled target domain.

Domain adaptation deals with the problem where data from two domains have common class label but divergent data distributions. Since traditional machine learning algorithms would fail to handle the situation, domain adaptation, one category of transfer learning [Pan and Yang, 2010], has been widely studied in many real world applications, e.g., image classification [Long et al., 2014b], text categorization [Ding et al., 2015] and video event recognition [Duan et al., 2012b].

The basic assumption of domain adaptation is that some common latent factors are shared by the involved domains. Therefore, the mechanism of domain adaptation is to explore these common latent factors, and utilize them to mitigate both the marginal and conditional distributions across domains, which can be done by one of the two strategies, i.e., instance re-weighting and feature extraction. Most approaches in the first group [Chu et al., 2013] try to train a sophisticated classifier on the source domain, e.g., multiple kernel SVM, which can be used in the target domain. Whereas approaches in the second group aim to preserve important data properties, e.g., statistical property and geometric structure. In most cases, feature extraction is more effective than training a complex classifier, and deep learning [Donahue et al., 2013] is a great example. In this paper, therefore, we focus on the feature extraction. However, previous methods in this group usually preserve the statistical property and geometric structure independently, e.g., [Ding et al., 2015; Pan et al., 2011] explore the statistical property by maximizing the empirical likelihood, while [Gong et al., 2012; Long et al., 2014a] optimize predefined objective function by exploring the geometric structure. In fact, these two properties are complementary with each other and jointly exploring them could benefit from both sides. The statistical properties and geometric structure are two observations of the data from different viewpoints. Each viewpoint has its theoretical base and exists unilateralism. Different viewpoints are not mutually exclusive and combining them normally could transcend the specific limitations of each perspective [Zhu and Lafferty, 2005].

In this paper, we explore the benefit of integrating the optimization of statistical property and geometric structure into a unified framework. On one hand, we seek a common subspace shared by the involved domains where common latent features can be uncovered and the data distribution gap across two domains can be mitigated. On the other hand, we deploy a graph structure to characterize the sample relationship. The motivation of this paper is illustrated in Fig. 1. Furthermore, if we simply select features from the source domain and the target domain by a general projection matrix, the selected results may be different for each dimension of the subspace learned from different domains [Gu et al., 2011]. The upper part of Fig. 1 shows an illustrative toy example. It can be seen that the selected features (colored ones) from different domains are different too. Therefore, it is hard to distinguish which features (corresponding rows) in both domains are really redundant. However, after row-sparsity regularization, the selection tends to be clear. Finally, the main contributions of this paper are summarized as follows:

1) A unified framework of feature selection and geometric
structure preservation is proposed for unsupervised domain adaptation, and it achieves state-of-the-art performance on several standard benchmarks even with up to 30% improvement in average compared with baselines.

2) In the aspect of feature selection, we deploy $\ell_{2,1}$-norm on the projection matrix, which leads to achieving row-sparsity and, as a result, selecting relevant features across the involved domains.

3) In the aspect of geometric structure preservation, not only the structure of samples is preserved by a nearest neighbor graph, but also the structure of features in the embedded space is preserved by a representation matrix.

The rest of this paper is organized as follows. Section 2 presents some brief discussion with related works. Section 3 introduces the proposed method in detail. Experiments are reported in Section 4, and Section 5 is the conclusion.

2 Related Works and Discussions

This paper focuses on domain adaptation [Pan and Yang, 2010] where the source domain and the target domain share the same task but have different data distributions.

According to the recent work [Long et al., 2014b], most of the unsupervised domain adaptation approaches work by learning a new feature representation to reduce the data distribution differences among domains. The new feature representation can be learned by: 1) exploring domain-invariant common factors [Ding and Fu, 2014; Ding et al., 2015], 2) minimizing proper distance measures [Gong et al., 2012; Long et al., 2014a], and 3) re-weighting relevant features with sparsity-promoting regularization [Gu et al., 2011; Long et al., 2014b]. Actually, these three groups can be concisely summed up in two bases: feature selection, which consists of 1) and 3), and geometric structure preservation. This paper aims to take full advantage of both feature selection and geometric structure preservation, incorporate them into a unified framework and jointly optimize them.

To our knowledge, this work is among the very leading works for domain adaptation to joint feature selection and geometric structure preservation. Notably, experiments demonstrate that our work can get better recognition accuracy than baselines with a significant advantage.

3 The Proposed Approach

3.1 Notations

In this paper, we use bold low-case symbols to represent vectors, bold upper-case symbols to represent matrices, specifically, I represents the identity matrix. A sample is denoted as a vector, e.g., $x$, and the $i$-th sample in a set is represented by the symbol $x_i$. For a matrix $M$, its $\ell_{2,1}$-norm is defined as: $\|M\|_{2,1} = \sum_j \sqrt{\sum_i (M_{ij})^2}$. We also use the Frobenius norm $\|M\|_F = \sum_i \delta_i(M)^2$, where $\delta_i(M)$ is the $i$-th singular value of the matrix $M$. The trace of matrix $M$ is represented by $\text{tr}(M)$. For clarity, we also show the frequently used notations in Table 1.

3.2 Problem Definition

Definition 1 A domain $D$ is defined by a feature space $X$ and its probability distribution $P(X)$, where $X \in \mathcal{X}$. For a specific domain, a classification task $T$ consists of class information $Y$ and a classifier $f(x)$, that is $T = \{Y, f(x)\}$.

We use subscripts $s$ and $t$ to indicate the source domain and the target domain, respectively. This paper focuses on the following problem:

**Problem 1** Given a labeled source domain $D_s$ and an unlabeled target domain $D_t$, where $D_s \neq D_t$, $Y_s = Y_t$, $P(X_s) \neq P(X_t)$ and $P(y_s|X_s) \neq P(y_t|X_t)$, find a subspace spanned by $P$ in which the common latent features shared by involved domains are uncovered, the data manifold structure is preserved, and the domain shift is minimized.

3.3 Problem Formulation

The basic assumption behind domain adaptation is that the involved domains share some common latent factors, these factors can be specific features or geometric structures, and in most cases, are both of them. If we assume that there exists a common latent subspace shared by both domains where the shared features can be uncovered, then we can find the subspace spanned by an appropriate basis $P$ where each sample from the target domain can be drawn from one subspace segmentation in the source domain. Thus, the goal of **Problem 1** can be formulated as optimizing the following objective:

$$
\min_{P,Z} \|P^T X_t - P^T X_s Z\|_F^2, \tag{1}
$$

where $P$ is the projection matrix, $Z$ is the reconstruction coefficient matrix corresponding to $X_s$, and $X_s$ serves as a dictionary [Qiu et al., 2012]. Since $X_s$ can represent $X_t$ by
appropriate $P$ and $Z$, and it is no doubt that $X$ can represent itself too. Therefore, we combine $X_s$ and $X_t$ together as complete $X$ to dig out more shared information. From [Yin et al., 2015], there are two explanations for $Z$ based on the model. Firstly, the $ij$-th element of $Z$ reflects the similarity between the sample pair $x_i$ and $x_j$. Secondly, the $ij$-th column of $Z$ serves as a better representation of $x_i$ such that the desired pattern, say subspace structure, is more prominent. From this perspective, $Z$ preserves the embedding manifold structure of samples. Furthermore, in order to learn a robust and efficient subspace, we introduce the Frobenius norm of $Z$ according to [Lu et al., 2012a].

As we have discussed in the introduction, the selected features by a general $P$ may be different for each dimension of the learned subspace, especially in the case of domain adaptation where $X_s$ and $X_t$ have divergent data distributions. This motivates us to deploy $\ell_{2,1}$-norm on $P$ [Lu et al., 2012b], which leads to selecting common features shared by the domains. As a result, we can further formulate our objective function as follows:

$$
\min_{P,Z,E} \|P\|_{2,1} + \frac{\beta}{2}\|Z\|_F^2 + \gamma \|E\|_1
$$

subject to $P^T X = P^T X_s Z + E$, where $E$ is used to detect the sample specific errors. $\beta > 0$ and $\gamma > 0$ are penalty parameters. Please note that $E$ can be very helpful when samples are corrupted, and it is also very useful to handle outliers because it is very difficult to guarantee that every sample in the target domain can be appropriately reconstructed by the source domain.

Finally, as we have discussed above, the common latent factors shared by domains are not only specific features, but also geometric structure. With the goal to jointly select features and preserve geometric structure, we introduce a graph based regularization term into our objective. Thus, the final objective function can be formulated as follows:

$$
\min_{P,Z,E} \|P\|_{2,1} + \frac{\beta}{2}\text{tr}(P^T XLX^T P) + \frac{\beta}{2}\|Z\|_F^2 + \gamma \|E\|_1
$$

subject to $P^T X = P^T X_s Z + E$, $P^T XDX^T P = I$, where $\lambda > 0$ is a penalty parameter, $L = D - W$ is the graph Laplacian [Chung, 1997] and $D = \sum_i W_{ij}$ is a diagonal matrix. $I$ is the identity matrix with proper size. The constraint $P^T XDX^T P = I$ is introduced to avoid trivial solutions. $W$ is a symmetric adjacency matrix with $W_{ij}$ characterizes the appropriate connection among the samples [Li et al., 2016], it can be computed by various criteria [Yan et al., 2007]. In this paper, we use $W$ to characterize the sample relationship and apply the heat kernel method to get $W$ as follows:

$$
W_{ij} = \begin{cases} 
\exp(-\frac{\|x_i-x_j\|^2}{2\sigma^2}), & \text{if } x_i \in kNN(x_j) \\
0, & \text{otherwise}
\end{cases}
$$

where $kNN(x_j)$ is the $k$-nearest neighbors of $x_j$.

### 3.4 Problem Optimization

Since the constraint in Eq. (3) is not convex, we convert Eq. (3) to the following equivalent equation to make it easier to optimize.

$$
\min_{P,Z,E} \|P\|_{2,1} + \frac{\beta}{2}\|P^T X - Y\|_F^2 + \frac{\beta}{2}\|Z\|_F^2 + \gamma \|E\|_1
$$

subject to $P^T X = P^T X_s Z + E$, where $Y$ is a matrix whose rows are eigenvectors of the eigen-problem $WY = \lambda Y$, and $\lambda$ is a diagonal matrix of which diagonal elements are eigenvalues. The equivalence proof of Eq. (3) and Eq. (5) can be found in [Cai et al., 2007; Gu et al., 2011].

Now, Eq. (5) can be optimized by the augmented Lagrangian multiplier (ALM) [Lin et al., 2010]. First, we transform Eq. (5) into the augmented Lagrangian function:

$$
\min_{P,Z,E} \|P\|_{2,1} + \frac{\lambda}{2}\|P^T X - Y\|_F^2 + \frac{\beta}{2}\|Z\|_F^2 + \gamma \|E\|_1 + \text{tr}(U^T (P^T X - P^T X_s Z - E)) + \frac{\mu}{2}\|P^T X - P^T X_s Z - E\|_F^2,
$$

where $\mu > 0$ is a penalty parameter and $U$ is a Lagrange multiplier. Since we cannot directly optimize all the variables in Eq. (6) at the same time, we introduce the alternating direction method of multipliers (ADMM) [Hestenes, 1969]. By deploying ADMM, we can alternately update each variable one by one in an iterative manner. Thus, Eq. (6) can be solved by the following steps:

1) To solve $Z$, by taking the derivative of Eq. (6) w.r.t $Z$, and setting the derivative to zero, we get:

$$
Z = (X^T P P^T X_s + \frac{\beta}{\mu} I)^{-1} X^T P (P^T X - E + U/\mu).
$$

2) For $E$, by ignoring the irrelevant terms w.r.t. $E$, we can optimize $E$ by:

$$
E = \arg \min_{E} \frac{\gamma}{\mu} \|E\|_1 + \|E - (P^T X - P^T X_s Z + U/\mu)\|_F^2.
$$

3) To solve $P$, by taking the derivative of Eq. (6) w.r.t $P$, and setting the derivative to zero, we get:

$$
P = \Phi^{-1}((X - X_s Z)(E^T - U^T/\mu) + \frac{\lambda}{\mu} X X^T),
$$

where $\Phi = (X - X_s Z)(X - X_s Z)^T + \frac{\beta}{\mu} XX^T + \frac{\gamma}{\mu} G$. Please note that $\|P\|_{2,1}$ is not smooth, therefore, as a surrogate, we compute its sub-gradient $G$, where $G$ is diagonal and its $i$-th diagonal element can be calculated by

$$
G_{ii} = \begin{cases} 
0, & \text{if } p_i = 0 \\
\frac{1}{2p_i^2}, & \text{otherwise}
\end{cases}
$$

where $p_i$ denotes the $i$-th row of $P$.

**Problem 1** specified that the aim of this work is to find a subspace, spanned by the appropriate basis $P$, in which the common latent features shared by the involved domains can be uncovered, the data manifold structure can be preserved, and the domain shift can be minimized. However, Eq. (9) shows that the optimization of $P$ involves some unknown variables, e.g., $Z$, $E$ and $Y$. To address this problem, we apply Principal Component Analysis (PCA) [Turk and Pentland, 1991] to the initialization of our algorithm. For clarity,
Algorithm 1. Joint Feature Selection and Structure Preservation for Unsupervised Domain Adaptation

Input: Sample sets $X_t$ and $X_s$, label information of $X_s$, balanced parameter $\lambda$, $\beta$ and $\gamma$.

Initialize: $Z = 0$, $E = 0$, $U = 0$.

Initialize $\mu = 10^{-4}$, $\mu_{opt} = 10^{6}$, $\rho = 1.3$, $\epsilon = 10^{-5}$.

Output: Label information of $X_t$.

1. Initialize $P_0$ by PCA.
2. Compute $W$, $D$ and $L$.
3. Learn $Y$ by solving the eigen-problem $WY = ADY$.

Repeat
4. Get $Z$ and $E$ by Eq. (7) and Eq. (8), respectively.
5. Optimize $P$ and $G$ by Eq. (9) and Eq. (10), respectively.
6. Update the multiplier via $U_{new} = U_{old} + \mu(P^\top X - P^\top X, Z - E)$.
7. Update $\mu$ via $\mu_{new} = \min(\mu_{old}, \mu_{max})$.
8. Check the convergence condition $\|P^\top X - P^\top X, Z - E\|_\infty < \epsilon$.

until Convergence
9. Project both $X_t$ and $X_s$ to the learned subspace by $P$, that is $P^\top X_t$ and $P^\top X_s$.
10. Classify $X_t$ in the subspace by Nearest Neighbor classifier, and $X_s$ is used as reference.

Algorithm 1 shows the details of our method. Limited by space, please refer to [Nie et al., 2010] for a similar convergence analysis of this algorithm.

3.5 Computational Complexity

The computational cost of Algorithm 1 is composed of several major parts listed as follows:
- The eigen-problem solved in step 3.
- Matrix inversion and multiplication in step 4 and 5.

Here we analyze the computational complexity by the big $O$ notation. For simplicity and without loss of generality, we assume the matrix which we handled are with the size of $n \times m$, and $d$ is the dimensionality of the learned subspace where $d \ll \min(m, n)$. The eigen-decomposition costs $O(dn^2)$, matrix inversion and multiplication cost a maximum of $O(m^3)$. Thus, the total cost of Algorithm 1 is much less than $O(kn^2)$ because lots of matrix operations are performed in the embedded low-dimensional space, where $k$ indicates the number of matrix operations. When $m$ is very large, we could adopt divide-and-conquer to address the large-scale data problem.

4 Experiments

In this section, we evaluate our algorithm on several standard benchmarks which consist of text dataset, image dataset and video dataset. We compare our algorithm with several state-of-the-art domain adaptation approaches, e.g., GFK [Gong et al., 2012], TJM [Long et al., 2014b], TCA [Pan et al., 2011], TSL [Si et al., 2010], and DLRC [Ding et al., 2015]. Since we apply PCA [Turk and Pentland, 1991] to the initialization of our algorithm and 1-Nearest Neighbor as the classifier (NCC), we also compare our method with both of them. Specifically, for PCA, we use the model trained on $X_s$ to recognize $X_t$, and for NCC, we use $X_s$ as reference to classify $X_t$ in the original data space. To fully demonstrate the superiority of our method, we also compare our method with several approaches in the group of instance re-weighting strategy on the evaluations of video event recognition. All of the reported results are the classification accuracy on the target domain, which is also widely used in literature [Gong et al., 2012; Long et al., 2014b]:

$$\text{accuracy} = \frac{|x : x \in X_t \land y_k = y_t|}{|x : x \in X_t|}$$

where $y_k$ is the predicted label of the target domain by each approach, and $y_t$ is the real label vector.

Each of the hyper-parameters used in our experiments is the optimal one chosen from a large range. We chose an acceptable common set of them for consistency. For the sake of fairness, all of the datasets used in our experiments were downloaded from the webpages of the related works, and we strictly followed the same experimental settings with them.

4.1 Data Description

Amazon, Caltech-256, DSLR, and Webcam (4DA) is the most popular benchmark in the field of domain adaptation. 4DA experimental setting was firstly introduced in [Gong et al., 2012], which is an extension of 3DA benchmark introduced in [Saenko et al., 2010]. 3DA includes object categories from Amazon (A, images downloaded from amazon.com), DSLR (D, high-resolution images by a digital SLR camera) and Webcam (W, low-resolution images by a web camera). A, D and W are three different domains, and each domain consists of 31 categories, e.g., monitor, keyboard and laptop. 4,652 images are the total number of 3DA. 4DA contains an additional domain, Caltech-256 (C) [Griffin et al., 2007], which has 30,607 images and 256 categories. Some of the selected samples from 4DA are shown in Fig. 2. Our experimental configuration on 4DA is identical with [Gong et al., 2012]. Specifically, 10 common classes shared by four datasets are selected. There are 8 to 151 samples per category per domain, and 2,533 images in total. Furthermore, 800 dimensional SURF features are extracted as our low-level input. Then the low-level input is normalized to unit.

Reuters-215782 is a challenging text dataset with several different categories. The widely used 3 largest top categories of Reuters-215782 are orgs, people, and place, each of the top categories consists of many subcategories. As suggested in [Ding et al., 2015], we evaluate our approach on the pre-processed version of this dataset with the same settings of [Gao et al., 2008].
We empirically set \( \lambda = 0.1, \beta = 0.1 \) and \( \gamma = 1 \). The dimensionality of subspace is set to 30, and the number of neighbors is set to 5. The nearest neighbor graph is learned in an unsupervised manner.

For MRSC+VOC, we perform two evaluations: 1) MRSC → VOC2007 and 2) VOC2007 → MRSC. For each evaluation, the first dataset serves as the source domain and is used for training, the second dataset serves as the target domain and is used for testing. The experimental results on this dataset are shown in Fig. 3.

For Reuters-215782, we perform six evaluations, i.e., people → orgs, orgs → people, place → orgs, orgs → place, place → people, and people → place. In each evaluation, the first dataset serves as the source domain and is used for training, the second dataset serves as the target domain and is used for training.

### Table 2: Recognition results (%) of domain adaptation on 4DA dataset. Since DLRC did not use DSLR as source domain for the reason of small sample number, we only report 9 results for DLRC.

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
<th>PCA</th>
<th>NNC</th>
<th>TCA</th>
<th>GFK</th>
<th>TSL</th>
<th>TJM</th>
<th>DLRC</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caltech-256</td>
<td>Amazon</td>
<td>37.58</td>
<td>23.70</td>
<td>35.70</td>
<td>41.05</td>
<td>45.25</td>
<td>46.76</td>
<td>49.75</td>
<td>75.78</td>
</tr>
<tr>
<td></td>
<td>Webcam</td>
<td>38.98</td>
<td>25.76</td>
<td>39.06</td>
<td>40.68</td>
<td>33.37</td>
<td>39.98</td>
<td>41.76</td>
<td>75.25</td>
</tr>
<tr>
<td></td>
<td>DSLR</td>
<td>42.04</td>
<td>25.48</td>
<td>41.44</td>
<td>38.81</td>
<td>44.15</td>
<td>44.59</td>
<td>47.85</td>
<td>76.43</td>
</tr>
<tr>
<td>Amazon</td>
<td>Caltech-256</td>
<td>38.22</td>
<td>26.00</td>
<td>37.36</td>
<td>40.28</td>
<td>37.51</td>
<td>39.45</td>
<td>42.75</td>
<td>79.16</td>
</tr>
<tr>
<td></td>
<td>Webcam</td>
<td>35.93</td>
<td>29.83</td>
<td>37.67</td>
<td>39.00</td>
<td>34.49</td>
<td>42.03</td>
<td>42.93</td>
<td>75.93</td>
</tr>
<tr>
<td></td>
<td>DSLR</td>
<td>29.94</td>
<td>25.48</td>
<td>33.32</td>
<td>36.35</td>
<td>27.81</td>
<td>45.22</td>
<td>41.86</td>
<td>74.52</td>
</tr>
<tr>
<td>Webcam</td>
<td>Caltech-256</td>
<td>26.11</td>
<td>10.95</td>
<td>29.30</td>
<td>30.73</td>
<td>28.97</td>
<td>30.19</td>
<td>33.85</td>
<td>81.85</td>
</tr>
<tr>
<td></td>
<td>Amazon</td>
<td>27.77</td>
<td>14.82</td>
<td>30.05</td>
<td>29.76</td>
<td>30.15</td>
<td>29.96</td>
<td>38.57</td>
<td>82.98</td>
</tr>
<tr>
<td></td>
<td>DSLR</td>
<td>73.25</td>
<td>24.20</td>
<td>87.29</td>
<td>80.83</td>
<td>86.57</td>
<td>89.17</td>
<td>94.31</td>
<td>93.63</td>
</tr>
<tr>
<td>DSLR</td>
<td>Caltech-256</td>
<td>26.18</td>
<td>10.60</td>
<td>31.81</td>
<td>30.28</td>
<td>28.49</td>
<td>31.43</td>
<td>38.81</td>
<td>93.22</td>
</tr>
<tr>
<td></td>
<td>Webcam</td>
<td>66.78</td>
<td>31.53</td>
<td>86.13</td>
<td>75.59</td>
<td>83.75</td>
<td>85.42</td>
<td>32.78</td>
<td>81.21</td>
</tr>
<tr>
<td></td>
<td>Amazon</td>
<td>39.12</td>
<td>11.69</td>
<td>32.29</td>
<td>32.06</td>
<td>29.06</td>
<td>32.78</td>
<td>48.18</td>
<td>81.06</td>
</tr>
</tbody>
</table>

**Figure 3:** Recognition results on MRSC+VOC2007. PCA and NNC are traditional learning methods, while others are transfer learning approaches.

**Figure 4:** Recognition results on Reuters-215782. For better visual effect, we selected the results of 5 methods to show. DLRC represents state-of-the-art performance of baselines.
and the results are promising. Better than NNC, which means perform domain adaptation or domain adaptation, is valuable and practical for real world domain adaptation approaches in the field: DASVM [Bruzzone and Marconcini, 2010], MKMM [Schweikert et al., 2009], DAM [Duan et al., 2009], and DSM [Duan et al., 2012a]. These methods include algorithms in the group of instance re-weighting strategy. Thus, the experiments on this dataset can also demonstrate the superiority of our method compared with the instance re-weighting ones. The experimental results are shown in Table 3.

For 4DA, two different datasets are randomly selected as the source domain and the target domain, respectively, thus leading to \(4 \times 3 = 12\) evaluations. The recognition results are reported in Table 2.

For CCV, we compare our method with four widely cited domain adaptation approaches in the field: DASVM [Bruzzone and Marconcini, 2010], MKMM [Schweikert et al., 2009], DAM [Duan et al., 2009], and DSM [Duan et al., 2012a]. These methods include algorithms in the group of instance re-weighting strategy. Thus, the experiments on this dataset can also demonstrate the superiority of our method compared with the instance re-weighting ones. The experimental results are shown in Table 3.

### 4.3 Discussions

From the experimental results, several observations can be drawn as follows:

1) Transfer learning methods perform much better than traditional (non-transfer) ones, which means transfer learning, or domain adaptation, is valuable and practical for real world applications.

2) All of the subspace learning approaches work much better than NNC, which means perform domain adaptation through a dimensionality reduction procedure is not trivial and the results are promising.

3) The baselines either try to maximize the empirical likelihood to explore specific features, e.g., TCA and DLRC, or aim to preserve geometric structure by minimizing proper distance measures, e.g., GFK, and each of them is the representative method in their own field, but none of them performs better than our FSSP in average. It exactly demonstrates the motivation of our work, that is jointly optimizing feature selection and geometric structure preservation is more optimal than optimizing them separately.

4) TIM performs joint feature matching and instance re-weighting, but it does not consider the data structure of samples. As a result, it performs worse than our FSSP.

5) From the results reported in Table 2, it can be seen that our FSSP significantly advances state-of-the-art baselines with 30% accuracy rates in average. It is quite impressive since 4DA is one of the most popular and challenging benchmarks in the literature.

6) It can be seen from Table 3 that our method outperforms the baselines notably. As [Duan et al., 2012a] pointed out, video event recognition is challenging because irrelevant source domains may be harmful for the classification performances in the target domain. Most baselines perform bad because the so-called negative transfer [Pan and Yang, 2010]. Our method performs well because we only select the relevant features, and use relevant neighbors for reconstruction. The dimensionality reduction which we performed can further filter negative information. Finally, the results can also demonstrate the effectiveness of feature extraction compared with instance re-weighting.

7) Fig. 5(a) shows the parameters sensitivity of our method. It can be seen that our algorithm is robust with different values of \(\beta\) and \(\gamma\) when \(\lambda\) is fixed, but \(\lambda\) needs to be carefully chosen from \([0.01, 1]\). Fig. 5(b) shows that our method performs smoothly with varying dimensionality of subspace. However, computational costs will grow with the dimensionality increasing. Fig. 5(c) shows that our algorithm converges very fast, usually within about 5-round iterations.

### 5 Conclusion

This paper proposes a unified framework of joint feature selection and geometric structure preservation for unsupervised domain adaptation. Experiments on both visual dataset and text dataset demonstrate that the joint optimization is much better than separate ones.
Acknowledgments
This work is supported in part by the National Science Foundation of China under Grants 61273254 and 61371183.

References
[Gao et al., 2008] Jing Gao, Wei Fan, Jing Jiang, and Jiawei Han. Knowledge transfer via multiple model local structure mapping. In ACM SIGKDD, pages 283–291. ACM, 2008.