2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Random Subspaces NMF for Unsupervised Transfer Learning

REDKO Ievgen and BENNANI Younès

***Abstract*— In this paper we propose a new unsupervised transfer learning approach which aims at finding a partition of unlabeled data in target domain using the knowledge obtained from clustering a source domain unlabeled data. The key idea behind our method is that finding partitions in different feature's subspaces of a source task can help to obtain a more accurate partition in a target one. From the set of source partitions we select only k nearest neighbors using some measure of similarity. Finally, multi-layer non-negative matrix factorization is performed to obtain a partition of objects in target domain. Experimental results show high potential and effectiveness of the proposed technique.**

I. Introduction

Machine Learning and data mining have already shown significant success in many areas of knowledge engineering, including classification, regression and clustering. However, many learning methods work well only under a common assumption: the training and test data are from the same feature space and the same distribution. When the distribution changes, most statistical models must be rebuilt from new collected data. In many real-world applications, it is expensive or impossible to collect new data needed to reconstruct the learning models. Therefore it is necessary to develop approaches to reduce the need and the effort to collect new data. In such cases, the transfer of knowledge and transfer learning between domains could be desirable. Many examples of knowledge engineering where transfer learning can really prove beneficial can be found. The key idea behind transfer learning is that learning one distribution can help to learn the other using the shared common latent structure as the bridge for knowledge transfer. Transfer learning involves two interrelated problems, aiming at using the knowledge acquired in a set of tasks and improve performance for another related task. Specifically, learning by transferring to a certain target task - the task on which performance is measured - is very dependent on the learning of auxiliary tasks.

A. Transfer Learning

Transfer learning is a widely known technique that was generally inspired by the ability of a human being to detect and to use previously gained knowledge in one area for efficient learning in another. The definition of transfer learning was given in [1] as:

Given a source domain D_S and a learning task T_S, a target domain D_T and a target task T_T, transfer learning aims to help improve the learning performance in D_T using knowledge gained from D_S and T_S , where $D_S \neq D_T$ and $T_S \neq T_T$.

It is worth mentioning that the notion of a domain in this definition is usually given by a pair of objects $D = (X, P(X))$. It means that the condition $D_S \neq D_T$ implies either $X_S \neq X_T$ or $P_S(X) \neq P_T(X)$. The same thing for a task: $T = (Y, P(Y|X))$ and $T_S \neq T_T$ implies either $Y_S \neq Y_T$ or $P_S(Y|X) \neq P_T(Y|X)$.

There are three types of transfer learning:

- supervised transfer learning (when data is labeled in both target and source learning tasks
- semi-supervised transfer learning (data is labeled only in the source learning task
- unsupervised transfer learning (no labeled data in source and in target learning tasks

According to the above mentioned survey, the number of methods dealing with the first two settings of transfer learning drastically exceeds the number of articles dedicated to the last one. Indeed, to the best of our knowledge there are only two algorithms of unsupervised transfer learning: self-taught clustering (STC) presented in [2] and transferred dimensionality reduction (TDA) presented in [3] that were proposed to solve this problem. Little research that has been done in this field of machine learning can be explained by the fact that unsupervised transfer learning is an extreme case of the transfer learning paradigm which, nevertheless, occurs in numerous real-world applications. Thus, unsupervised transfer learning becomes a topic of an ongoing interest for further researches.

B. Subspaces approaches

In this paper we propose a new approach called Random Subspace NMF (RS-NMF) for unsupervised transfer learning. This approach combines the sampling technique in the feature space and the ensemble idea.

Bagging [4], a name derived from bootstrap aggregation, was the first effective method of ensemble learning and is one of the simplest methods of arching. The meta-algorithm, which is a special case of model averaging, was originally designed for classification and it is usually applied on decision tree models, but it can be used with any type of model, whether for classification or regression. The random subspace principle is an interesting method of combining models. Learning machines are trained on randomly chosen subspaces of the original input space (i.e. the training set is sampled in the feature space). The outputs of the models are then combined, usually by a simple majority vote. Several researchers have tried to use this principle for many classifiers. In the supervised case, fast algorithms, such as

REDKO Ievgen and BENNANI Younès are with Laboratoire d'Informatique de Paris-Nord, CNRS (UMR 7030), Université Paris 13, Sorbonne Paris Cité, F-93430, Villetaneuse, France(email: {ievgen.redko, younes.bennani}@lipn.univ-paris13.fr).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

Decision trees are commonly used with ensembles, although slower algorithms can benefit from ensemble techniques as well. Examples of such approach can be found in [5] [6], where the classifier consists of multiple trees constructed systematically by pseudo-randomly selecting subsets of components of the feature vector, that is, trees constructed in randomly chosen subspaces. The essence of the method is to build multiple trees in randomly selected subspaces of the feature space (Random Forest). Trees in, different subspaces generalize their classification in complementary ways, and their combined classification can be monotonically improved.

In the unsupervised case, subspace clustering is an extension of traditional clustering that seeks to clusters in different subspaces within a dataset. Traditional clustering algorithms consider all of the dimensions of an input dataset in an attempt to learn as much as possible about each instance described. In high dimensional data, however, many of the dimensions are often irrelevant. These irrelevant dimensions confuse clustering algorithms by hiding clusters in noisy data. In very high dimensions the concept of distance becomes less precise as the number of dimensions grows, since the distance between any two points in a given dataset converges. The discrimination of the nearest and farthest point in particular becomes meaningless. Subspace clustering is the task of detecting all clusters in all subspaces. This means that a point might be a member of multiple clusters, each existing in a different subspace. Subspace clustering algorithms localize the search for relevant dimensions also wing them to clusters that exist in multiple, possibly overlapping subspaces. Therefore, there is a need to simultaneously cluster the data into multiple subspaces and find a low-dimensional subspace fitting each group of points. This problem, known as subspace clustering, has found numerous applications. A number of approaches to subspace clustering have been proposed in the past two decades. A review of methods from the data mining community can be found in [7]. In our approach RS-NMF, the subspace paradigm will be very beneficial for transfer learning. Indeed, this paradigm will break down the knowledge of the source space into subspaces of knowledge and make a selection of the most relevant knowledge for transfer to the target space. The principle of our approach is close to the multi-view approach proposed in [8] [9].

The rest of this paper is organized as follows: in section 2 we will briefly introduce basic notations of standard nonnegative matrix factorizations, in section 3 we are introducing Random Subspace NMF (RS-NMF) for unsupervised transfer learning. We will summarize the results in section 4. Finally, we will point out some ideas about the future extension of our method in section 5.

II. Preliminary knowledge

A. Standard NMF and Convex NMF

A standard NMF [10] seeks the following decomposition:

$$X \simeq FG^T, X \in \mathbb{R}^{n\times m}, F \in \mathbb{R}^{n\times k}, G \in \mathbb{R}^{m\times k}$$

$$X, F, G \geq 0.$$

where

- X is an input data matrix
- columns of F can be considered as basis vectors
- columns of G are considered as cluster assignments for each data object
- k is the desired number of clusters

To develop Convex NMF (C-NMF) [11], we consider the factorization of the following form:

$$X \simeq FG^T = XWG^T, X \in \mathbb{R}^{n\times m}, W \in \mathbb{R}^{m\times k}, G \in \mathbb{R}^{m\times k}$$

$$X, W, G \geq 0.$$

where the column vectors of F lie within the column space of X:

$$F = XW.$$

B. Symmetric NMF

The nonnegative symmetric factorization (Sym-NMF) [12] of the similarity matrix A is formulated as following optimization problem:

$$A \simeq GG^T, A \in \mathbb{R}^{n\times n}, G \in \mathbb{R}^{n\times k}$$

where A is a similarity matrix calculated based on an arbitrary similarity measure, n is a number of objects, k is the number of clusters requested. Compared to NMF, SymNMF is more flexible in terms of choosing similarities for the data points. Any similarity measure that well describes the inherent cluster structure can be chosen. In fact, the formulation of NMF can be related to SymNMF when $A = X^TX$ in the formulation. This means that NMF implicitly chooses inner products as the similarity measure, which might not be suitable to distinguish different clusters.

C. Multilayer NMF

In order to improve performance of the NMF, especially for illconditioned and badly scaled data and also to reduce risk of getting stuck in local minima of a cost function, a simple hierarchical and multistage procedure to perform a sequential decomposition of nonnegative matrices was developed in [13]. In the first step, a basic decomposition

$$X \simeq F_1G_1$$

is performed using any available NMF algorithm. In the second stage, the results obtained from the first stage are used to perform the similar decomposition:

$$G_1 \simeq F_2G_2$$

using the same or different update rules, and so on. The decomposition takes into account only the last achieved components. The process can be repeated arbitrary many times until some stopping criteria are satisfied. In each step, gradual improvements of the performance are usually obtained. Thus, the Multilayer NMF is of the following form:

$$X \simeq F_1F_2...F_LG_L,$$

with the basis matrix defined as $F = F_1F_2...F_L$. Physically, this means that we build up a system that has many layers or cascade connection of L mixing subsystems.

III. Proposed approach

We would like to propose a very simple and yet effective way to generate a sequence of partition matrices for a given data set further used to learn a sequence of prototype matrices that can be applied as a bridge for transfer learning between two tasks.

A. Random Subspaces NMF

The idea of Random Subspace NMF (RS-NMF) is to perform the knowledge "decomposition" of a given data set $X \in \mathbb{R}^{n \times m}$ that basically consists in finding a sequence of partition matrices $\{G_i\}_{i=1}^{M}$ that were calculated on the subspaces of X.

We randomly choose $\sqrt{m}$ features and perform any arbitrary type of NMF for the reduced matrices $\{X_{ss_i}\}_{i=1}^{M}$. We obtain a sequence of partition matrices $\{G_i\}_{i=1}^{M}$ that can be used further for majority voting or some other consensus technique.

In order to show that these decompositions can produce a better clustering result, we used 30-fold cross-validation for 8 data sets from UCI machine learning repository and evaluated the results using two different measures: entropy and purity [14]. These are the standard measures of clustering quality. Entropy measures how the various semantic classes are distributed within each cluster. Given a particular cluster S_r of size n_r , the entropy of this cluster is defined to be:

$$E(S_r) = -\frac{1}{q}\sum_{i=1}^{q}\frac{n_r^i}{n_r}log\frac{n_r^i}{n_r}$$

where q is the number of classes in the dataset, and n_r^i is the number of elements of the ith class that were assigned to the rth cluster. The entropy of the entire clustering is then the sum of the individual cluster entropies weighted according to the cluster size:

$$entropy = \sum_{r=1}^{k}\frac{n_r}{n}E(S_r)$$

The clustering solution is perfect if clusters only contain observations from one single class; in that case the entropy of the clustering solution is zero. Smaller entropy values indicate better clustering solutions.

Using the same mathematical definitions, the purity of a cluster is defined as:

$$Pu(S_r) = \frac{1}{n_r}\max_i n_r^i$$

The purity gives the fraction of the overall cluster size that the largest class of elements assigned to that cluster represents. The purity of the clustering solution is then again the weighted sum of the individual cluster purities:

$$purity = \sum_{r=1}^{k}\frac{n_r}{n}Pu(S_r)$$

Larger purity values indicate better clustering solutions. Relatively high number of iterations for cross-validation can be explained by the numerical instability of NMF. In Table 1 and 2, the values representing average purity and entropy for NMF, Symmetric NMF and the maximum achieved purity using RS-NMF are reported.

TABLE I
NMF, Sym-NMF and RS-NMF purity values on various data sets

Data set	NMF	Sym-NMF	RS-NMF
Iris(3)	0.6600	0.6667	**0.8000**
Ecoli(8)	0.5685	0.6488	**0.7470**
Wine(3)	0.5843	0.3989	**0.6404**
Glass(7)	0.5187	0.5467	**0.6308**
Australian(2)	0.5783	0.5551	**0.6319**
Hepatitis(2)	0.7935	0.7935	**0.7935**
Breast Wisconsin(2)	0.6552	0.6552	**0.7997**
Yeast(10)	0.4178	0.4212	**0.4549**

TABLE II
NMF, Sym-NMF and RS-NMF entropy values on various data sets

Data set	NMF	Sym-NMF	RS-NMF
Iris(3)	0.3730	0.3385	**0.2963**
Ecoli(8)	0.2973	0.3358	**0.2336**
Wine(3)	0.6426	0.9167	**0.6158**
Glass(7)	0.4831	0.4648	**0.4195**
Australian(2)	0.9212	0.9379	**0.9209**
Hepatitis(2)	0.6438	0.6405	**0.6275**
Breast Wisconsin(2)	0.7460	0.6917	**0.6867**
Yeast(10)	0.5844	0.5924	**0.5749**

In order to facilitate the analysis we also plotted all the purity and entropy values in Figure 1 and 2.

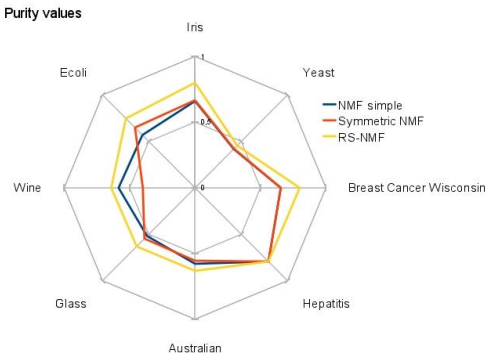

Fig. 1. Radar plot of purity values

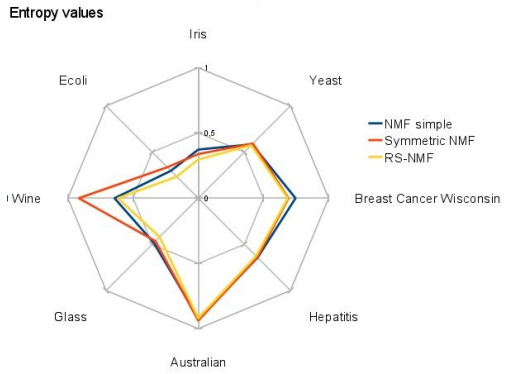

Fig. 2. Radar plot of entropy values

It is easy to see that performance on each data set can be increased by choosing the features that are most pertinent to a given classification task.

B. RS-NMF for Transfer Learning

As it was said before, we will consider a situation where only unlabeled data for both tasks is available. Let us consider two tasks T_S and T_T defined by two matrices X_S = $(x_{s_1}, x_{s_2}, ..., x_{s_n})$ and X_T = $(x_{t_1}, x_{t_2}, ..., x_{t_n})$ where each line represents one object from the data of the corresponding task. For the sake of convenience, we will consider two matrices with the same number of lines. This inconvenience can be overcome in two ways: by sub-sampling the bigger dataset or by using any kind of a bootstrap to increase the size of the smaller dataset.

We will now apply the method described above as an initialization step for a new transfer learning approach. After calculating a sequence of partition matrices $\{G_i\}_{i=1}^{M}$ for source data set X_S, we search a partition of data of a target data set X_T by using any arbitrary form of NMF (for example, C-NMF).

$$X_T \simeq X_T W_T G_T^T.$$

In this expression, the prototype matrix can be calculated as $P_T = X_T W_T$.

C. Learning "link" matrices

In order to construct a sequence of learned weighting matrices that represent the associations between clusters in different subspaces we calculate the correlation between all the matrices $\{G_i\}_{i=1}^{M}$ and G_T. We use a simple correlation function defined as

$$corr(X, Y) = \frac{cov(X, Y)}{\sigma_X \sigma_Y}$$

and select k nearest neighbors of G_T.

At this step we obtain a reduced sequence $\{G_i\}_{i=1}^{k}$. We take each of the chosen matrices and perform the NMF of the following form:

$$G_i = W_i G_i^*, G_i \in \mathbb{R}^{k \times n}, W \in \mathbb{R}^{k \times k}, G_i^* \in \mathbb{R}^{k \times n}$$

$$\forall\ i = 1...k.$$

After doing that, we have a sequence of "link" matrices $\{W_i\}_{i=1}^{k}$ calculated using NMF with the partition matrices from source task that are nearest to the partition of data in the target task. The idea behind constructing this sequence of "link" matrices is that they capture the relationships between clusters and thus reflect the structure of a data set. Simply using a sequence of partition matrices is not enough because they are closely related to the data itself but what we try to do is to discover the common parts in structures of both data sets to adapt them using NMF and to use further as a link.

D. Multilayer NMF with learned "link" matrices

Finally, we have a sequence of matrices $\left\{P_T, \{W_i\}_{i=1}^{k}\right\}$ that we will use in a final stage. In our opinion it is very important to use the initial matrix P_T that can be seen as a guide of the transfer learning process. We recall that Multilayer NMF is of the following form:

$$X \simeq F_1 F_2 ... F_L G_L,$$

We perform Multilayer NMF with "link" matrices fixed to our learned "link" matrices. At the end, our Multilayer NMF takes the following form:

$$X_T \simeq P_T W_1 ... W_k G_T^*,$$

where G_T^* is a final result of our algorithm after the transfer process.

IV. Experimental results

Here we will describe all the experiments that were made in order to show the efficiency of our approach. We recall that we work in the unsupervised setting and so we are not able to use purity and entropy used to validate RS-NMF.

A. Clustering evaluation criteria

There are two classes of clustering evaluation metrics: internal and external clustering evaluation indexes. Speaking about unsupervised clustering, we can only use internal metrics because they are based only on the information intrinsic to the data alone. Among them, the most referenced in literature are the following ones: the Bayesian information criteria, Calinski-Harabasz index, Davies-Bouldin index(DBI), Silhouette index, Dunn index and NIVA index. To estimate the effectiveness of clustering we will use two of the most effective (according to [15]) clustering indexes, the Dunn's index and Calinski-Harabasz index. Dunn's index scheme is calculated as follows:

$$Dunn = \min_{1 \le i \le c} \left\{ min \left\{ \frac{d(c_i, c_j)}{\max_{1 \le k \le c}(d(X_k))} \right\} \right\}$$

where c denotes the number of clusters, i and j are cluster labels, $d(c_i, c_j)$ defines the intercluster distance between clusters X_i X_j; $d(X_k)$ represents the intracluster of X_k. This index aims to identify sets of clusters that are compact

and well separated. Large values of Dunn's index indicates a "better" clustering solution.

Calinski-Harabasz evaluation scheme is given by the following expression:

$$CH = \frac{trace(S_B)}{trace(S_W)} \frac{n_p - 1}{n_p - k}$$

where S_B is a between-cluster scatter matrix, S_W is the internal scatter matrix, n_p is a number of clustered samples and k is a number of clusters. Large values of Calinski-Harabasz index stands for more accurate clustering.

B. Datasets

The following data sets from UCI machine learning repository were chosen:

- Iris dataset: this is perhaps the best known database to be found in the pattern recognition literature. Fisher's paper is a classic in the field and is referenced frequently to this day. The dataset contains 3 classes of 50 instances each described by 4 features (sepal length, sepal width, petal length, petal width), where each class refers to a type of iris plant (Setosa, Versicolour, Virginica). One class is linearly separable from the other 2; the latter are not linearly separable from each other [16].
- Wine dataset: these data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. This dataset consists of 178 instances described by 13 features and divided in three classes [17].
- Glass dataset: the glass identification database is composed of 214 examples with 9 variables, the data is divided in seven classes [18].
- Yeast dataset: this database contains information about a set of Yeast cells and consists of 1484 observations divided into 10 classes and described by 10 features. The task is to determine the localization site of each cell [19].
- Hepatitis dataset: the clynical observations of people that were suffering from hepatitis is composed of 155 examples with 19 variables, the data is divided in 2 classes [20].
- Ecoli dataset: the prokaryotic gram-negative bacterium Escherichie Col (E.coli) 336 patterns data are classified to eight classes and described by 8 features. Classes of this data set are drastically imbalanced [19].
- Australian Credit Approval dataset: this database concerns credit card applications. There are 6 numerical and 8 categorical attributes. All data represents two classes of applications: accepted and declined [21].
- Breast Cancer Wisconsin: this breast cancer databases was obtained from the University of Wisconsin Hospitals with 10 attributes divided into two classes [22].
- Heart disease: these data are the results of 303 observation of heart diseases. This database is composed of 13 fetures and is divided into 5 classes [23].

In Table 3 we can see the Dunn's index values of the experimental tests of our approach for transfer between two different domains. We indicate also k - the number of learned prototype matrices. We compare the results obtained using our approach with the partition given by C-NMF.

TABLE III
DUNN'S INDEX VALUES FOR TRANSFER BETWEEN DIFFERENT DOMAINS

Source Task → Target Task	No Transfer **C-NMF**	Transfer **RS-NMF**	k
Iris → Wine	0.0033	**0.0271**	1
Wine → Iris	0.1650	**2.9523**	9
Wine → Glass	0.0118	**0.0134**	2
Glass → Wine	0.0035	**0.0385**	6
Iris → Glass	**0.0126**	0.0112	4
Glass → Iris	0.2697	**1.5800**	4
Hepatitis → Heart	0.1479	**1.8449**	10
Heart → Hepatitis	1.7610	**1.7903**	3
Ecoli → Heart	0.0167	**0.1671**	5
Heart → Ecoli	0.0384	**0.3815**	1
Australian → Breast Wisconsin	0.6115	**0.9095**	1
Breast Wisconsin → Australian	**0.0943**	0.0773	9

In order to validate our approach, we considered also the values of Calinski-Harabasz index in Table 4.

TABLE IV
CALINSKI-HARABASZ INDEX FOR TRANSFER BETWEEN DIFFERENT DOMAINS

Source Task → Target Task	No Transfer **C-NMF**	Transfer **RS-NMF**	k
Iris → Wine	0.6996	**13.9911**	1
Wine → Iris	0.3454	**1.7041**	9
Wine → Glass	4.7092	**5.2175**	2
Glass → Wine	2.8487	**9.8428**	6
Iris → Glass	5.1969	**5.9693**	4
Glass → Iris	0.1416	**2.2370**	6
Hepatitis → Heart	10.4467	**187.5136**	10
Heart → Hepatitis	123.7336	**128.9377**	3
Ecoli → Heart	15.3534	**123.7448**	5
Heart → Ecoli	11.9365	**178.9586**	1
Australian → Breast Wisc.	80.0094	**161.8653**	1
Breast Wisc. → Australian	**279.7122**	213.3286	9

We can see that our approach failed twice in terms of Dunn's index and only once in terms of Calinski-Harabasz index. In all the other cases transfer of learning outperforms "no transfer" C-NMF approach. It is important to add that C-NMF is equivalent to relaxed k-means clustering ([24]) and thus comparing BC-NMF with C-NMF means that we compare it to relaxed k-means at the same time.

To complete the analysis we plotted also the results of transfer obtained with different values of k for all datasets in Figures 3-14.

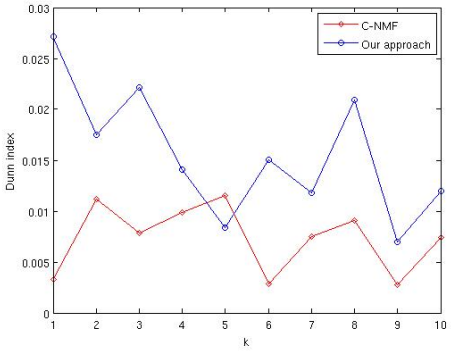

Fig. 3. Iris → Wine transfer

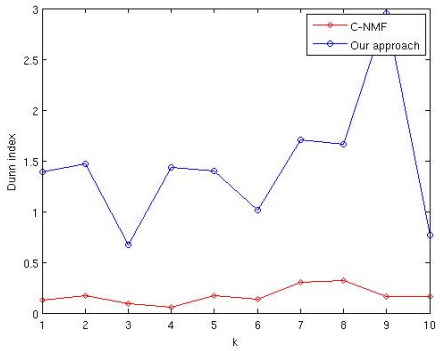

Fig. 4. Wine → Iris transfer

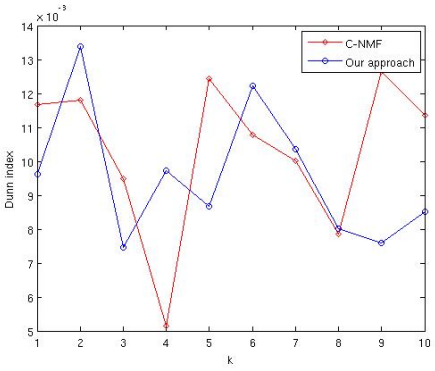

Fig. 5. Wine → Glass transfer

V. Conclusions

In this paper we proposed a new approach for unsupervised transfer learning. We perform multiple matrix decompositions of source data set's subspaces (RS-NMF) in order to generate a sequence of partition matrices and to further choose k most close of them in terms of correlation with initial target partition. We construct a sequence of prototype matrices using the previously selected partition matrices. Then, we use them in Multilayer NMF. In this way, we inject the knowledge from source to target task. This procedure is guided by the initial prototype matrix. Considering the results

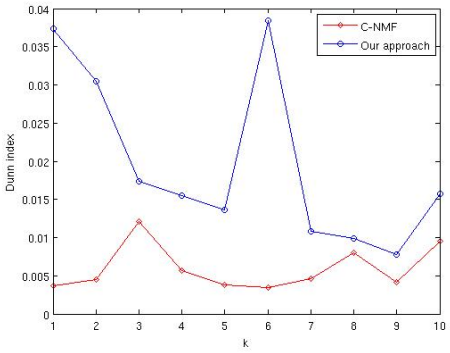

Fig. 6. Glass → Wine transfer

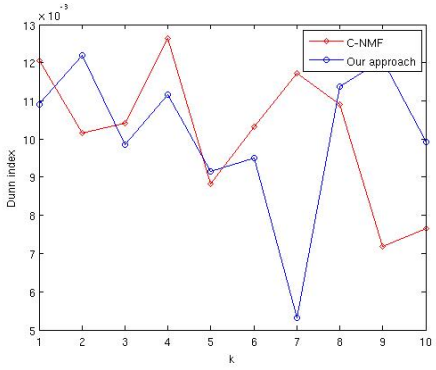

Fig. 7. Iris → Glass transfer

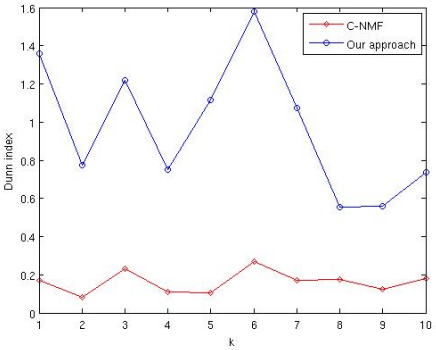

Fig. 8. Glass → Iris transfer

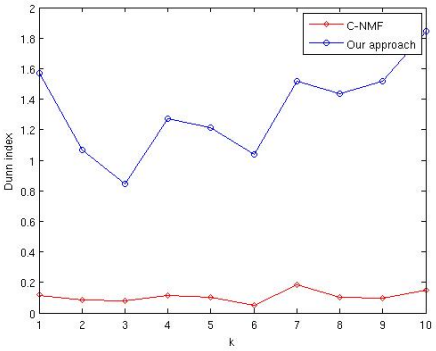

Fig. 9. Hepatitis → Hearts transfer

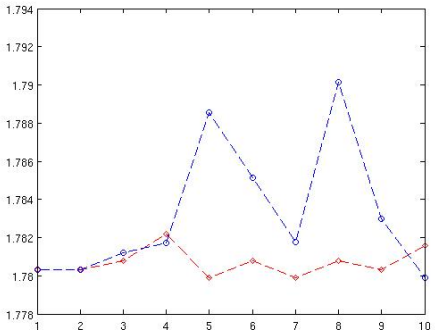

Fig. 10. Hearts → Hepatitis transfer

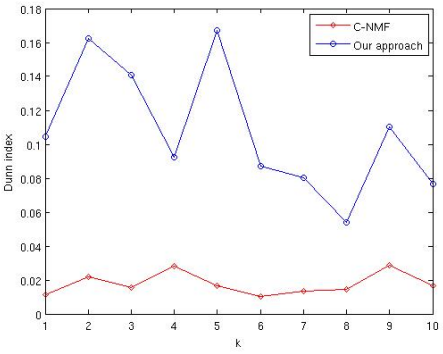

Fig. 11. Ecoli → Hearts transfer

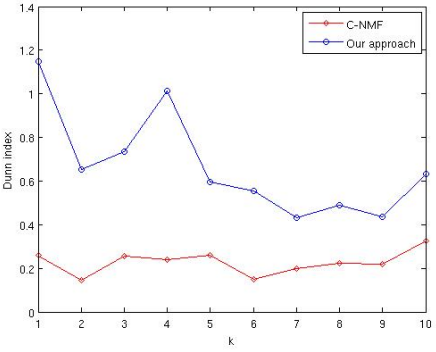

Fig. 12. Hearts → Ecoli transfer

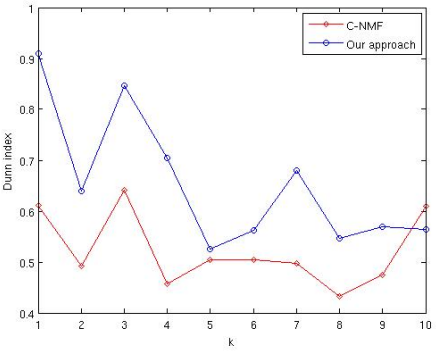

Fig. 13. Australian → Breast transfer

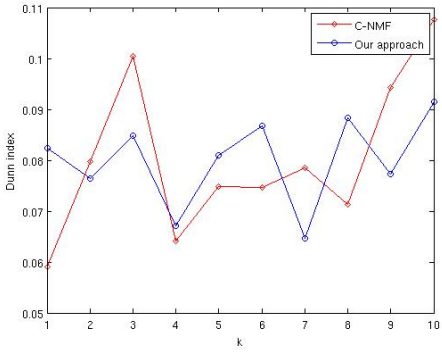

Fig. 14. Breast → Australian transfer

obtained during our experiments, we can conclude that our approach efficiently increases the quality of clustering of target data using the knowledge obtained from the source one.

In future, our work can be extended in multiple directions. First of all, our approach can be easily developed into a multi-task transfer learning algorithm - using data from multiple sources can further increase the gain from the transfer of knowledge. Secondly, it would be useful to study what is an optimal value of k - the number of partitions that will be used to construct the prototype matrices. We can see that for some data sets the choice of k can be crucial. The last thing that we want to say is that this approach is extremely simple and easy to implement.

REFERENCES

[1] Sinno Jialin Pan, Qiang Yang, "A Survey on Transfer Learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, 2010.

[2] Dai W., Yang Q., Xue G.R., Yu Y, "Self-taught clustering," *Proceedings of the 25th International Conference on Machine Learning*, New York, NY, USA, ACM, 2008.

[3] Zheng Wang, Yangqiu Song, Changshui Zhang, "Transferred Dimensionality Reduction," *Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases*, Part II, 2008.

[4] Breiman L., "Bagging Predictors," *Machine Learning*, vol. 24, pp. 123-140, 1996.

[5] Ho, Tin Kam., "The Random Subspace Method for Constructing Decision Forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20:(8), pp.832-844, 1998.

[6] Breiman, L., "Random forest," *Machine Learning*, vol. 45, 2001.

[7] L. Parsons, E. Haque, and H. Liu., "Subspace clustering for high dimensional data: a review," *ACM SIGKDD Explorations Newsletter*, 2004.

[8] Akata, Z., Thurau, C., Bauckhage, C., "Non-negative Matrix Factorization in Multimodality Data for Segmentation and Label Prediction," *In 16th Computer Vision Winter Workshop*, pp. 1-8, 2011.

[9] Jialu L., Chi W., Jing G., Jiawei, H., "Multi-View Clustering via Joint Nonnegative Matrix Factorization," *Proc. of 2013 SIAM Data Mining Conf.*, Austin, TX, May 2013.

[10] D.D. Lee, H.S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788 - 791, 1999.

[11] C. H. Ding, T. Li, and M. I. Jordan "Convex and SemiNonnegative Matrix Factorizations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, , vol. 99, no. 1, 2006.

[12] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," *Proc. of SIAM Int. Conf. on Data Mining*, pp. 606610, 2005

[13] Cichocki, A. and Zdunek, R., "Multilayer nonnegative matrix factorization," *Electronics Letters*, vol. 42, pp. 947-948, 2006.
[14] Y. Zhao and G. Karypis, "Empirical and theoretical comparisons of selected criterion functions for document clustering," *E. Machine Learning*, 55(3):311331, 2004.
[15] Erendira Rendon, Itzel Abundez, Alejandra Arizmendi and Elvia M. Quiroz, "Internal versus external cluster validation indexes," *International Journal of Computers and Communications*, vol. 5, no. 1, 2011.
[16] Gates, G.W., "The Reduced Nearest Neighbor Rule," *IEEE Transactions on Information Theory*, 431-433, May 1972.
[17] S. Aeberhard, D. Coomans and O. de Vel, "The classification performance of RDA," *Tech. Rep. no. 92-01*, Dept. of Computer Science and Dept. of Mathematics and Statistics, James Cook University of North Queensland, 1992.
[18] Ian W. Evett and Ernest J. Spiehler, "Rule Induction in Forensic Science," *Central Research Establishment*, Home Office Forensic Science Service. Aldermaston, Reading, Berkshire RG7 4PN.
[19] Paul Horton and Kenta Nakai, "A Probablistic Classification System for Predicting the Cellular Localization Sites of Proteins," *Intelligent Systems in Molecular Biology*, 109-115. St. Louis, USA 1996.
[20] Diaconis,P. and Efron,B., *Computer-Intensive Methods in Statistics*, Scientific American, Volume 248, 1983.
[21] Ross Quinlan, "Simplifying decision trees," *Int J Man-Machine Studies 27*, pp. 221-234, 1987.
[22] Wolberg, W.H., and Mangasarian, O.L., "Multisurface method of pattern separation for medical diagnosis applied to breast cytology," *In Proceedings of the National Academy of Sciences*, pp. 9193–9196, 1990.
[23] Gennari, J.H., Langley, P., and Fisher, D., "Models of incremental concept formation," *Artificial Intelligence*, vol. 40, pp. 11-61, 1989.
[24] T. Li and C. Ding., "Relationships Among Various Non-negative Matrix Factorization Methods for Clustering," *In Proc. of ICDM*, pp.362-371, 2006.