Efficiency Investigation of Manifold Matching for Text Document Classification

Ming Sun\(^a\), Carey E. Priebe\(^b,\)\(^*\)

\(^a\)Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA.
\(^b\)Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218, USA.

Abstract

Manifold matching works to identify embeddings of multiple disparate data spaces into the same low-dimensional space, where joint inference can be pursued. It is an enabling methodology for fusion and inference from multiple and massive disparate data sources. In this paper three methods of manifold matching are considered: PoM, which stands for Multidimensional Scaling (MDS) composed with Procrustes; CCA (Canonical Correlation Analysis) and JOFC (Joint Optimization of Fidelity and Commensurability). We present a comparative efficiency investigation of the three methods for a particular text document classification application.

Keywords: Manifold matching, MDS, procrustes, CCA, JOFC, efficiency, classification.

1. Introduction

1.1. Purpose

In the real world, one single object may have different representations in different domains. For example, the Declaration of Independence has versions translated into different languages. Let \(n\) denote the number of objects \(O_i, \ i = 1, \ldots, n\), and \(K\) be the number of domains. Then we have

\[
x_{i1} \sim \cdots \sim x_{ik} \sim \cdots \sim x_{iK}, \ i = 1, \ldots, n
\]  

where the \(i\)th object \(O_i\) has \(K\) measurements \(x_{ik}, \ k = 1, \ldots, K\); \(x_{ik} \in \Xi_k\) is the representation for object \(O_i\) in space \(\Xi_k\).

*Corresponding author. Phone: 410-516-7200. Fax: 410-516-7459.

Email addresses: msun8@jhu.edu (Ming Sun), cep@jhu.edu (Carey E. Priebe)

The Introduction and Background sections and parts of the methodology and experiment setup are taken from [30] by the same authors. In that paper, Canonical Correlation Analysis and Regularized Canonical Correlation Analysis are compared and contrasted for the same classification task. This paper investigates different manifold matching techniques.
The problem explored in this paper is that for \( m \) new objects \( O'_i \), \( i = 1, \ldots, m \), how to classify their representations \( y_{ik} \in \Xi_k \) given the representations \( y_{i'k'} \in \Xi_{k'} \) with \( k \neq k' \). For this task, \( x_{ik} \), \( x_{i'k'} \), \( i = 1, \ldots, n \), described above are needed to learn the relation between \( \Xi_k \) and \( \Xi_{k'} \) so that we can map data from \( \Xi_k \) and \( \Xi_{k'} \) to a common space \( \chi \). Thus \( x_{ik} \), \( x_{i'k'} \) are the domain relation learning training data. This idea is shown in Figure 1. The domain relation learning training data \( x_{ik}, x_{i'k'} \) are labeled by the filled circles in \( \Xi_k \) and \( \Xi_{k'} \) respectively. The filled squares in \( \Xi_{k'} \) represent the classifier training data \( y_{i'k'} \), which are used to train a classifier \( g \). The classification testing data \( y_{ik} \) is shown by the unfilled square in space \( \Xi_k \). We consider three different domain relation learning methods: PoM, which stands for Multidimensional Scaling (MDS) composed with Procrustes; CCA (Canonical Correlation Analysis) and JOFC (Joint Optimization of Fidelity and Commensurability). We investigate classification performance in the common space \( \chi \) obtained via PoM, CCA and JOFC, training the classifier on \( y_{i'k'} \) and testing on \( y_{ik} \). The focus of this paper is not on optimizing the classifier; rather, we investigate performance for given classifiers (5-Nearest Neighbor, SVM with degree 2 polynomial kernel) as a function of the number of domain relation learning training data observation \( n \) used to learn \( \chi \).

1.2. Summary

The structure of the paper is as follows: Section 2 talks about related work. Section 3 discusses the methods employed, including the manifold matching framework as well as embedding and classification details. Experimental setup and results are presented in section 4. Section 5 is the conclusion.
2. Background

Different methods of transfer learning, multitask learning and domain adaptation are discussed in a recent survey [23]. There are algorithms developed on unsupervised document clustering where training and testing data are of different kinds [14]. The problem explored in this paper can be viewed as a domain adaptation problem, for which the training and testing data of the classifier are from different domains. When the classification is on the text documents in different languages, as described in the later sections of this paper, it is called cross-language text classification. There is much work on inducing correspondences between different language pairs, including using bilingual dictionaries [22], latent semantic analysis (LSA) features [7], kernel canonical correlation analysis (KCCA) [17], etc. Machine translation is also involved in the cross-language text classification, which translates the documents into a single domain [26, 8, 18].

The methods investigated in this paper are closely related to various manifold learning and alignment techniques. There has been intensive work done by many people. Principal Components Analysis (PCA) [13] and Multidimensional Scaling (MDS) [32, 4, 3] are classical linear methods to learn low-dimensional representations for high-dimensional observations. In recent years different non-linear manifold learning methods are developed, including kernel PCA [21], Isomap [31], Locally Linear Embedding (LLE) [27], Laplacian Eigenmaps [2], etc. Regarding manifold alignment, [35] applies procrustes for manifold alignment. The use of diffusion maps is discussed in [16]. [6] learns a common manifold for documents from multilingual corpora such that the embeddings of documents are clustered based on topics. Manifold learning and alignment are also widely used for image analysis [10, 37]. [19] presents a discriminative multi-manifold analysis method to solve the single sample per person problem in face recognition. Manifold alignment can be done in a semisupervised way [9, 34], or without pairwise correspondence information [36, 38]. In [24], unsupervised manifold alignment is conducted based on parameterized distance curves.

3. Method

In this paper, we focus on manifold matching. The whole procedure can be divided into the following steps:

- For each single space $\Xi_k$, calculate the dissimilarity matrix for all domain relation learning training data observations $O_i$.
- Run different manifold matching methods on the dissimilarity matrix to get embedding in a common space $\chi$.
- Pursue joint inference (i.e. classification) in the common space $\chi$.

3.1. Manifold Matching Framework

The framework structure for manifold matching is shown in Figure 2 [20, 25]. For each of the $n$ objects $O_i \in \Xi$, $i = 1, \ldots, n$, there are $K$ representations
Figure 2: Manifold matching model

\[ x_{ik} \in \Xi_k, \ k = 1, \ldots, K \]

Manifold matching works to find \( \rho_1, \ldots, \rho_K \) to map \( x_{i1}, \ldots, x_{iK} \) to a low-dimensional common space \( \chi = \mathbb{R}^d \):

\[ \tilde{x}_{ik} = \rho_k(x_{ik}), \ i = 1, \ldots, n, \ k = 1, \ldots, K. \] (2)

After learning the \( \rho_k \)s, we can map a new measurement \( y_k \in \Xi_k \) into the common space \( \chi = \mathbb{R}^d \) via:

\[ \tilde{y}_k = \rho_k(y_k) \] (3)

This allows joint inference to proceed in \( \mathbb{R}^d \).

3.2. Embedding

The work described in this paper is based on dissimilarity measures. Let \( \delta_k \) denote the dissimilarity measure in the \( k \)th space \( \Xi_k \), and \( \tilde{\delta} \) be the Euclidean distance in the common space \( \mathbb{R}^d \). There are two kinds of mapping errors induced by the \( \rho_k \)s: fidelity error and commensurability error.

Fidelity measures how well the original dissimilarities is preserved in the mapping \( x_{ik} \mapsto \tilde{x}_{ik} \), and the fidelity error is defined as the within-condition squared error:

\[ \epsilon^2_{f_k} = \frac{1}{n} \sum_{1 \leq i < j \leq n} (\tilde{\delta}(\tilde{x}_{ik}, \tilde{x}_{jk}) - \delta_k(x_{ik}, x_{jk}))^2 \] (4)

Commensurability measures how well the matchedness is preserved in the mapping, and the commensurability error is defined as the between-condition
squared error:
\[
\epsilon_{c_1,c_2}^2 = \frac{1}{n} \sum_{1 \leq i \leq n} (\delta(\tilde{x}_{ik_1}, \tilde{x}_{ik_2}))^2
\]  \hspace{1cm} (5)

3.2.1. Procrustes ◦ MDS (P◦M)
Multidimensional Scaling (MDS) \[32, 4, 3\] works to get a Euclidean representation while approximately preserving the dissimilarities. Given the \(n \times n\) dissimilarity matrix \(\Delta_k = [\delta_k(x_{ik}, x_{jk})]\) in space \(\Xi_k\), multidimensional scaling generates embeddings \(\tilde{x}_{ik}' \in \mathbb{R}^{d'}\) for \(x_{ik} \in \Xi_k\), \(i = 1, \ldots, n, k = 1, \ldots, K\), which attempts to optimize fidelity.

For the \(K = 2\) case, multidimensional scaling generates \(n \times d'\) matrices \(\tilde{X}_1'\) from \(\Delta_1\) and \(\tilde{X}_2'\) from \(\Delta_2\). The \(i\)th row vector \(\tilde{x}_{ik}'\) of \(\tilde{X}_k'\) is the multidimensional scaling embedding for \(x_{ik}\).

Procrustes works to get the mapping matrix \(Q^*\) which satisfies
\[
Q^* = \arg \min_{Q^TQ = I} \|\tilde{X}_1' - \tilde{X}_2'Q\|_F.
\]  \hspace{1cm} (6)

For the new data \(y_k, k = 1, 2\), based on \(\delta_k(y_k, x_{ik})\), \(i = 1, \ldots, n\), out-of-sample embedding \[1, 33\] produces \(y_k \mapsto \tilde{y}_k\) with \(d(\tilde{y}_k, \tilde{x}_{ik}')\) being close to \(\delta_k(y_k, x_{ik})\). The final embeddings for \(y_1\) and \(y_2\) in the common space \(\mathbb{R}^d\) are given by \(\tilde{y}_1 = \tilde{y}_1'\) and \(\tilde{y}_2 = ((\tilde{y}_{2}')^TQ^*)^T\).

P◦M optimizes fidelity without regard for commensurability \[25\].

3.2.2. Canonical Correlation Analysis (CCA)
Canonical correlation analysis is applied to the multidimensional scaling results. Canonical correlation works to find \(d' \times d\) matrices \(U_1 : \tilde{X}_1' \mapsto \tilde{X}_1\) and \(U_2 : \tilde{X}_2' \mapsto \tilde{X}_2\) as the linear mapping method to maximize correlation for the mappings into \(\mathbb{R}^d\).

For new data \(y_k, k = 1, 2\), out-of-sample embedding for multidimensional scaling generates \(d'\) dimensional column vector \(\tilde{y}_k'\). The final embeddings in the common space \(\mathbb{R}^d\) are given by \(\tilde{y}_1 = U_1^T\tilde{y}_1'\) and \(\tilde{y}_2 = U_2^T\tilde{y}_2'\).

Canonical correlation analysis optimizes commensurability without regard for fidelity \[25\]. For our work, first we use multidimensional scaling to generate a fidelity-inspired Euclidean representation, and then we use canonical correlation analysis to enforce low dimensional commensurability.

3.2.3. Omnibus Embedding (JOFC)
The omnibus embedding method described in \[25, 20\] jointly optimizes fidelity and commensurability. Given the \(n \times n\) dissimilarity matrices \(\Delta_1 \in \Xi_1\) and \(\Delta_2 \in \Xi_2\), the \(2n \times 2n\) omnibus dissimilarity matrix \(M\) is constructed as shown in Figure 3.

The off-diagonal block is \(L = (\Delta_1 + \Delta_2)/2\). The embeddings \(\tilde{x}_{i1}, \tilde{x}_{i2} \in \mathbb{R}^d\) can be obtained by running multidimensional scaling on \(M\) directly.
\[
M_{2n \times 2n} = \begin{bmatrix}
\Delta_1 & L \\
L^T & \Delta_2
\end{bmatrix}
\]

Figure 3: Omnibus dissimilarity matrix

Similar to PoM and CCA, for the new data \(y_k, k = 1, 2\), its embedding in the common space \(\tilde{y}_k \in \mathbb{R}^d\) can be obtained directly from out-of-sample embedding based on \(\delta_k(y_k, x_{ik}), i = 1, \ldots, n\).

3.3. Classification

Given the measurements of \(m\) new data points \(y_{ik} \in \Xi_k, i = 1, \ldots, m\), for the classification of \(y_{ik}\), we consider the problem in which there are no training data available in \(\Xi_k\) and we must borrow training data from another space \(\Xi_{k'}\). Let \(y'_{ik'} \in \Xi_{k'}\) denote the training data. The classification procedure begins with projecting both testing data \(y_{ik}\) and training data \(y'_{ik'}\) to a common space \(\mathbb{R}^d\). The manifold matching methods PoM, CCA and JOFC described in section 3.2 embed \(y_{ik} \mapsto \tilde{y}_{ik} \in \mathbb{R}^d\) and \(y'_{ik'} \mapsto \tilde{y'}_{ik'} \in \mathbb{R}^d\). As a result, a classifier is trained on \(\tilde{y'}_{ik'}\) and tested on \(\tilde{y}_{ik}\). This problem is motivated by the fact that in many situations there is a lack of training data in the space where the testing data lie. We will discuss the classification problem in more details in Section 4.4.

3.4. Efficiency Investigation

We investigate the effect of the number of domain relation learning training data observations on the classification performance. That is, given a subset of available domain relation learning training data, we are interested in how different manifold matching techniques perform in the cross-domain classification task. The classification accuracy is expected to improve with increasing amount of domain relation learning training data. To achieve the same classification accuracy, the method using the smallest amount of domain relation learning training data is identified as the most efficient one.

4. Experiments

In this section the experimental details are described. Section 4.1 describes the dataset used for our experiments. Section 4.2 discusses the dissimilarity matrix calculation. Our method for choosing proper embedding dimensions is presented in Section 4.3. The classification setting and results are described and analyzed in Section 4.4 and Section 4.5.
4.1. Dataset

Our experiments apply canonical correlation analysis and its generalization to text document classification. The dataset is obtained from wikipedia, an open-source multilingual web-based encyclopedia with around 19 million articles in more than 280 languages. Each document may have links pointing to other documents in the same language which explain certain terms in its content as well as the documents in other languages for the same subject. Articles of the same subject in different languages are not necessarily the exact translations of one another. They can be written by different people and their contents can differ significantly.

English articles within a 2-neighborhood of the English article "Algebraic Geometry" are collected. The corresponding French documents of those English ones are also collected. So this data set can be viewed as a two space case: \( \Xi_1 \) is the English space and \( \Xi_2 \) is the French space. There are in total 1382 documents in each space. That is, \( z_{1,1}, \ldots, z_{1382,1} \in \Xi_1, \) and \( z_{1,2}, \ldots, z_{1382,2} \in \Xi_2. \) Note that \( z_{ik}, \ i = 1, \ldots, 1382, \ k = 1, 2 \) includes both domain relation learning training data \( x_{ik}, \ i = 1, \ldots, n \) and new data points \( y_{ik}, \ i = 1, \ldots, m \) \((m + n = 1382)\) used for classification training and testing.

All 1382 documents are manually labeled into 5 disjoint classes \((0 - 4)\) based on their topics. Topics are category, people, locations, date and math respectively. Documents in classes 0, 2, 4 are the domain relation learning training data \( x_{ik}, \ i = 1, \ldots, n, \ k = 1, 2. \) There are in total 819 documents in those 3 classes \((n = 819).\) The rest 563 \((m = 563)\) documents in classes 1, 3 are the new data \( y_{ik}, i = 1, \ldots, m, k = 1, 2. \) They are used to train a classifier and run the classification test.

4.2. Dissimilarity Matrix

The method described in section 3.2 starts with the dissimilarity matrix. For our work two different kinds of dissimilarity measures are considered: text content dissimilarity matrix \( \Delta_k^t \) and graph topology dissimilarity matrix \( \Delta_k^g. \) Both matrices are of dimension 1382 \times 1382, containing the dissimilarity information for all data points \( z_{1k}, \ldots, z_{1382k}. \)

Graphs \( G_k(V, E_k) \) can be derived from model the dataset; \( V \) represents the set of vertices which are the 1382 wikipedia documents, and \( E_k \) is the set of edges connecting those documents in language \( k. \)

The \((i, j)\) entry \( \Delta_k^t(i, j) \in \Delta_k^t \) is the number of steps on the shortest path from document \( i \) to document \( j \) in \( G_k. \) In the English space \( \Xi_1, \) \( \Delta_k^t(i, j) \in \{0, \ldots, 4\}, \) where the 4 comes from the 2-neighborhood document collection. In the French space \( \Xi_2, \) \( z_{12} \) is the French corresponding document for the English one \( z_{11} \in \Xi_1, \) and \( \Delta_k^t(i, j) \in \Delta_k^t \) depends on the French graph connections. It is possible that \( \Delta_k^t(i, j) \neq \Delta_k^t(i, j). \) At the extreme end, \( \Delta_k^t(i, j) = \infty \) when \( z_{12} \) and \( z_{j2} \) are not connected. We set \( \Delta_k^t(i, j) = 6 \) for \( \Delta_k^t(i, j) > 4. \) Because the upperbound of the English graph topology dissimilarity is 4, this choice makes the French graph topology dissimilarity comparable to the English one. On one hand, the French graph topology dissimilarity upperbound is larger since the...
Figure 4: Square root of eigenvalues for embedding’s variance matrix (all data used)

actual French graph topology dissimilarity can be larger than the English one. On the other hand, the French graph topology dissimilarity upperbound is set to be a value not too big to make sure it does not overwhelm the embedding for the French graph topology dissimilarity matrix. Optimal pre-processing to put the dissimilarities on the same footing is the subject of ongoing investigation.

\[ \Delta_t^k(i,j) \in \Delta_k \] is based on the text processing features for documents \( z_{ik}, z_{jk} \in \Xi_k \). Given the feature vectors \( f_{ik}, f_{jk}, \Delta_k^k(i,j) \) is calculated by the cosine dissimilarity \( \Delta_k^k(i,j) = 1 - \frac{f_{ik} \cdot f_{jk}}{\|f_{ik}\|_2 \|f_{jk}\|_2} \). For our experiments, we consider term frequency-inverse document frequency (TFIDF) features [28] as \( f \).

We use multidimensional scaling to embed into Euclidean space \( \mathbb{R}^d \) while approximating dissimilarity information; in this space, Euclidean distance is appropriate.

4.3. Embedding Dimension Selection

To choose the dimension \( d \) for the common space \( \mathbb{R}^d \), we pick a sufficiently large dimension and embed \( \Delta_k^k \) and \( \Delta_k^g \) via multidimensional scaling. The square root information of the embedding’s covariance matrix is shown in Figure 4.

Based on the plots in Figure 4, we choose the dimension \( d = 15 \) (dimension of the joint space \( \chi = \mathbb{R}^d \)), which is low but preserves most of the variance [13]. This model selection choice of dimension is an important issue in its own
right; for this paper, we fix $d = 15$ throughout. The focus of this paper is not model selection. Our selection of $d$ generates satisfactory experimental results, as described in Section 4.5. These results are illustration of performance to be expected when incorporating a proper model selection methodology.

For canonical correlation analysis, since it requires to multidimensional scale the dissimilarity matrices to $d'$ at the beginning, as described in section 3.2, when we choose different number $n'$ of domain relation learning training documents, $d'$ depends on $n'$. We choose the value of $d'$ as large as possible while avoiding numerical underflow. The values of $d'$ with different $n'$ are shown in Table 1. The second column indicates what percentage of the total manifold matching training data $x_{ik}$ is used.

### 4.4. Classification Setting

The classifiers used in the experiment are $\kappa$-nearest neighbor ($\kappa$-NN) [29] and support vector machine (SVM) [5]. For $\kappa$-NN, the class label of the test data is assigned by the majority class label of the $\kappa$ closest training data points. The distance used is the usual Euclidean distance. For our experiments we use 5-nearest neighbor classifier. SVM sets the separation hyperplane via maximizing the margin. By using the kernel method, SVM can provide non-linear discriminates. We use a polynomial kernel with degree 2.

There are 563 new data points $y_{ik}$ in classes 1 and 3. Class 1 has 372 data points, and the remaining 191 have class label 3. For each $n'$ in Table 1, we randomly sample $n'$ out of the total 819 domain relation learning training documents to learn the common space $\mathbb{R}^{d'}$ into which we project the new data points. The classification is run in a leave-one-out way. We use 200 Monte Carlo replicates to calculate the average performance.

<table>
<thead>
<tr>
<th>$n'$</th>
<th>% of $n$</th>
<th>$d'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>82</td>
<td>10%</td>
<td>40</td>
</tr>
<tr>
<td>164</td>
<td>20%</td>
<td>80</td>
</tr>
<tr>
<td>246</td>
<td>30%</td>
<td>100</td>
</tr>
<tr>
<td>328</td>
<td>40%</td>
<td>100</td>
</tr>
<tr>
<td>410</td>
<td>50%</td>
<td>150</td>
</tr>
<tr>
<td>491</td>
<td>60%</td>
<td>150</td>
</tr>
<tr>
<td>573</td>
<td>70%</td>
<td>150</td>
</tr>
<tr>
<td>655</td>
<td>80%</td>
<td>200</td>
</tr>
<tr>
<td>737</td>
<td>90%</td>
<td>200</td>
</tr>
<tr>
<td>819</td>
<td>100%</td>
<td>200</td>
</tr>
</tbody>
</table>
The method described in section 3.2 generates the embeddings $\tilde{y}_{ik} \in \mathbb{R}^{15}$, $i = 1, \ldots, 563$, $k = 1, 2$. Because there are two kinds of dissimilarity matrices considered, we have $\Delta_k^1 \mapsto \tilde{y}_{ik}^1$ and $\Delta_k^2 \mapsto \tilde{y}_{ik}^2$. The training and testing data can be chosen from not only different spaces (i.e. English space and French space), but also from different dissimilarity measures (i.e. text content dissimilarity and graph topology dissimilarity). Classification results are shown in Figure 5.

4.5. Classification Results

In Figure 5, the $x$-axis label $S$ indicates what proportion of the total $n$ data points are used for domain relation learning training, that is, $S = \frac{n'}{n}$; the $y$-axis is classification accuracy. GE means the embeddings from English graph
topology dissimilarity matrix $\Delta^g_1$ are used. Similarly, GF and TF represent the embeddings from French graph topology dissimilarity matrix $\Delta^g_2$ and French text content dissimilarity matrix $\Delta^f_2$ respectively. GF$\rightarrow$GE means $\Delta^g_2$ is used for classifier training and $\Delta^f_2$ is for testing. TF$\rightarrow$GE means the classifier is trained on $\Delta^f_1$ and tested on $\Delta^f_2$.

In Figure 5a, $\Delta^g_2$ is used for training and $\Delta^f_1$ is for testing, thus $x_{ik}^g$, $i = 1, \ldots, n'$, $k = 1, 2$ are employed to learn the manifold matching methods. The solid circle curve is for PoM, while the dashed triangle and dotted plus curves represent CCA and JOFC respectively. For each test data point $y_{i'}^g$, $i' \in \{1, \ldots, m\}$, the 5-NN classifier is trained on $y_{i'+1}^g$, $i'+1 = 1, \ldots, i-1, i+1, \ldots, m$, and the classification accuracy is calculated as $m'/m$, where $m'$ is the number of correctly classified testing data points. For each $n'$, 200 Monte Carlo replicates are run to randomly sample $n'$ out of the total $n$ domain relation learning training data points $x_{ik}^g$, $i = 1, \ldots, n$. The average accuracy is plotted; the standard errors are available via bootstrap resampling.

Figure 5c is similar to Figure 5a except the training data is from $\Delta^f_1$ instead of $\Delta^g_2$. Since $\Delta^f_1$ and $\Delta^f_2$ are within different ranges, prescaling is needed, which is done by $\Delta^f_1 = \Delta^f_1 \frac{\|\Delta^f_1\|_F}{\|\Delta^f_2\|_F}$.

Similarly, Figure 5b and Figure 5d show the classification results of PoM, CCA and JOFC using SVM with degree 2 polynomial kernel. Figure 5b uses $\Delta^g_2$ to classify $\Delta^f_1$, while Figure 5d uses $\Delta^f_1$ to classify $\Delta^f_2$.

Based on the results shown in Figures 5a, 5b, 5c and 5d, we can see as a general guideline JOFC outperforms both PoM and CCA with regard to the cross-language text document classification. The superior performance of JOFC comes from its ability to jointly preserve fidelity and commensurability in the mapping. We do not claim that this dominance holds uniformly. Indeed, there are exception points in the plots of Figure 5; for example, when there are few domain relation learning training data (Figures 5a, 5b), or for the case when all domain relation learning training data are used (Figure 5c). But as a general guideline JOFC is a better choice compared to PoM and CCA in terms of

<table>
<thead>
<tr>
<th></th>
<th>GF $\rightarrow$ GE</th>
<th>TF $\rightarrow$ GE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$S = 10%$</td>
<td>$S = 10%$</td>
</tr>
<tr>
<td></td>
<td>$d' = 40$</td>
<td>$d' = 200$</td>
</tr>
<tr>
<td>5-NN</td>
<td>PoM</td>
<td>CCA</td>
</tr>
<tr>
<td></td>
<td>PoM</td>
<td>67.20% $\pm$ 0.11%</td>
</tr>
<tr>
<td></td>
<td>CCA</td>
<td>61.95% $\pm$ 0.12%</td>
</tr>
<tr>
<td></td>
<td>JOFC</td>
<td>66.08% $\pm$ 0.15%</td>
</tr>
<tr>
<td>SVM</td>
<td>PoM</td>
<td>63.23% $\pm$ 0.10%</td>
</tr>
<tr>
<td></td>
<td>CCA</td>
<td>64.10% $\pm$ 0.07%</td>
</tr>
<tr>
<td></td>
<td>JOFC</td>
<td>61.76% $\pm$ 0.12%</td>
</tr>
</tbody>
</table>
classification performance and efficiency.

With increasing amount of domain relation learning training data, the classification performance of JOFC improves, while for PoM and CCA, their classification performance does not necessarily increase with more domain relation learning training data, as shown in Figure 5a.

In the case of using $\Delta^2_g$ to classify $\Delta^1_g$, for both PoM and JOFC, 5-NN has a higher classification accuracy than SVM with degree 2 polynomial kernel. But for CCA, 5-NN gets lower classification accuracy for certain cases. In the case of using $\Delta^1_t$ to classify $\Delta^1_g$, for all PoM, CCA and JOFC, 5-NN yields better classification performance than SVM.

Table 2 shows the classification accuracy of various methods for $S = 10\%$ and $S = 100\%$. The standard error is obtained via bootstrapping for 1000 samples.

5. Conclusion

In this paper we investigate the performance of three manifold matching methods (PoM, CCA and JOFC) on a cross-language text classification task. We show their performance with manifold matching training data from different domains and different dissimilarity measures, and we also investigate their efficiency by choosing different amounts of domain relation learning training data. In our framework each document is assigned a single topic. The case of multi-topic document assignment from Probabilistic Latent Semantic Analysis (PLSA) or Latent Dirichlet Allocation (LDA) is an interesting extension for ongoing investigation. The experimental results indicate that JOFC, which jointly optimizes fidelity and commensurability, outperforms both PoM and CCA. These results provide significant impetus for further investigation of jointly optimizing fidelity and commensurability for general cross-language inference.

In [30], a regularized version of CCA was investigated for the same classification task; the results presented there demonstrate that JOFC is superior to regularized CCA for $GF \rightarrow GE$, but the regularized CCA is superior for $TF \rightarrow GE$. The specific comparative analysis of JOFC vs regularized CCA remains a topic of investigation.

References


