Sparse Manifold Alignment

Chang Wang
IBM Watson Research Center
Yorktown Heights, NY
chwang@cs.umass.edu

Bo Liu
Department of Computer Science
University of Massachusetts, Amherst
boliu@cs.umass.edu

Hoa Vu
Department of Computer Science
University of Massachusetts, Amherst
hvu@cs.umass.edu

Sridhar Mahadevan
Department of Computer Science
University of Massachusetts, Amherst
mahadeva@cs.umass.edu

September 5, 2012

Abstract

Previous approaches to manifold alignment are based on solving a (generalized) eigenvector problem. We propose a least squares formulation of a class of manifold alignment approaches, which has the potential of scaling better to real-world data sets. Furthermore, the least-squares formulation enables various regularization techniques to be readily incorporated to improve model sparsity and generalization ability. In particular, it enables using the l_1 norm regularization framework to make previous manifold alignment algorithms more robust. The new approach can prune domain-dependent features automatically helping to improve transfer learning. This extension significantly broadens the scope of manifold alignment techniques and leads to faster algorithms. We present detailed experiments to illustrate the approach using the domains of cross-lingual information retrieval and social network analysis.

1 Motivation

Manifold alignment is a new approach to domain adaptation and transfer learning [1, 3, 13]. The key idea underlying this approach is to map different domains to a new latent space, simultaneously matching the corresponding instances and preserving the local (or global) geometry of each input domain. Manifold alignment makes use of both unlabeled and labeled data. The ability to exploit unlabeled data is particularly useful for domain adaptation, where the number of labeled instances in the target domain is usually limited. Many previous approaches to manifold alignment involve solving a (generalized) eigenvector problem, which can be prohibitively expensive in large real-world domains.

A key difficulty in applying manifold alignment to domain adaptation is that each input domain may have many input features, and some of them do not provide knowledge shared across all input datasets. Such features could lead to overfitting, especially when the given correspondence information is insufficient. In this paper, we show that a class of manifold alignment problems can be formulated as a least squares problem. As a result, various regularization techniques can be readily incorporated into the formulation to improve model sparsity (feature selection) and generalization ability. This idea is based on the observation that once the manifold alignment problem is formalized as a least squares problem, the general framework of l_1 regularization can be incorporated into the loss function for feature selection. This enhancement provides manifold alignment with the extra power to remove useless features, and can speed up the computation since algorithms for solving least squares regularization are often significantly more efficient compared to methods for solving generalized eigenvalue decomposition used in previous manifold alignment methods.

Our approach is designed to learn *sparse* mapping functions to project the source and target domains to a new latent space, simultaneously matching the instances in correspondence and preserving the local geometry of each input domain. The contributions of this paper are twofold. From the perspective of transfer learning and domain adaptation, our contribution is a new approach to address the problem of transfer, removing the domain-dependent features automatically. From the perspective of manifold alignment, our contribution is to incorporate the l_1 norm optimization framework to improve previous approaches. This enhancement significantly broadens the scope of manifold alignment techniques and helps speed up the computation. We present detailed experiments applying the new approach to cross-lingual information retrieval and social network analysis.

2 Background

2.1 Manifold Alignment

We now review some background material necessary for the analysis of our alignment framework.

2.1.1 Instance-level alignment

Semi-supervised alignment [3] finds the best alignment mapping for instances x_i and y_i by minimizing the following cost function:

$$C(f,g) = \mu\sum_{i=1}^{l}\|f_i - g_i\|^2 + 0.5\sum_{i,j}\|f_i - f_j\|^2 W_x^{i,j} + 0.5\sum_{i,j}\|g_i - g_j\|^2 W_y^{i,j} = \left[f^T\ g^T\right] L \left[\begin{array}{c} f \\ g \end{array}\right],$$

where f_i is the embedding of x_i, g_i is the embedding of y_i, and μ is the weight of the first term. The first term penalizes the differences between X and Y on the embeddings of the corresponding instances. The second and third terms ensure that the local geometries within X and Y will be preserved. L is the combinatorial graph Laplacian matrix. To remove an arbitrary scaling factor in the embedding, an extra constraint is imposed: $f^T f + g^T g = \gamma^T \gamma = I$. Then, the d-dimensional alignment result is given by

$$\left[\begin{array}{c} f \\ g \end{array}\right] = [\gamma_1 \cdots \gamma_d] = \gamma,$$

where f, g are of size $n \times d$, and $\gamma_1 \cdots \gamma_d$ are eigenvectors of $L\xi = \lambda\xi$ corresponding to the d smallest non-zero eigenvalues.

2.1.2 Feature-level Alignment

Manifold projections [13] learns mapping functions α and β for alignment. When the correspondence is given, its cost function is given as:[1]

$$\begin{aligned} C(F,G) &= \mu\sum_{i}^{l}\left\|F^T x_i - G^T y_i\right\|^2 + 0.5\sum_{i,j}\left\|F^T x_i - F^T x_j\right\|^2 W_x^{i,j} + 0.5\sum_{i,j}\left\|G^T y_i - G^T y_j\right\|^2 W_y^{i,j} \\ &= \left[F^T X\ G^T Y\right] L \left[\begin{array}{c} X^T F \\ Y^T G \end{array}\right]. \end{aligned}$$

To remove an arbitrary scaling factor in the embedding, an extra constraint is needed: $F^T XX^T F + G^T YY^T G = \gamma^T ZZ^T \gamma = I$. Then, the d-dimensional mapping function is given by

$$\left[\begin{array}{c} F \\ G \end{array}\right] = [\gamma_1 \cdots \gamma_d] = \gamma,$$

where $\gamma_1 \cdots \gamma_d$ are the eigenvectors of $ZLZ^T\xi = \lambda ZZ^T\xi$ corresponding to the d smallest non-zero eigenvalues. Manifold projections builds mappings between features (rather than instances) across manifolds, so it can handle new test instances without requiring out of sample extension and makes direct knowledge transfer possible.

[1] When no correspondence is given or when one instance can match multiple instances in another dataset, the loss function can be specified in a more general manner.

2.2 Lasso and Its Variants

In machine learning, regularization is often used to avoid overfitting. The most common approach of regularization is to impose some penalty on the norm of the coefficients in order to achieve a sparse and simple model. Recently, l_1 regularization has attracted a lot of attention because of its ability to obtain sparsity. A well-known example of l_1 regularization is regression shrinkage and selection via the Lasso [10]. Lasso performs l_1 penalized regression by solving the optimization problem given by:

$$\underset{w}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i - w^T x_i\right)^2 + \alpha ||w||_1.$$

l_1 penalized regression has the following limitations: 1) if $p \gg n$, methods such as Lasso can select at most n features. 2) They tends to select only one feature from a group of correlated features and do not distinguish between which one is selected. 3) when $n > p$ and the correlation between the features is high, ridge regression empirically performs better [16]. Elastic net [16] aims to have a better grouping effect, and performs better when $p \gg n$. The elastic net is a linear interpolation of the ridge and the Lasso objective functions, and its loss function is as follows:

$$\arg\min_{w} \sum_{i=1}^{n} \left(y_i - w^T x_i\right)^2 + \alpha ||w||_1 + \gamma ||w||_2.$$

Elastic net has been shown to successfully produce a sparse model with good accuracy and have a better grouping effect over the Lasso. In addition, if we know the feature group in advance, group Lasso [15] tends to outperform other approaches, achieving sparsity at the group level. When the features can be ordered in a meaningful way, a method called Fused Lasso [11] can yield a sparse solution with small successive difference. The loss function of the Fused Lasso is in the form of

$$\arg\min_{w} \sum_{i=1}^{n} \left(y_i - w^T x_i\right)^2 + \alpha ||w||_1 + \beta \sum_{j=2}^{p} |w_j - w_{j-1}|.$$

3 Sparse Manifold Alignment

3.1 High Level Explanation

Given k input manifolds, our goal is to construct k *sparse* mapping functions to project the input manifolds to a common latent space for alignment. To achieve this goal, we first formulate the manifold alignment problem as a least squares problem. Then we incorporate the regularization terms (like l_1 term) in the least-squares formulation for model sparsity and generalization ability. The resulting alignment result produced by the least squares solver is optimal regarding the new loss function, but not for the previous loss function used in

$x_i \in R^p$; $X = \{x_1, \cdots, x_m\}$ is a $p \times m$ matrix; $X_l = \{x_1, \cdots, x_l\}$ is a $p \times l$ matrix.
$y_i \in R^q$; $Y = \{y_1, \cdots, y_n\}$ is a $q \times n$ matrix; $Y_l = \{y_1, \cdots, y_l\}$ is a $q \times l$ matrix .
X_l and Y_l are in correspondence: $x_i \in X_l \longleftrightarrow y_i \in Y_l$.

W_x is a similarity matrix, e.g. $W_x^{i,j} = e^{-\frac{||x_i - x_j||^2}{2\sigma^2}}$. D_x is a full rank diagonal matrix: $D_x^{i,i} = \sum_j W_x^{i,j}$;
$L_x = D_x - W_x$ is the combinatorial Laplacian matrix.
W_y, D_y and L_y are defined similarly.

$\Omega_1 - \Omega_4$ are all diagonal matrices having μ on the top l elements of the diagonal (the other elements are 0s); Ω_1 is an $m \times m$ matrix; Ω_2 and Ω_3^T are $m \times n$ matrices; Ω_4 is an $n \times n$ matrix.
$Z = \begin{pmatrix} X & 0 \\ 0 & Y \end{pmatrix}$ is a $(p+q) \times (m+n)$ matrix.
$D = \begin{pmatrix} D_x & 0 \\ 0 & D_y \end{pmatrix}$ and $L = \begin{pmatrix} L_x + \Omega_1 & -\Omega_2 \\ -\Omega_3 & L_y + \Omega_4 \end{pmatrix}$ are both $(m+n) \times (m+n)$ matrices.
F is a $(p+q) \times r$ matrix, where r is the rank of ZZ^T and $FF^T = ZZ^T$. F can be constructed by SVD.
$(\cdot)^+$ represents the Moore-Penrose pseudoinverse.

H is an $(m+n) \times h$ matrix, where $L = HH^T$, and h is the rank of H.
Q and R are the thin QR decomposition results of H, where $H = QR$, $Q \in \mathcal{R}^{(m+n)\times h}$ and $R \in \mathcal{R}^{h \times h}$.
U_h, Σ_h and V_h are all $h \times h$ matrices, and $R = U_h \Sigma_h V_h^T$.

Figure 1: Some notation used in this paper.

the original eigenvalue decomposition-based approach. Compared against the previous approaches, the added sparsity constraints prune domain-dependent features automatically, further lower the chances of overfitting, and help improve transfer learning.

3.2 Justification

Theorem 1 proves that the manifold alignment problem previously formulated as generalized eigenvalue decomposition problems can be reformulated as regular eigenvalue decomposition problems. Theorem 2 proves the equivalence of the result of Theorem 1 and a least squares regularization problem under the condition: $rank(Z) = m+n-1$, where Z is the joint instance matrix. This condition is likely to hold when the data dimensionality is larger than the sample size, and it has also been used in the other approaches in the literature [9]. It is worth noting that the solution to a similar problem is given in [9], assuming $XSX^Tw = \lambda XX^Tw$ can be reformulated as $(XX^T)^+XSX^Tw = \lambda w$, where $()^+$ represents the pseudoinverse. This assumption does not hold for our problem, since $(XX^T)^+XX^T$ is not an identity matrix when the data dimensionality is larger than the sample size.

Theorem 1: The solution to the generalized eigenvalue decomposition $ZLZ^T\gamma = \lambda ZZ^T\gamma$ is given by $((F^T)^+x, \lambda)$, where x and λ are eigenvector and eigenvalue of $F^+ZLZ^T(F^T)^+x = \lambda x$. (See supplemental material for the proof).

Theorem 2: The solution to the generalized eigenvalue decomposition $ZLZ^T\gamma = \lambda ZZ^T\gamma$ is given by minimization of $\|W^TZ - U_h^TQ^T\|_F^2$ under the condition of $rank(Z) = m+n-1$.
Proof:

L is a Laplacian matrix, so

$$e^TLe = 0, \quad \text{where } e \text{ is a vector of all ones.} \tag{1}$$

Since L is a positive semi-definite matrix, we can decompose L as follows:

$$L \longrightarrow HH^T, \tag{2}$$

where $H \in \mathcal{R}^{(m+n)\times h}$ and h is the rank of H.

$$\text{Let} \qquad H = QR \tag{3}$$

be the thin QR decomposition of H, where $Q \in \mathcal{R}^{(m+n)\times h}, R \in \mathcal{R}^{h\times h}$.

$$\text{Let} \qquad R = U_h\Sigma_hV_h^T \tag{4}$$

be the Singular Value Decomposition of R, where $U_h, \Sigma_h, V_h \in \mathcal{R}^{h\times h}$.

From Equation (2), (3) and (4), we have $L = HH^T = QRR^TQ^T = QU_h\Sigma_h^2U_h^TQ^T$. (5)

$$\text{Let} \qquad Z = U_Z \Sigma_Z V_Z^T \tag{6}$$

be the compact Singular Value Decomposition of Z, where Σ_Z is a full rank square matrix. This implies that $ZZ^T = U_Z \Sigma_Z \Sigma_Z U_Z^T$. From the definition of F in Figure 1, we know

$$F = U_Z \Sigma_Z. \tag{7}$$

From Equation (7), we have the following results:

$$F^T = \Sigma_Z U_Z^T,\ F^+ = \Sigma_Z^+ U_Z^+, (F^T)^+ = (U_Z^T)^+ \Sigma_Z^+ = U_Z \Sigma_Z^+. \tag{8}$$

Combining Equation (5), (6) and (8), we have

$$\begin{aligned} F^+ Z L Z^T (F^T)^+ &= \Sigma_Z^+ U_Z^+ U_Z \Sigma_Z V_Z^T L V_Z \Sigma_Z U_Z^T (U_Z^T)^+ \Sigma_Z^+ && (9) \\ &= \Sigma_Z^+ \Sigma_Z V_Z^T L V_Z \Sigma_Z (\Sigma_Z)^+ = V_Z^T L V_Z = (V_Z^T Q U_h) \Sigma_h^2 (U_h^T Q^T V_Z) && (10) \end{aligned}$$

To show the columns of matrix $V_Z^T Q U_h$ are the eigenvectors of matrix $F^+ Z L Z^T (F^T)^+$, we need to prove

$$(U_h^T Q^T V_Z)(V_Z^T Q U_h) = I. \tag{11}$$

When Z is centered, we have $Ze = 0$. From Equation (6), we have

$$\begin{aligned} Ze = 0 \quad &\Rightarrow \quad e^T Z^T Z e = 0 \Rightarrow e^T V_Z \Sigma_Z U_Z^T U_Z \Sigma_Z V_Z^T e = 0 && (12) \\ &\Rightarrow \quad e^T V_Z \Sigma_Z^2 V_Z^T e = 0 \Rightarrow V_Z^T e = 0. && (13) \end{aligned}$$

From Equation (5), we have

$$e^T L e = 0 \Rightarrow e^T Q U_h \Sigma_h^2 U_h^T Q^T e = 0 \Rightarrow U_h^T Q^T e = 0. \tag{14}$$

It can be verified that

$$V_Z^T V_Z = I \ \text{ and } \ (Q U_h)^T Q U_h = U_h^T Q^T Q U_h = I. \tag{15}$$

Using Equation (12)-(15), the given condition and Lemma 2 in [9], we know

$$(U_h^T Q^T V_Z)(V_Z^T Q U_h) = I. \tag{16}$$

Combining Theorem 1, Equation (8) and (16), we know the solution to the generalized eigenvalue decomposition $Z L Z^T \gamma = \lambda Z Z^T \gamma$ is given by

$$(F^T)^+ V_Z^T Q U_h = U_Z \Sigma_Z^+ V_Z^T Q U_h. \tag{17}$$

From Theorem 2 in [9] and Equation (17), we know the eigenvector solution to our generalized eigenvalue decomposition is also given by

$$\|W^T Z - U_h^T Q^T\|_F^2. \tag{18}$$

□

4 The Algorithmic Framework

Given X, X_l, Y, Y_l, the notation defined in Figure 1, the algorithm to compute the least squares formulation of manifold alignment is as follows:

Algorithm : Sparse Manifold Alignment

1. **Center X and Y.**

2. **Use Lasso solver [8] to find W that minimizes $\|W^T Z - U_h^T Q^T\|_F^2$ or its variants (Section 4.1).**

3. **Compute mapping functions for manifold alignment:**
   $\begin{bmatrix} F \\ G \end{bmatrix} = W$ is a $(p+q) \times h$ matrix.

   For any i and j, $F^T x_i$ and $G^T y_j$ are in the same h dimensional space and can be directly compared.

4.1 Extensions

Instance-level Manifold Alignment: To use our framework to perform nonlinear instance-level alignment (described in Section 2.1.1), we need to minimize the cost function $C(f, g)$ instead of $C(F, G)$. This problem is less challenging and can be solved in a similar manner.

Manifold Alignment with Lasso: The Lasso achieves both feature selection and weight shrinkage. Integrating Lasso in the manifold alignment loss function can help us achieve a sparsified projection to align manifolds. To use Lasso, we can simply replace the loss function used in Step 2 with the one given below (h dimensional embedding):

$$\|W^T Z - U_h^T Q^T\|_F^2 + \alpha \|W\|_{1,1}.$$

Manifold Alignment with Fused Lasso: When the input features can be ordered in a meaningful way, Fused Lasso [11] can be integrated into manifold alignment, yielding a sparse projection function with small successive differences in the weights. To use the Fused Lasso, we need to replace the loss function used in Step 2 with the following one (involving an h-dimensional embedding):

$$\|W^T Z - U_h^T Q^T\|_F^2 \quad + \quad \alpha \|W\|_{1,1} + \beta \sum_{j=1}^{h} \sum_{k=2}^{p+q} |w_{j,k} - w_{j,k-1}|.$$

5 Applications and Results

We compare our Lasso/Fused Lasso-based alignment methods against instance-level [3] and feature-level [13] manifold alignment techniques and other state of the art approaches, including Canonical Correlation Analysis (CCA) [4], Affine matching based alignment [6] and Procrustes alignment [12]. We use the SLEP package [7] [8] to solve the Lasso and Fused Lasso.

5.1 Cross-Lingual Information Retrieval

Nine approaches are tested in an experiment involving cross-lingual information retrieval. Three of them are instance-level approaches: Procrustes alignment with Laplacian eigenmaps, Affine matching with Laplacian eigenmaps, and instance-level manifold alignment. The other six are feature-level approaches: Procrustes alignment with LPP, Affine matching with LPP, CCA, feature-level manifold alignment and our Lasso and Fused Lasso-based manifold alignment approaches. The order used in the Fused Lasso is based on the word frequency in the corpus. Procrustes alignment and Affine matching can only handle pairwise alignment, so when we align two collections the third collection is not taken into consideration. The other manifold alignment approaches and CCA align all input data simultaneously. In all methods, we use k-nearest neighbor matrix ($k = 10$) as adjacency graphs. The parameters $\alpha = 0.1$ and $\beta = 0.01$ in this experiment.

In this experiment, we make use of the proceedings of European Parliament [5], dating from 04/1996 to 10/2009. The corpus includes versions in 11 European languages. Altogether, the corpus comprises of about 55 million words for each language. The data for our experiment comes from the English, Italian and German collections. The dataset has many files. Each file contains the utterances of one speaker in turn. We treat an utterance as a document. We filtered out stop words, and extracted English-Italian-German document triples where all three documents have at least 75 words. This resulted in 70,458 document triples. We then represented each English document with the most commonly used 2,500 English words, each Italian document with the most commonly used 2,500 Italian words, and each German document with the most commonly used 2,500 German words. The documents were represented as bags of words, and no tag information was included. The topical structure of each collection can be thought as a manifold over documents. Each document is a sample from the manifold.

5.1.1 Experiment 1 (1,500 Test Documents)

Instance-level manifold alignment cannot process a very large collection since it needs to do an eigenvalue decomposition of an $(m_1+m_2+m_3)\times(m_1+m_2+m_3)$ matrix, where m_i represents the number of examples in the i^{th} input dataset. Approaches based on Laplacian eigenmaps suffer from a similar problem. In this experiment, we use a small subset of the whole dataset to test all nine approaches. $1,000$ document triples were used as corresponding triples in training and $1,500$ other document triples were used as unlabeled documents for both training and testing, i.e. $p_1 = p_2 = p_3 = 2,500$, $m_1 = m_2 = m_3 = 2,500$. $x_1^i \longleftrightarrow x_2^i \longleftrightarrow x_3^i$ for $i \in [1, 1000]$. Similarity matrices W_1, W_2 and W_3 were all $2,500 \times 2,500$ adjacency matrices constructed by nearest neighbor approach, where $k = 10$. To use Procrustes alignment and Affine matching, we ran a pre-processing step with Laplacian eigenmaps and LPP to project the data to a $d = 200$ dimensional space. In CCA and feature-level manifold alignment, d is also 200. Our testing scheme is as follows: for each given English document, we retrieve its top k most similar Italian documents. The probability that the true match is among the top k documents is used to measure the performance of the method. We also consider two other scenarios in the same setting: English → German and Italian → German. Figure 2 summarizes the average performance of these three scenarios.

Our first finding is that the regular Lasso-based approach performs much better than previous approaches. Given a document in one language, it has a 25% probability of finding the true match if we retrieve the most similar document in another language. If we retrieve 10 most similar documents, the probability of finding the true match increases to 44%. By integrating the l_1 term in the manifold alignment loss function, the resulting projection functions are sparse. Sparsity often leads to a good generalization ability and our result supports this.

The second result shown in Figure 2 is that the Fused Lasso-based manifold alignment outperforms all the other approaches by a large margin. It has a 73% probability of finding the true match if we retrieve the most similar document in another language. The order used in the Fused Lasso is based on the word frequency in the corpus, so our Fused Lasso-based method will encourage the words with similar background frequencies to be processed in a similar manner. This is similar to taking the inverse document frequency (idf) into consideration in the alignment process. It is well-known that integrating the existing knowledge (like term background frequency) can help solve overfitting problems when the training data is insufficient. This is justified by our result.

The third result is that CCA does a very poor job in aligning the test documents. CCA can be shown as a special case of feature-level manifold alignment preserving local geometry when manifold topology is not respected. When the training data is limited, CCA has a large chance of overfitting the given correspondences. Manifold alignment does not suffer from this problem, since the manifold topology also needs to be respected in the alignment.

5.1.2 Experiment 2 (10,000 Test Documents)

Feature-level approaches have two advantages over instance-level approaches. Firstly, feature-level approaches learn feature correlations, so they can be applied to a large dataset and directly generalize to new test data. Secondly, they are less sensitive to overfitting compared to instance-level approaches due to the "linear" constraint on mapping functions. In our second setting, we apply the resulting mapping functions from the feature-level approaches in the previous test to a larger testset with 10,000 test English-Italian-German document triples. This test is used to measure the generalization ability of each approach. This test under the second setting is in fact very hard, since we have thousands of features in each input dataset but only 1,000 given corresponding triples to learn the projection functions.

The results are summarized in Figure 3. The Lasso-based approaches again perform much better than the existing feature-level manifold alignment ap-

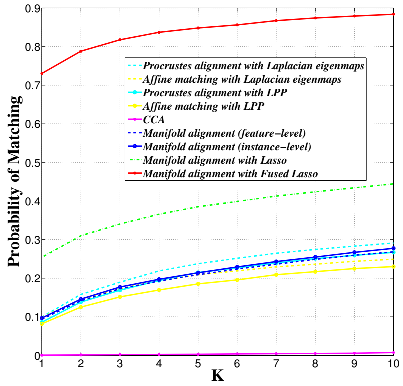

Figure 2: Cross-Lingual Retrieval Test 1

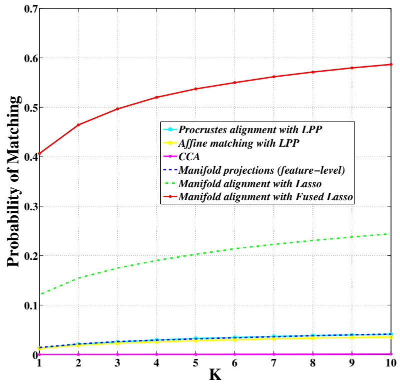

Figure 3: Cross-Lingual Retrieval Test 2

proaches. For any given document, if we retrieve the most similar document in another language, the Fused Lasso-based approach has a 40% chance of getting the true match. If we retrieve 10 most similar Italian documents, the new approach has a 60% probability of getting the true match. For regular Lasso-based approach, the two numbers are 12% and 24%, which are also 10% better than the previous approaches. This result shows that applying a regularization to the projection function is quite important. It significantly improves the generalization ability. Further, the Fused Lasso can take the prior information into consideration (background term frequency in the corpus for our test), and fit particularly for applications like text mining. Such information is not used in the previous manifold alignment techniques. In contrast to most approaches in cross-lingual knowledge transfer, we are not using any specialized pre-processing technique from information retrieval to tune our framework to this task.

5.2 Alignment of Social Networks

In this experiment, we align multiple social networks. The networks were constructed from the snapshots of DBLP authorship networks evolving over time. Following the approach presented in [2], we selected a set of authors, who contributed in at least eight of the ten years dating from 1995 to 2002. From this set of authors, we chose the largest connected component of the first snapshot (1995). This resulted in a set of 2,538 authors at 10 different time points. We built a data set for each year, using the authors as instances and diffusion topics [14] as features. To build diffusion topics for each year, we first represented each author using the keywords from the titles of all his papers published in that year. After removing the stop words, we created a word-word matrix $T = A^T A$ from the author-word matrix A with the 5,000 most popular words. Since the rank of T is 2,538 for all 10 years, the finest levels of all diffusion topic models

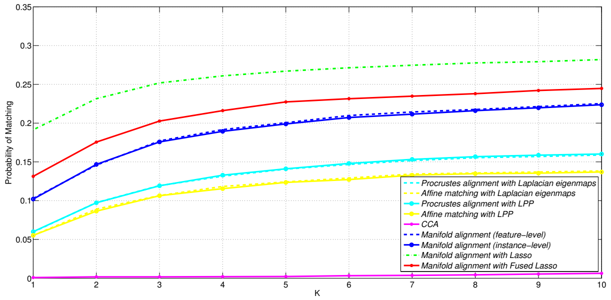

Figure 4: Alignment of Social Networks

consist of 2,538 topics [14]. For each year, we projected the authors onto the subspace spanned by the corresponding 2,538 topics, resulting in ten author-topic matrices used for our task.

We aligned the manifolds corresponding to the year of 2000, 2001 and 2002. Setting of this task is similar to the previous one. $1,000$ author triples were used as corresponding triples in training and $1,538$ other author triples were used as unlabeled documents for both training and testing. The same nine approaches were tested in this experiment. In all approaches, the adjacency graphs were constructed as follows: two authors are neighbors in the graph if they co-authored a paper in that year. The testing scheme is as follows: for each author in a manifold, we retrieve his top k most similar authors in another manifold. The probability that the author himself is included in the top k authors is used to measure the performance of the method. $\alpha = 0.001$, $\beta = 0.01$ and $d = 200$ are used for this test. Results (Figure 4) show that the regular Lasso-based approach outperforms all other approaches. Given an author in one year, it has a 19% probability of finding the true match if we retrieve the most similar author in another year. The Fused Lasso-based manifold alignment also performs better than the previous approaches by a large margin, but a little worse than the regular Lasso. One reason for this is that the order of the topics are almost arbitrary. This result shows that the Fused Lasso-based alignment does not necessarily help improve the system performance, especially when the order of features is not directly related to the learning task.

6 Conclusions

In this paper, we propose a least squares formulation of a class of manifold alignment approaches. A key aspect of this approach is that various regularization techniques can be readily incorporated into the formulation to improve model sparsity and generalization ability. The new approach can make use of prior knowledge, and remove the domain-dependent features automatically during knowledge transfer. It also significantly broadens the scope of manifold alignment techniques and helps speed up the computation by combining l_1 norm optimization with the previous manifold alignment framework. We presented a detailed theoretical and experimental evaluation of our approach, providing results showing useful knowledge transfer from one domain to another. Case studies on cross-lingual information retrieval and social network analysis were also presented.

References

[1] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 120–128, 2006.

[2] C. Fang, M. Kohram, X. Meng, and A. Ralescu. Graph embedding framework for link prediction and vertex behavior modeling in temporal social networks. In *Proceddings of the SIGKDD Workshop on Social Network Mining and Analysis*, 2011.

[3] J. Ham, D. Lee, and L. Saul. Semisupervised alignment of manifolds. In *Proceddings of the International Workshop on Artificial Intelligence and Statistics*, pages 120–127, 2005.

[4] H. Hotelling. Relations between two sets of variates. *Biometrika*, 10:321–377, 1936.

[5] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*, 2005.

[6] S. Lafon, Y. Keller, and R. Coifman. Data fusion and multicue data matching by diffusion maps. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 28(11):1784–1797, 2006.

[7] J. Liu, J. Chen, and J. Ye. Large-scale sparse logistic regression. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 547–556. ACM, 2009.

[8] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections.* Arizona State University, 2009.

[9] L. Sun, S. Ji, and J. Ye. A least squares formulation for a class of generalized eigenvalue problems in machine learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

[10] R. Tibshirani. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(3):273–282, 2011.

[11] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

[12] C. Wang and S. Mahadevan. Manifold alignment using procrustes analysis. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1120–1127, 2008.

[13] C. Wang and S. Mahadevan. Manifold alignment without correspondence. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1273–1278, 2009.

[14] C. Wang and S. Mahadevan. Multiscale analysis of document corpora based on diffusion models. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1592–1597, 2009.

[15] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.

[16] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.