A Weakly-Supervised Factorization Method with Dynamic Graph Embedding

Seyed Amjad Seyedi
Department of Computer Engineering
University of Kurdistan
Sanandaj, Iran
amjadseyedi@eng.uok.ac.ir

Parham Moradi
Department of Computer Engineering
University of Kurdistan
Sanandaj, Iran
p.moradi@uok.ac.ir

Fardin Akhlaghian Tab
Department of Computer Engineering
University of Kurdistan
Sanandaj, Iran
f.akhlaghian@uok.ac.ir

Abstract—**Nonnegative matrix factorization (NMF) is an effective method to learn a vigorous representation of nonnegative data and has been successfully applied in different machine learning tasks. Using NMF in semi-supervised classification problems, its factors are the label matrix and the membership values of data points. In this paper, a dynamic weakly supervised factorization is proposed to learn a classifier using NMF framework and partially supervised data. Also, a label propagation mechanism is used to initialize the label matrix factor of NMF. Besides a graph based method is used to dynamically update the partially labeled data in each iteration. This mechanism leads to enriching the supervised information in each iteration and consequently improves the classification performance. Several experiments were performed to evaluate the performance of the proposed method and the results show its superiority compared to a state-of-the-art method.**

Keywords- Semi-supervised learning; Semi nonnegative matrix factorization; Graph Regularization; Label propagation

I. INTRODUCTION

Semi-supervised learning (SSL) aims to learn through using a small quantity of available labeled data and a large amount of unlabeled data. SSL can use unlabeled data along with the labeled data during the training process, thus it is an effective approach to improve the performance of the learning algorithms[1].

Due to the expensive and boring human labeling process, in recent years, semi-supervised learning has attracted much attention in the literature [2, 3]. A comprehensive review of semi-supervised learning methods can be found in [4, 5]. SSL methods can be viewed as two main tasks: using constraints and using partially labeled data. The first task introduces semi-supervision as pairwise information that indicates whether two instances clustered together or not, thus it can be employed when it is not possible to access labeled elements. On the other hand, the second task is supervised classification and aims to use both labeled and unlabeled instances to learn a classification model. However, SSL methods require a considerably smaller number of labeled instances compared to supervised ones. Nonnegative matrix factorization (NMF) has been shown as an effective method in this category that uses both labeled and unlabeled data to find a proper classification model.

In general, NMF has been proven as a multivariate analysis method to learn a vigorous representation of nonnegative data such as images and documents [6]. Up to now, many NMF methods have been successfully applied in pattern recognition, data clustering [7], signal processing [8, 9] and recommender systems[10-12]. To learn an effective classifier, it is applicable to incorporate the information on class labels into NMF while only a portion of labeled data is available. To this end, several types of research have been conducted to employ NMF in semi-supervised learning tasks. For example, the authors of [13], formulated NMF for semi-supervised learning as a joint factorization of the data matrix and the label matrix that shares a common factor matrix for consistency. Their method uses weighted residuals for the decomposition of the data matrix to handle missing data. Also, they use the label matrix to incorporate partially labeled data. On the other hand, NMF has been successfully applied to semi-supervised clustering. For instance, in [14] the authors proposed a method called *CPSNMF* that utilizes limited supervised information with pairwise constraints for unsupervised learning. This method, before starting the factorization process, uses an extra process that propagate the constraints to unconstrained samples .Also, in [15, 16] the authors introduced the *Semi-NMF* to extend NMF methods in cases that the data is not strictly non-negative. The *Semi-NMF* model imposes non-negativity constraints not only on first factor but allows mixed signs in both data and second factor. In this case, first factor indicates cluster centers and the second factor shows soft membership indicators for each instance. Besides, the authors of [17] extend semi-nonnegative matrix factorization in an incorporating deep structure. Recently, they also proposed weakly supervised factorization (in short *WSF*) [18], that incorporates available limited supervised information as class labels from the known attributes of a dataset. This method can be utilized for datasets that partially labeled. It can also be used in the case of a combination of different data sources that each one provides different attribute information.

In this paper, we propose a method called dynamic *WSF* (in short *DWSF*) that extends *WSF* in such a way where the class labels are iteratively updated and incorporated in the learning process. To this end, in each iteration, a portion of data takes their uncertain class labels and then they are further used as supervised information that leads to improving the convergence

978-1-5386-2585-9/17/$31.00 ©2017 IEEE

speed. *DWSF* decomposes the data matrix into two matrix factors. The first factor is label matrix and the other is representative matrix. Also, in order to initialize the label matrix factor, *DWSF* utilizes a *label propagation* algorithm in its process to propagate class labels from labeled data instances to unlabeled ones. In order to evaluate the performance of the proposed method, several experiments were performed on four real-world datasets. The obtained results show that the proposed method and its variants outperformed the *WSF* in term of classifier performance and execution time.

II. SEMI-NONNEGATIVE MATRIX FACTORIZATION

Traditional NMF methods suppose that every element in the data matrix should be non-negative. While in *Semi-NMF*, the data matrix is unconstrained, one of the factors is still restricted to be non-negative and the other has no restriction[19]. Suppose that in *Semi-NMF* the data matrix X should be approximated by two factors C and M. In S*emi-NMF* the non-negativity constraint is relaxed of the NMF and X and C are allowed to have mixed signs, while it restricts only M so that it comprises of strictly non-negative components. On the other hand, the data matrix is approximated by the following factors:

$$X^{\pm} \approx C^{\pm} M^{+} \tag{1}$$

where M should be non-negative, X and C have no constraints. From a clustering perspective, the factor C can be viewed as cluster centers and then M can be viewed as cluster memberships for each data point. In fact, M is non-negative and orthogonal, thus every column vector of M would have only one positive element that makes *Semi-NMF* similar to *k-means*. If *Semi-NMF*, does not impose an orthogonality constraint on its membership matrix, it can be viewed as a soft clustering method (such as Fuzzy c-means) as follows:

$$S_{semi-nmf} = \sum_{i}^{n}\sum_{j}^{k} m_{ij} \left\| x_i - c_j \right\|^2 = \left\| X - CM \right\|_F^2 \tag{2}$$

This objective function can be solved using descend gradient method and thus it is computed via an iterative updating process that alternatively updates C and M using the following equations [16]:

$$C = XM^{\dagger} = XM^{T}(MM^{T})^{-1} \tag{3}$$

Note that MM^T is a $k \times k$ positive semidefinite matrix. The inversion of this small matrix is trivial. In most cases, M^TM is nonsingular. When M^TM is singular, we take the pseudo-inverse.

$$M = M \odot \sqrt{\frac{\left[C^T X\right]^{+} + \left[C^T C\right]^{-} M}{\left[C^T X\right]^{-} + \left[C^T C\right]^{+} M}}$$
$$A^{+} = \frac{|A| + A}{2},\ A^{-} = \frac{|A| - A}{2} \tag{4}$$

where we separate the positive and negative parts of a matrix A.

III. PROPOSED METHOD

In this paper, a novel weakly-supervised method is proposed that incorporates class labels as a limited supervision in its process. This method, inspiring from [20, 21], a novel graph regularization technique with a *label propagation* mechanism is used to enrich class labels. In the proposed method which is called *DWSF*, to construct the static graph W from data, each node shows a data point in our initial dataset. Also, a node i is connected to another node j iff we have a priori knowledge that those samples share the same label, and this edge has a weight w_{ij} that is obtained as follows:

$$w(i,j) = \begin{cases} exp\left(-\left\|x_i - x_j\right\|^2 / \mu\sigma^2\right) & y_i = y_j \\ 0 & otherwise \end{cases} \tag{5}$$

Then the Laplacian [22] of this graph is defined as $L = D - W$, where D is a diagonal matrix with $D_{ii} = \Sigma_k w_{ik}$, whose entries are column sums of W. In order to control the amount of embedded information in the graph a term R is introduced [20] that controls the smoothness of the low dimensional representation and defined as follows:

$$\begin{aligned} R &= \frac{1}{2}\sum_{i,j}\left\|m_i - m_j\right\|^2 w_{ij} \\ &= \sum_j m_j^T m_j D_{jj} - \sum_{i,j} m_i^T m_j W_{ij} \\ &= Tr(M^T DM) - Tr(M^T WM) \\ &= Tr(M^T LM) \end{aligned} \tag{6}$$

where m_i is a vector that shows a membership value of an instance i to all classes. By minimizing this term R, it ensures that if two instances have the same class labels, the difference between m_i and m_j should be minimized. By combining the term R introduced in (6), with the cost function of *Semi-NMF* (i.e. (2)) the following cost function is obtained:

$$S_{wsf} = \left\|X - MC\right\|_F^2 + \lambda Tr(M^T LM) \tag{7}$$

where C is a representative matrix and M show the label matrix. Using the method proposed in [16], the update rule of the above objective function is obtained as follows:

$$C = XM^{\dagger} = XM^{T}(MM^{T})^{-1} \tag{8}$$

$$M = M \odot \sqrt{\frac{\left[C^T X\right]^{+} + \left[C^T C\right]^{-} M + \lambda M^T W}{\left[C^T X\right]^{-} + \left[C^T C\right]^{+} M + \lambda M^T D + \varepsilon I}}$$
$$A^{+} = \frac{|A| + A}{2},\ A^{-} = \frac{|A| - A}{2} \tag{9}$$

Note that the label matrix M includes information about class labels, and the correlation of these labels can be viewed as the similarity between data points in the label space. Therefore, class labels can be converted into a similarity graph W. Then, using this graph, the Laplacian matrix is updated in each iteration. This means that in each iteration, the label matrix is enriched by assigning labels to those of unlabeled data. On the other hand, it can say that we have two types of labels including implicit and explicit labels. The explicit labels are those that are available in the learning process, while the implicit ones are those that are obtained. The correlation matrix is quantified as follows:

$$W_{t+1} = M_t^T * M_t \tag{10}$$

Note that in each iteration, the explicit labels may be updated simultaneously with implicit labels and thus it needs to reset them to their original values as follows:

$$M^{(l)} = M_0^{(l)} \tag{11}$$

The pseudocode of the proposed *DWSF* method is provided in Algorithm 1.

Algorithm 1. Pseudocode of ***DWSF***

Input: data matrix $\boldsymbol{X} \in \mathbb{R}^{d\times n}$, label Vector $\boldsymbol{L} \in \mathbb{N}^{1\times n}$
Output: membership matrix ***M***
Begin
  Construct the affinity graph ***W*** by label matrix ***Y***
  Initialize random membership matrix $\boldsymbol{M_0}$
Repeat
  1. Reset certain labels by (11)
  2. Update graph ***W*** by (10)
  3. Update class matrix ***C*** by (8)
  4. Update membership matrix ***M*** by (9)
  Until a Stopping criterion is reached
End

A. Initializing label matrix

Generally, in NMF, the factors are initialized randomly. It has been shown that the proper initialization of the factor leads to improve the performance of NMF methods [13]. One way is to use clustering algorithms such as k-means to initialize membership values in clustering methods. This kind of initialization cannot be effective for semi-supervised based NMF methods with available partial labeled data. In this paper, to improve the performance of DWSF, *label propagation(LP)* [3] that is a well-known method based on probability distribution is used to initialize the label matrix. This initialization mechanism also leads to improving the convergence speed of the proposed *DWSF* method. To this end, a fully connected graph W is constructed where each node represents a data point and the weight of each edge between two nodes x_i and x_j is obtained as follows:

$$w(i,j) = \exp\left(-\left\|x_i - x_j\right\|^2 / \mu\sigma^2\right) \tag{12}$$

where μ and σ are hyper-parameters. σ is learned by the mean distance to *k-nearest neighborhoods* [23]. A natural transition matrix can be defined by normalizing the weight matrix as follows:

$$p(i,j) = \frac{w(i,j)}{\sum_k w(i,k)} \tag{13}$$

Note that P is asymmetric and the sum of each row in P is equal to 1. Suppose a dataset as $X = \{X_l \cup X_u\}$, where X_l represents the labeled data and X_u represents the unlabeled data. For the multi-class problem, *1-of-C coding* representation is often used, so the label matrix is $Y_0 = [Y^{(l)}; Y^{(u)}] \in \mathbb{R}^{c\times n}$, where n is the number of data points, c is the number of classes, $Y^{(l)}$ is the label matrix for labeled data, and $Y^{(u)}$ is the label matrix for unlabeled data. We let $Y^{(l)}(i,k)$ be 1 if x_i is labeled as class k, otherwise, it is assigned by 0. In this process, labels are propagated by matrix product:

$$M_0 = Y * P \tag{14}$$

By using (12), those of unlabeled data in M_0, will be assigned by uncertain implicit class labels. Finally, membership matrix M_0 is normalized as follows:

$$m_0(i,j) = \frac{m_0(i,j)}{\sum_c m_0(i,c)} \tag{15}$$

The pseudocode of the proposed *DWSF* method with a *label propagation* mechanism to initialize label matrix factor (M) is presented in Algorithm 2.

Algorithm 2. Pseudocode of ***DWSF+LP***

Input: data matrix $\boldsymbol{X} \in \mathbb{R}^{d\times n}$, label Vector $\boldsymbol{L} \in \mathbb{N}^{1\times n}$
Output: membership matrix ***M***
Begin
  Construct the transition matrix ***P*** by (13)
  Construct the label matrix ***Y*** by *1-of-c Coding*
  Construct the affinity graph ***W*** by label matrix ***Y***
  Initialize membership matrix $\boldsymbol{M_0}$
    Propagate the labels by (14)
  Repeat
    5. Reset certain labels by (11)
    6. Update graph ***W*** by (10)
    7. Update class matrix ***C*** by (8)
    8. Update membership matrix ***M*** by (9)
  Until a Stopping criterion is reached
End

IV. EXPERIMENTAL RESULTS

This section addresses the evaluation of the proposed method for the SSL classification. The experiments involved the comparison among semi-supervised methods on several datasets with a variety of feature space dimensions. To evaluate the proposed method, efficacies of the semi-supervised initialization

and graph update processes individually will be analyzed. Then, the effect of both processes simultaneously will be examined. Finally, the runtime will be discussed for each method. During the implementation process, we adjust the parameters μ and λ of all algorithms to 2 and 0.1.

A. Datasets

We conduct experiments on four real-world datasets from the UCI data repository [24], which contains real application data collected in various fields and is widely used to test the performance of different machine learning algorithms. Table 1 presents the selected datasets with their number of samples, attributes, and classes.

- ***Heart* dataset** is the absence or presence of heart disease. The dataset is composed of studies derived from 270 patients, including cases considered and 13 features such as age, sex and several vital signs of a heart disease.
- ***Ionosphere* dataset** was collected by a system in Goose Bay, Labrador that consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kW. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass directly through the ionosphere.
- ***Glass* dataset** classifies glass. The results of a chemical analysis of glass splinters (as percentages of eight real-valued constituent elements) and the refractive index are used to classify a sample as from either float-processed or non-float-processed building windows, vehicle windows, containers, tableware, or head lamps.
- ***Seeds* dataset** comprised kernels belonging to three different varieties of wheat: Kama, Rosa and Canadian, 70 elements each, randomly selected for the experiment. Visualization of the internal kernel structure was detected using a soft X-ray technique.

TABLE I. UCI DATASETS

dataset	Properties		
	Samples	***Attributes***	***Classes***
Heart	270	13	2
Ionosphere	351	35	2
Seeds	210	7	3
Glass	214	9	6

B. Measurement

The purpose of semi-supervised classification is to predict labels of unlabeled samples. In the evaluation process, a well-known metric: *F-measure* [25] is used to measure the closeness of predicted labels to the actual ones. The F-measure can be viewed as a compromise between *recall* and *precision*. It is high only when both recall and precision are high. The F-measure assumes values in the interval [0,1]. It is 0 when no relevant samples have been retrieved, and is 1 if all retrieved samples are relevant and all relevant samples have been retrieved.

$$F-measure = 2\frac{Precision \times Recall}{Precision + Recall} \tag{16}$$

Precision or confidence (as it is called in data mining) denotes the proportion of predicted positive cases that are correctly real positives. This is what; machine learning, data mining and information retrieval focus on.

$$Percision = \frac{T_p}{T_p + F_p} \tag{17}$$

Recall or sensitivity is the proportion of real positive cases that are correctly Predicted Positive. This measures the coverage of the real positive cases by the predicted positive rule. Its desirable feature is that it reflects how many of the relevant cases the predicted positive rule picks up.

$$Recall = \frac{T_p}{T_p + F_n} \tag{18}$$

C. Results

To define a semi-supervised classification problem, we randomly sample different percentages of labeled data with label rate ranging from 2% to 10%. For each fixed label rate, we perform 10 independent experiments with random sampling. In figure 1, the average F - measure is reported. Both *LP* initialization and graph updating lead to better performance in all datasets. *LP* initialization because of using available information of labeled data instead of using random values is not a blind beginning and it can leverage the optimization. Also, graph updating simultaneously uses the implicit and explicit labels for a more accurate regularization process. As can be seen in figure 1, *DWSF+LP* is more efficient for multi-class datasets (i.e. *Seeds* and *Glass*).

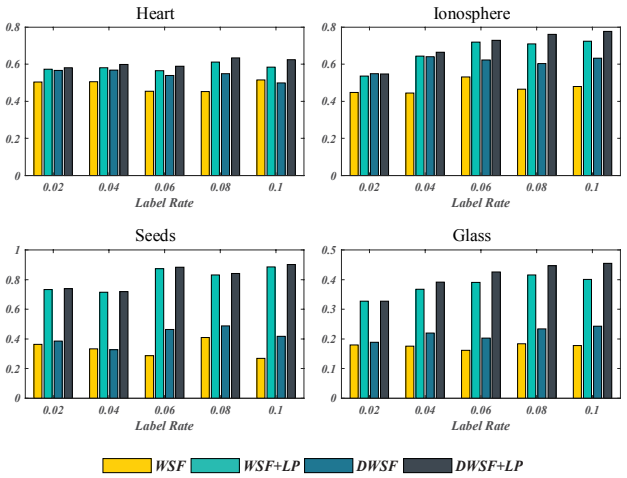

Figure 1. F-measure of the WSF and Proposed algorithms on four datasets.

D. Time

Figure 2 shows the average execute time of the *WSF* and our proposed algorithms on four datasets. This reported times for each method include three parts: initializing matrix M, steps of

optimization (updating two factors) and graph updating. In the most cases, despite higher time complexity of *LP* initialization than time complexity of random initialization, it leads to faster convergence (e.g. in *Glass* and *Seeds* datasets) because a reasonable starter matrix leads to a proper point for start the optimization process. Also, because graph updating has an extra calculation, time spent for *DWSF* algorithm will be increased (except *Glass* Dataset). However, a combination of both processes leads to lower total time spent in most datasets (e.g. in *Heart*, *Glass* and *Seeds*). The algorithms were implemented in *MATLAB R2017a* and executed on an *Intel Core i5* processor, with *3.2GHz CPU* and *8GB* of *RAM*.

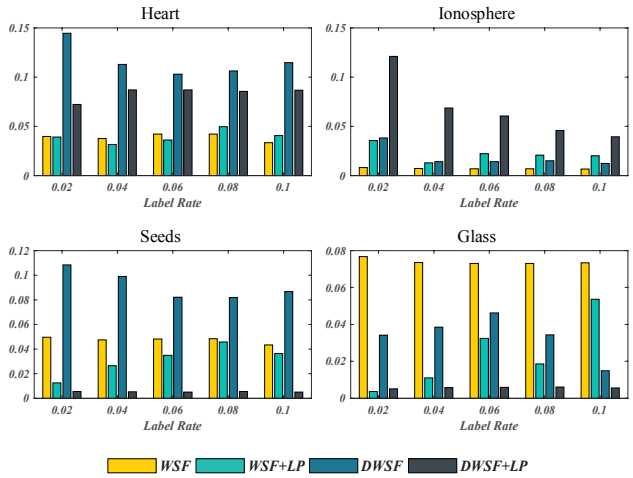

Figure 2. Average execute time comparison of the WSF and Proposed algorithms on four datasets. (time/s).

I. CONCLUSION

In this paper, we have proposed a novel semi-supervised learning method called dynamic weakly supervised factorization (DWSF), which improves the discriminative power in multi-class problems. Our method uses partially labeled data as a limited supervision in the framework of the non-negative matrix factorization concept. The proposed method also employs label propagation mechanism to initialize the label matrix factor. Also, in the proposed method, the label matrix is updated in each iteration. This mechanism leads to expedite the convergence speed as well as the performance of the learning model. The proposed method has been evaluated and compared to a state-of-the-art semi-supervised method and the obtained results show the superiority of the proposed method.

REFERENCES

[1] O. Chapelle, B. Scholkopf, and A. Zien, *Semi-Supervised Learning*: MIT Press, 2006.

[2] X. Zhu, "Semi-supervised learning," *Encyclopedia of Machine Learning*, pp. 892-897: Springer, 2011.

[3] X. Zhu, "Semi-supervised learning with graphs," Doctoral thesis, Department of Computer Science, Carnegie Mellon University, 2005.

[4] I. Diaz-Valenzuela, M. A. Vila, and M. J. Martin-Bautista, "On the Use of Fuzzy Constraints in Semisupervised Clustering," *IEEE Transactions on Fuzzy Systems,* vol. 24, no. 4, pp. 992-999, 2016.

[5] B. Wang, Z. Tu, and J. K. Tsotsos, "Dynamic label propagation for semi-supervised multi-class multi-label classification," in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 425-432.

[6] D. D. Lee, and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature,* vol. 401, no. 6755, pp. 788-791, 1999.

[7] W. Xu, X. Liu, and Y. Gong, "Document clustering based on non-negative matrix factorization," in Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, 2003, pp. 267-273.

[8] A. Cichocki, H. Lee, Y.-D. Kim, and S. Choi, "Non-negative matrix factorization with α-divergence," *Pattern Recognition Letters,* vol. 29, no. 9, pp. 1433-1440, 2008.

[9] A. Cichocki, Y. Washizawa, T. Rutkowski, H. Bakardjian, A.-H. Phan, S. Choi, H. Lee, Q. Zhao, L. Zhang, and Y. Li, "Noninvasive BCIs: Multiway signal-processing array decompositions," *Computer,* vol. 41, no. 10, 2008.

[10] D. Z. Navgaran, P. Moradi, and F. Akhlaghian, "Evolutionary based matrix factorization method for collaborative filtering systems," in 2013 21st Iranian Conference on Electrical Engineering (ICEE), 2013, pp. 1-5.

[11] M. Ranjbar, P. Moradi, M. Azami, and M. Jalili, "An imputation-based matrix factorization method for improving accuracy of collaborative filtering systems," *Engineering Applications of Artificial Intelligence,* vol. 46, no. Part A, pp. 58-66, 2015.

[12] Z. Shajarian, S. A. Seyedi, and P. Moradi, "A clustering-based matrix factorization method to improve the accuracy of recommendation systems," in 2017 Iranian Conference on Electrical Engineering (ICEE), 2017, pp. 2241-2246.

[13] H. Lee, J. Yoo, and S. Choi, "Semi-supervised nonnegative matrix factorization," *IEEE Signal Processing Letters,* vol. 17, no. 1, pp. 4-7, 2010.

[14] D. Wang, X. Gao, and X. Wang, "Semi-supervised nonnegative matrix factorization via constraint propagation," *IEEE transactions on cybernetics,* vol. 46, no. 1, pp. 233-244, 2016.

[15] C. Ding, T. Li, and W. Peng, "Nonnegative matrix factorization and probabilistic latent semantic indexing: Equivalence chi-square statistic, and a hybrid method," in American Association for Artificial Intelligence, 2006, pp. 137-143.

[16] C. H. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE transactions on pattern analysis and machine intelligence,* vol. 32, no. 1, pp. 45-55, 2010.

[17] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A Deep Semi-NMF Model for Learning Hidden Representations," in Proceedings of the 31th International conference on Machine learning, 2014, pp. 1692-1700.

[18] G. Trigeorgis, K. Bousmalis, S. Zafeiriou, and B. W. Schuller, "A deep matrix factorization method for learning attribute representations," *IEEE transactions on pattern analysis and machine intelligence,* vol. 39, no. 3, pp. 417-429, 2017.

[19] Y.-X. Wang, and Y.-J. Zhang, "Nonnegative matrix factorization: A comprehensive review," *IEEE Transactions on Knowledge and Data Engineering,* vol. 25, no. 6, pp. 1336-1353, 2013.

[20] M. Belkin, and P. Niyogi, "Using manifold stucture for partially labeled classification," in Advances in neural information processing systems, 2003, pp. 953-960.

[21] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *Journal of machine learning research,* vol. 7, no. Nov, pp. 2399-2434, 2006.

[22] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 33, no. 8, pp. 1548-1560, 2011.

[23] X. Yang, X. Bai, L. Latecki, and Z. Tu, "Improving shape retrieval by learning graph transduction," in 10th European Conference on Computer Vision, 2008, pp. 788-801.

[24] K. Bache, and M. Lichman, "UCI machine learning repository," I. University of California, School of Information and Computer Sciences, ed., 2013.

[25] D. M. W. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” *International Journal of Machine Learning Technology,* vol. 2, no. 1, pp. 37-63, 2011.