Pattern Recognition 52 (2016) 75–84

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Dynamic label propagation for semi-supervised multi-class multi-label classification

Bo Wang [a,*], John Tsotsos [b]

[a] Department of Computer Science, Stanford University, USA
[b] Department of Electrical Engineering and Computer Science, York University, Canada

ARTICLE INFO

Article history:
Received 6 January 2015
Received in revised form
1 September 2015
Accepted 8 October 2015
Available online 19 October 2015

Keywords:
Dynamic label propagation
Multi-modality
Semi-supervised

ABSTRACT

Existing semi-supervised methods often have difficulty in dealing with multi-class/multi-label problems due to insufficient consideration of label correlation, and lack an unified framework for multi-modality data. Also, the classification rate is highly dependent on the size of the available labeled data, as well as the accuracy of the similarity measures. To overcome these disadvantages, we propose a semi-supervised multi-class/multi-label classification scheme, dynamic label propagation (DLP), which performs transductive learning through propagation in a dynamic process. Our algorithm emphasizes dynamic metric fusion with label information. A multi-modality extension of the proposed method has been demonstrated to be capable to deal with multiple data types. Significant improvement over the state-of-the-art methods is observed on benchmark datasets for both multi-class and multi-label tasks. The proposed method is proved to be particularly advantageous with very few labeled data.

1. Introduction

In classification, it is often hard to obtain a single fixed distance metric for points in the entire data space. Moreover, nice properties enjoyed by graph-based (built on the distance metric) two-class semi-supervised classification [58] become less obvious in the multi-class classification situations [19], due to the correlations of the multiple labels.

Supervised metric learning methods often learn a Mahalanobis distance by encouraging small distances among points of the same label while maintaining large distances for points of different labels [51,49,50]. Graph-based semi-supervised learning frameworks on the other hand utilize a limited amount of labeled data to explore information on a large volume of unlabeled data. Label propagation (LP) [58] specifically assumes that nodes connected by edges of large similarity tend to have the same label through information propagated within the graph. A wide range of applications such as classification [33], clustering [18,7], dimension reduction [40] and retrieval [31,11,32] have adopted the label propagation strategy. Another type of semi-supervised learning, co-training [9], utilizes multi-modality features to help each other by pulling out unlabeled data to re-train and enhance the classifiers.

The above methods are mainly designed to deal with the binary classification problem. For the multi-class/multi-label case, the label propagation algorithm [58] becomes more problematic, and therefore some special care needs to be taken. A common approach to address the multi-class/multi-label learning is to use a one vs. all strategy. The disadvantage of one vs. all approaches is, however, that the correlations among different classes are not fully utilized. As discussed in [57], taking the class correlations into account often leads to a significant performance improvement.

Recent years have witnessed an emergence of applying deep learning together with semi-supervised learning [3,28,22]. Different from graph-based semi-supervised learning methods, these deep models make use of latent representations learned from large amount of unlabeled data, and also try to learn an embedding from labeled data. These models usually need a great number of data together with careful manipulation of model configuration to work well.

An additional often underappreciated issue of existing semi-supervised methods is that they lack enough power to learn a good manifold that can propagate labels when very few labeled data are given. This is due to the weak interactions between labeled and unlabeled data. However, local structures can be leveraged to improve this situation. In this paper, we propose a new method, dynamic label propagation (DLP), to simultaneously deal with the multi-class and multi-label problems. Our method incorporates the label correlations and instance similarities into a new way of performing label propagation. Our intuition in DLP is to update the similarity measures dynamically by fusing multi-

* Corresponding author.
E-mail addresses: bowang87@stanford.edu (B. Wang), tsotsos@cse.yorku.ca (J. Tsotsos).

http://dx.doi.org/10.1016/j.patcog.2015.10.006
0031-3203/

label/multi-class information, which can be understood in a probabilistic framework. The K nearest neighbor (KNN) matrix is used to preserve the intrinsic structure of the input data. We present comprehensive experimental results illustrating the advantages of the proposed method on multi-class categorization, object recognition, and multi-label classification.

To summarize, the contributions of the proposed DLP are three-fold: (1) it extends the original LP to be able to deal with multi-class/multi-label problems with high efficiency; (2) it improves the ability of semi-supervised learning when the number of labeled data is small. (3) it can combine multiple data modalities into the current semi-supervised framework and significantly improve the accuracy over simple data fusion.

2. Related work

As discussed in Section 1, a popular strategy toward multi-class/multi-label learning is to divide it into a set of binary classification problems, using techniques such as one-versus-the-rest, one-versus-one, and error-correcting coding [1]. These methods however have certain limitations including: (1) the difficulty to scale up to large data sets, and (2) inability to exploit the coherence and relations among classes due to the use of independent classifiers. Also, they may result in unbalanced classification outputs, especially when the number of classes is large.

A lot of recent attention has been focused on addressing these limitations of semi-supervised multi-class learning. The existing algorithms can be roughly classified into three categories: density-based, boosting-based and graph-based. First, a notable advance in density-based method is a multi-class extension to the TSVM by [52]; however, its high computational cost limits it from being widely adopted. Second, there are a variety of semi-supervised multi-class extensions to the boosting-based methods [45,37]; these methods differ in the loss functions and regularization techniques; the disadvantage of them is the lack of ability to utilize the correlation between labels and input features (especially for the unlabeled data), which, to some extent, jeopardizing the classification accuracy. Third, some recent advances in graph-based methods adopt Gaussian Processes [41,34] or Markov Random Walks [2]. Transduction by Laplacian graph [8,17] is also shown to be able to solve multi-class semi-supervised problems; although these algorithms make use of the relationship between unlabeled and labeled data, their computational complexity is demanding, e.g. of $\mathcal{O}(n^3)$.

However, there are far fewer attempts to tackle the semi-supervised multi-label problem, despite there being a rich body of literature about supervised multi-label learning. One popular method is label ranking [13], which learns a ranking function of category labels from the labeled instances and classifies each unlabeled instance by thresholding the scores of the learned ranking functions. Although being easy to scale up, label ranking fails to exploit the correlations among data categories. Recently, category correlations are given more attention in multi-label learning. A maximum entropy method is employed to model the correlations among categories in [57]. [35] studies a hierarchical structure to handle the correlation information. In [21], a correlated label propagation framework is developed for multi-label learning that explicitly fuses the information of different classes. However, these methods are only for supervised learning, and how to make use of label correlation among unlabeled instances is still unclear. [30] uses constrained non-negative matrix factorization to propagate the label information by enforcing the examples with similar input patterns to share similar sets of class labels. Another semi-supervised multi-label learning technique [12] develops a regularization with two energy terms about smoothness of input instances and label information by solving a Sylvester Equation. A similar algorithm [54] solves the multi-label problem with an optimization framework with regularization of the Laplacian matrix.

Different from these semi-supervised multi-label methods, the proposed method explicitly merges the input data and label correlations. Moreover, by doing projection on the fused manifolds, DLP further takes advantage of the correlations among labeling information of unlabeled data. Our work also differs significantly from a very recent algorithm [23], which emphasizes the learning of fusion parameters for unlabeled data; the focus here is however the dynamic update of the similarity functions from both data and label information. In addition, our method is a unifying framework for both multi-class and multi-label classification.

The current literature addressing combined multi-class and multi-label problem is still limited. The reason is two-fold. First, the multi-label problem considers the label correlations, but it may lead to a loss in the discrimination power of the multi-class classifiers. On the other hand, the prediction function learned in the multi-class problem often fails to solve the multiple overlaps of different labels in the multi-label problem. The proposed dynamic label propagation method (DLP) aims to solve semi-supervised multi-class and multi-label problem simultaneously by combining the discriminative graph similarities and the label correlations in a dynamic way, while preserving the intrinsic structure of input data. These two steps can well balance the different needs of the multi-class and multi-label problems.

3. Label propagation

First, a brief introduction of the well-known label propagation algorithm is provided in this section. We are given a finite weighted graph $G=(V,E,W)$, consisting of a set of vertices V based on a data set $X=\{x_i, i=1,\ldots,n\}$, a set of edges E of $V\times V$, and a nonnegative symmetric weight function $W: E\rightarrow[0,1]$. If $W(i,j)>0$, we say that there is an edge between x_i and x_j. We interpret the weight $W(i,j)$ as a similarity measure between the vertices x_i and x_j. If ρ is a distance metric defined on the graph, then the similarities matrix can be constructed as follows:

$$W(i,j)=h\left(\frac{\rho(x_i,x_j)^2}{\mu\sigma^2}\right), \tag{1}$$

for some function h with exponential decay at infinity. A common choice is $h(x)=\exp(-x)$. Note that μ and σ are hyper-parameters. σ is learned by the mean distance to K-nearest neighborhoods [53].

A natural transition matrix on V can be defined by normalizing the weight matrix as

$$P(i,j)=\frac{W(i,j)}{\sum_{k\in V}W(i,k)}, \tag{2}$$

so that $\sum_{j\in V}P(i,j)=1$. Note that P is asymmetric after the normalization.

Denote the dataset as $X=\{X_l \bigcup X_u\}$, where X_l represents the labeled data and X_u represents the unlabeled data. One important step in label propagation (LP) is clamping, i.e., the labels of labeled data must be reset after each iteration. For the two-class LP, we refer readers to [58]; for the multi-class problem, 1-of-C coding representation is often used, so the label matrix is $Y_{t=0}=[Y^{(l)};Y^{(u)}]\in\mathbb{R}^{n\times C}$, 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, and 0 otherwise. During each iteration, two steps are performed: (1) labels are propagated $Y_t=P*Y_{t-1}$; and,

1. Construct a probabilistic transition matrix P by Eqn.(2).
2. Let $Y_0 = [Y_0^l; \mathbf{0}]$.
3. Performing the following steps for T steps:
   3.a $Y_{t+1} = P * Y_t$,
   3.b $Y_{t+1}^{(l)} = Y_0^l$.
4. Output Y_T

Fig. 1. Algorithm of label propagation (LP).

(2) labels of labeled data X_l are reset. The main algorithm of label propagation is summarized in Fig. 1.

4. Dynamic label propagation

4.1. Local similarity

Given a dataset X and its corresponding graph $G = (V, E, W)$, we construct a KNN graph $\mathcal{G} = (V, \mathcal{E}, \mathcal{W})$: the vertices of $\mathcal{G}$ are the same as in G, and weighted edges are those nearby ones only. In other words, those similarities between non-neighboring points (in terms of the pairwise similarity values) are set to zero. Essentially we make the assumption that local similarities (high values) are more reliable than far-away ones and accordingly local similarities can be propagated to non-local points through a diffusion process on the graph. This is a mild assumption widely adopted by other manifold learning algorithms [43,36].

Using K nearest neighbor (KNN) to measure local affinity, we construct $\mathcal{G}$ with associated similarity matrix:

$$\mathcal{W}(i,j) = \begin{cases} W(i,j) & \text{if } x_j \in KNN(x_i) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Then the corresponding KNN matrix becomes

$$\mathcal{P}(i,j) = \frac{\mathcal{W}(i,j)}{\sum_{x_k \in KNN(x_i)} \mathcal{W}(i,k)}. \tag{4}$$

Note that P carries the full pair-wise similarity information among the data whereas $\mathcal{P}$ only encodes the similarity to nearby data points. However, $\mathcal{P}$ incorporates the robust structural information about the input data space. For clarity, we call P the status matrix and $\mathcal{P}$ the corresponding KNN matrix.

4.2. Label fusion on diffusion space

One disadvantage of label propagation is that it does not work well on multi-class/multi-label classification problem due to a lack of interplay among labels within different classes. In this paper, we propose a dynamic version of label propagation that aims to improve the effectiveness on multi-class/multi-label classification. Our main idea is to have an improved transition matrix by fusing information of both data features and data labels in each iteration (Fig. 2).

Given the kernel P_t, where t denotes the number of iterations, we can define the diffusion distance [25] at time t as

$$D_t(i,j) = \| P_t(i,:) - P_t(j,:) \|. \tag{5}$$

The diffusion process maps the data space into an n-dimensional space $\Re_t^n$ (called diffusion space as in [25]) in which each data point is represented by its transition probability to the other data points. Our dynamic label propagation process, akin to label propagation, is an iterative algorithm. At iteration t, $\mathbf{x}_t$ corresponding to the latent representation in the diffusion space for the data while $\mathbf{y}_t$ corresponding to the latent representation in the diffusion space for the label. It is reasonable to assume that for each data $\mathbf{x}_t \in \Re_t^n$, we have $p(\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_t | \boldsymbol{\mu}_t, P_t)$, where μ_t is unknown. Note that the label matrix Y_t contains information about

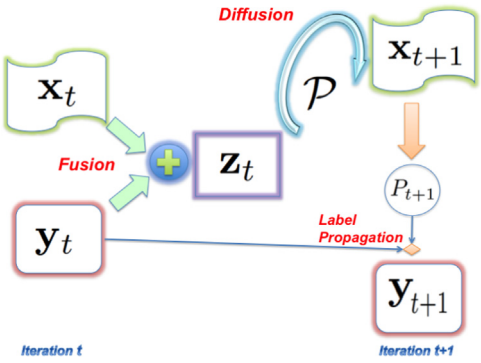

Fig. 2. An illustrative flow chart of dynamic label propagation (DLP).

class labels, and the correlation of these labels $K_Y = Y_t Y_t^T$ can be viewed as the similarity between data points in the label space $\Re_t^n$, and data points in this label space $\Re_t^n$ have the probability $p(\mathbf{y}_t) = \mathcal{N}(\mathbf{y}_t | 0, K_t)$ (Fig. 2).

We divide our method into two steps: Kernel Fusion and Kernel Diffusion, and we begin by describing kernel fusion. First, we want to combine both data and label spaces, hoping labels albeit partial label information can help boost the accuracy of label propagation. It is straightforward to combine $\mathbf{x}_t$ and $\mathbf{y}_t$ in the diffusion space:

$$\mathbf{z}_t = \mathbf{x}_t + \sqrt{\alpha}\mathbf{y}_t. \tag{6}$$

Here α is a hyper-parameter that represents the weights for the label information. And $\mathbf{z}_t$ denotes the newly fused latent representation in the diffusion space. It is easy to verify that $\mathbf{z}_t$ still follows with a Gaussian distribution:

$$p(\mathbf{z}_t) = \mathcal{N}(\mathbf{z}_t | \mu_t, P_t + \alpha Y_t Y_t^T). \tag{7}$$

This also sheds light on how to operate this fusion process explicitly, which is the fusion of the status matrix P_t and the label kernel $K_Y = Y_t Y_t^T$. Hence, the fused kernel is then

$$F_t = (P_t + \alpha Y_t Y_t^T). \tag{8}$$

This simple fusion technique considers the correlation among the instance label vectors. The underlying assumption is that two instances with high correlated label vectors tend to have high similarity in the input data space. The correlation between label vectors can represent the label dependency among instances, especially for the multi-label/multi-class problem. The advantage of the fusing transition kernel and the label correlation is two-fold. On one hand, two instances with high correlated label vectors are likely to have high similarity in input data space, this fusion process therefore enhances the fitness of the kernel matrix for the input manifold. On the other hand, the resulting kernel matrix leads to better label information through next round of label propagation. In this way, we build up a dynamic interaction process between the feature space and label space. However, since the label information is dynamically updated during the propagation process, the resulting label information after the initial several rounds no longer improves the transition matrix, sometimes even makes it worse. To deal with this problem, we design a novel fusion-operator based on the local neighbors as follows.

The second main step is kernel diffusion. Assume P_0 is the initial status matrix of the input data calculated using Eqs. (1) and (2), and $\mathcal{P} = KNN(P_0)$ by Eqs. (3) and (4); We employ this linear operator $\mathcal{P}$ to do the projection

$$\mathbf{x}_{t+1} = \mathcal{P}\mathbf{z}_t + \lambda_t \boldsymbol{\varepsilon}, \tag{9}$$

where $\boldsymbol{\varepsilon}$ is white noise, i.e. $p(\boldsymbol{\varepsilon}) = \mathcal{N}(\varepsilon | 0, 1)$. Note that $\mathcal{P}$ is a sparse version of P_0 and only local neighbor information in the space is

kept in the operator $\mathcal{P}$:

$$\mathbf{x}_{t+1}(i) = \sum_{j \in KNN(i)} P_0(i,j)\mathbf{z}_t(j) + \lambda_t \varepsilon$$
$$= \sum_{j \in KNN(i)} P_0(i,j)(\mathbf{x}_t(j) + \alpha \mathbf{y}_t(j)) + \lambda_t \varepsilon$$

With this linear operation, we have

$$p(\mathbf{x}_{t+1} | \mathbf{z}_t) = \mathcal{N}(\mathbf{x}_{t+1} | \mathcal{P}\mathbf{z}_t, \lambda_t I). \quad (10)$$

The marginal distribution of $\mathbf{x}_{t+1}$ is

$$p(\mathbf{x}_{t+1}) = \int \mathcal{N}(\mathbf{z}_t | \boldsymbol{\mu}_t, F_t)\mathcal{N}(\mathbf{x}_{t+1} | \mathcal{P}\mathbf{z}_t, \lambda_t I) d\mathbf{z}_t = \mathcal{N}(\mathbf{x}_{t+1} | \mathcal{P}\boldsymbol{\mu}_t, \mathcal{P}F_t(\mathcal{P})^T + \lambda_t I). \quad (11)$$

The above equation implies that the essence of dynamic label propagation is to do linear operations on diffusion space iteratively. Note that $\mathbf{x}_{t+1}$ is a point in the diffusion space. Instead of performing linear projection in the original data space, we do projection in the diffusion space. The advantages of projection onto the diffusion space are two-fold: (1) we avoid the need to perform computational expensive sampling procedures in the input space; (2) the resulting variance matrix again is a good diffusion kernel for label propagation.

The intuition behind this projection lies in the fact that simple fusion of label correlation in Eq. (8) would result in a degeneration at the first round when the learned label information of unlabeled data is not accurate enough to infer the similarities in the input space. Hence, inspired by [46,20], we need to re-emphasize the intrinsic structure between all the input data by the KNN matrix. From (13), we can see that the diffusion process propagates the similarities through the KNN matrix. In this way, we can adjust the fused kernel matrix to maintain part of the information of the initial structure.

The direct reflection of this projection on diffusion space is that, at each iteration, we construct the transition matrix for next iteration to be

$$P_{t+1} = \mathcal{P}(P_t + \alpha Y_t Y_t^T)\mathcal{P}^T + \lambda_t I. \quad (12)$$

Thus, we have

$$P_{t+1}(i,j) = \sum_{k \in KNN(i)} \sum_{l \in KNN(j)} P_0(i,k)P_0(j,l)(P_t(k,l) + \alpha \langle Y_t(k,:), Y_t(l,:) \rangle) + \lambda_t \delta_{ij}. \quad (13)$$

where $\langle \mathbf{x}_1, \mathbf{x}_2 \rangle$ denotes the inner product of two vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, and $\delta_{ij} = 1$ if $i=j$, 0 otherwise. From Eq. (13), we see that only information between dominant neighbors are propagated into the transition matrix of next iteration. An important observation is that if data i and j have common dominant neighbors in both similarity metrics, it is highly possible that they belong to the same class.

We summarize the details of dynamic label propagation in Fig. 3.

1. Construct a probabilistic transition matrix P_0 by Eqn.(2).
2. Let $Y_0 = [Y_0^l; \mathbf{0}]$.
3. Calculate the KNN matrix $\mathcal{P}$ of P_0,
4. Performing the following steps for a desired T steps:
   - 4.a $Y_{t+1} = P_t * Y_t$,
   - 4.b $Y_{t+1}^{(l)} = Y_0^l$,
   - 4.c $P_{t+1} = \mathcal{P}(P_t + \alpha Y_t Y_t^T)\mathcal{P}^T + \lambda_t I$.
5. Output Y_T.

Fig. 3. Algorithm of dynamic label propagation (DLP).

4.3. Analysis

4.3.1. Convergence analysis

It is difficult to give a formal theoretical proof of the convergence of DLP. However, empirical experience shows DLP converges much faster than LP (see Fig. 5). Usually, LP needs 1000–5000 iterations to converge, while DLP only needs 10–50 iterations. This is because the diffusion process projects the fused manifold into a KNN structure where only local similarities are preserved. The learned labels can improve the similarity between input instances quickly.

A loose theoretical proof of convergence can be constructed based on the spectral analysis of the diffusion projection $\mathcal{P}$. Since $\mathcal{P}$ is a KNN matrix of P_0, it is easy to see that the spectral radius of $\mathcal{P}$ is less than 1. We have

$$Y_t \propto Y^{(\infty)} + [(\mathcal{P})^t(P_0 + \alpha Y_0 Y_0^T)(\mathcal{P}^T)^t]P_0 Y_0 + o(t) \quad (14)$$

where $o(t)$ is an infinitesimal as t approaches infinity, and $Y^{(\infty)} \in \mathbb{R}^{n \times C}$ is a constant label matrix. We observe that since the spectral radius of $\mathcal{P}$ is less than 1, we have $\lim_{t \to \infty} \mathcal{P}^t \to \mathbf{0}$. Hence, the final label is $\lim_{t \to \infty} Y_t = Y^{(\infty)}$, although we do not have a closed form for $Y^{(\infty)}$ at present.

4.3.2. Time complexity

The traditional Label Propagation algorithm has a complexity of $\mathcal{O}(n^2)$, however, since our DLP only diffuses the similarities on KNN structures, DLP shares the same scale of time complexity. For the step of kernel fusion, we only perform the addition of two matrices, so the time cost is $\mathcal{O}(n^2)$. For the step of diffusion in Eq. (12), we decompose it as in Eq. (13), from which we observe that only local neighbors are used to propagate the similarities. An easy way to speed up the diffusion process is, first we keep a record of the *KNN* matrix and then every time we perform the diffusion process, we extract the fixed local structure from the KNN structure and only perform multiplication K times for each pair of data points. Therefore we can update the transition kernel in (12) in time $Kn^2 + Kn$. To summarize, the overall time complexity of DLP is $\mathcal{O}(Kn^2)$, where $K \ll n$.

4.3.3. Parameter analysis

There are several parameters to tune in DLP. How to choose the number of neighbors in the KNN matrix $\mathcal{P}$ remains an open problem. A small K leads to insufficient structural information in $\mathcal{P}$; a large K value results in an increase in the time complexity and loss in the sparsity in $\mathcal{P}$. There is a trade-off between accuracy and complexity. In all our experiments, we choose K from $\{10, 20, 30, 40, 50\}$ by 10-fold cross-validation. Another two important parameters in DLP are α and λ. α is the weight of label correlation, while λ represents the importance of regularization. Fortunately, DLP is not sensitive to these two parameters.

1. Construct m probabilistic transition matrix $P_0^{(i)}$ by Eqn.(2) and a consensus graph $P_0^{(c)} = \frac{1}{m}(\sum_{i=1}^m P_0^{(i)})$.
2. Let $Y_0 = [Y_0^l; \mathbf{0}]$.
3. Calculate the KNN matrix $\mathcal{P}^{(i)}$ of $P_0^{(i)}$,
4. Performing the following steps for a desired T steps:
   - 4.a $Y_{t+1} = P_t^{(c)} * Y_t$,
   - 4.b $Y_{t+1}^{(l)} = Y_0^l$,
   - 4.c $P_{t+1}^{(i)} = \mathcal{P}^{(i)}(P_t^{(c)} + \alpha Y_t Y_t^T)(\mathcal{P}^{(i)})^T + \lambda_t I$.
   - 4.d $P_{t+1}^{(c)} = \frac{1}{m}(\sum_{i=1}^m P_{t+1}^{(i)})$.
5. Output Y_T.

Fig. 4. Algorithm of multi-modality dynamic label propagation (MDLP).

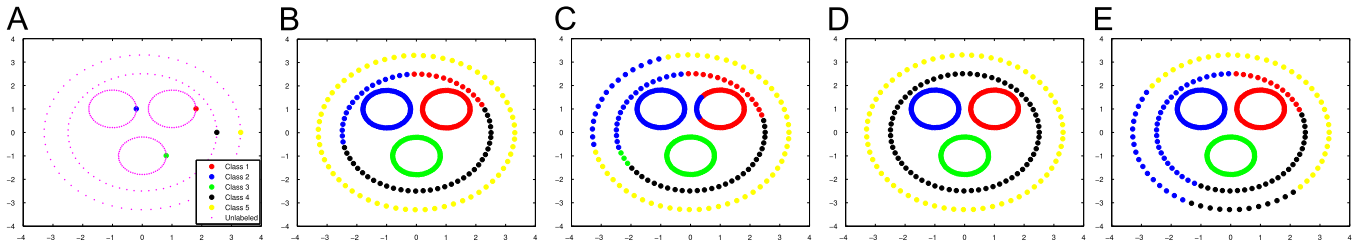

Fig. 5. (A) is the toy data with only one labeled data (the colored dots) for each class. (B) is the classification result without using label correlations. (C) is the classification result without using diffusion process. (D) is the result of DLP with only 20 iterations. (E) is the result of LP with 5000 iterations. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 1
A detailed description of the selected 6 benchmark multi-class datasets.

Name	#Class	#Features	#Data	Field
Wine	3	13	178	Pattern recognition
Iris	3	4	150	Agriculture
Dna	3	180	2000	Biology
Vowel	11	10	528	Pattern recognition
Vehicle	4	18	846	Pattern recognition
Segment	7	19	2310	Pattern recognition

4.4. Extension to multi-modality DLP

It is not uncommon to have multiple description of a set of data in many data science applications. For example, for a same image, we can have more than one feature to describe this image. Typical examples are SIFT, HOG and various variants of these descriptors. Different descriptors often give a partial representation of an image with different focus [46]. It remains to be an open problem to best combine multiple views of a same set of data.

Here we provide a natural extension of multi-modality DLP. The main idea is that we use the average of multiple graphs as a "consensus" graph that contains an agreement of multiple graphs. Given m different views $\{X^{(i)}, i=1,\ldots,m\}$, we can construct m transition graphs $\{P^{(i)}, i=1,\ldots,m\}$ and corresponding KNN graphs $\{\mathcal{P}^{(i)}, i=1,\ldots,m\}$ from these views. Similar with DLP, we aim to update the similarity graphs iteratively. On the t-th iteration, we construct a consensus graph $P_t^c = (1/m)\sum_{i=1}^{m} P^{(i)}$. We run label propagation on this consensus graph:

$$Y_{t+1} = P_t * Y_t, \quad Y_{t+1}^{(l)} = Y_0^l. \tag{15}$$

We also project this consensus graph to each individual KNN graph and update the similarities:

$$P_{t+1}^{(i)} = \mathcal{P}^{(i)}(P_t^{(c)} + \alpha Y_t Y_t^T)(\mathcal{P}^{(i)})^T + \lambda_t I \tag{16}$$

We summarize the details of multi-modality dynamic label propagation in Fig. 4.

4.5. A toy data

We first test our dynamic label propagation on a toy data set. It consists of five circles (i.e., 5 classes) (see Fig. 5(A)). This is a challenging dataset since it contains multiple classes and only one in each class is labeled. We test the effect of the two steps in the dynamic label propagation. We construct the KNN matrix as in [47]. We omit the first step that fuses the label correlation with the kernel matrix. The other steps are all the same. The result is shown in Fig. 5(B). Second, we do the first step to fuse label correlations but omit the second step of kernel diffusion. The result is shown in Fig. 5(C). Comparing these two results, we see that each step is important to the final result of DLP. Without the label correlation, DLP fails to capture the dependence between different classes; without the kernel diffusion process, the DLP goes wild because the label correlation in the beginning provides a poor guidance for the kernel matrix. In addition, we show the classification results of DLP and LP in Fig. 5(D) and (E). It is observed that our method only needs a few iterations to converge while LP gets a reasonable result only after thousands of iterations.

5. Experiments

5.1. Semi-supervised multi-class learning

We compare our DLP with several popular semi-supervised learning methods: (1) Label Propagation (LP); (2) a variant of LP on KNN structure (LP+KNN) [42]; (3) Local and Global Consistency (LGC) [56]; (5) Transductive SVM (TSVM) [39]; (6) LapRLS [5]. Note that for LP and LGC, we use one-vs-the-rest methods to deal with multi-class problems; for TSVM and LapRLS, they have their own multi-class extensions.

We use 6 multi-class datasets[1] to perform a thorough comparison. These datasets cover different research fields (e.g., signal processing, computational biology) and exhibit different properties, such as their different dimensions. Detailed statistics about these datasets can be found in Table 1. Results are shown in Fig. 6. We randomly sample different percentages of labeled data with label rate ranging from 1% to 10%. For each fixed label rate, we perform 10 independent experiments with random sampling. Average accuracy is reported. An noticeable observation is that, when the label rate is small, our DLP can achieve significant improvement over existing methods. One reason is that our DLP can exploit both local structure and potential label correlation to help enhance the graph structures that are essential to help graph-based semi-supervised method. To connect a strong interaction between labels and unlabeled data is very helpful for label propagation.

5.2. Semi-supervised multi-label classification

In this section, we test our method on the task of semi-supervised multi-label classification. Three standard datasets about multi-label learning are used: Corel dataset for automatic image annotation [14], Yeast data for gene functional analysis [15], and Scene dataset for natural scene classification [10]. Table 2 shows the summary of these three datasets.

We evaluate the performance using two common evaluation metrics: (1) *F1 Micro* which can be seen as the weighted average of F1 scores over all the categories; (2) average precision which evaluates the average fraction of labels ranked correctly aligned with the true labels. Apparently, the higher these two metrics are, the better the performance is. We refer the details of the evaluation metrics to [44]. We compare our methods with three popular multi-label methods: (1) Mlknn [55], which uses a naive but effective KNN classification

[1] http://www.csie.ntu.edu.tw/cjlin/libsvmtools/datasets/multiclass.html

method; (2) BoosTtexter [38] which applies a boosting style ranking algorithm with particular effectiveness on text categorization; (3) Tram [23], which adopts graph-based methods to propagate sets of labels. Tram is a similar method with the proposed method in the sense that we fall into the same streamline of propagating labels on graphs. However, the difference lies in the fact that DLP iteratively updates the graph while Tram uses static graphs.

We show the comparisons in Fig. 7. Again, we observe that DLP obtains the most improvement over the alternatives when the label rate is low. This suggests that label correlations can help boost up the performance even if only potential labels are used [21].

5.3. Multi-modality semi-supervised learning

In this section we benchmark our method on the task of multi-modality classification in which multiple features/similarities are

Table 2
A summary of the selected 3 benchmark multi-label datasets.

Data	#Instance	#Attributes	# Labels	Tasks
Corel	4800	500	43	Image annotation
Yeast	2417	103	14	Gene functional analysis
Scene	2407	2940	6	Natural scene classification

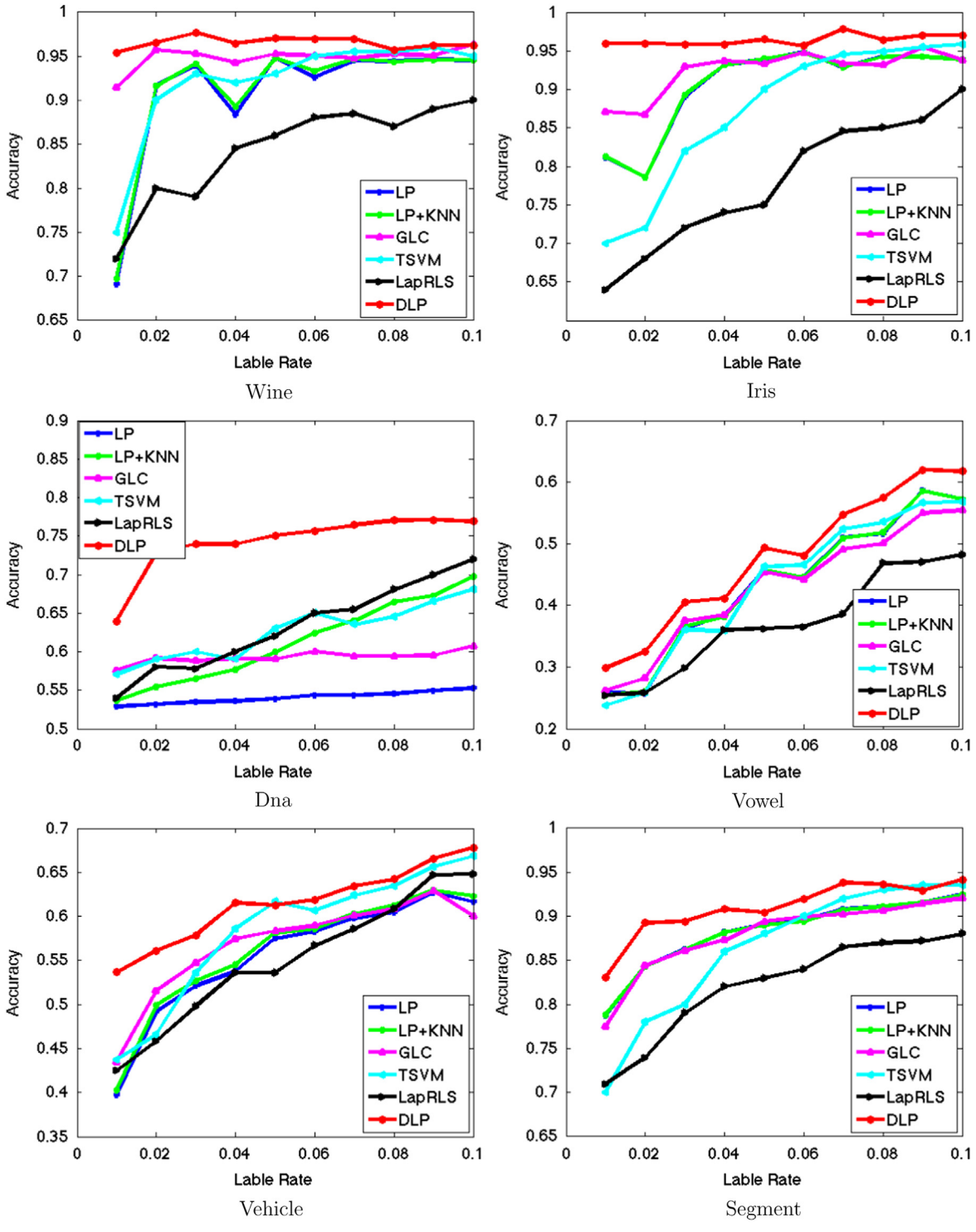

Fig. 6. Results on six multi-class datasets. Label rate refers to the percentage of labeled data.

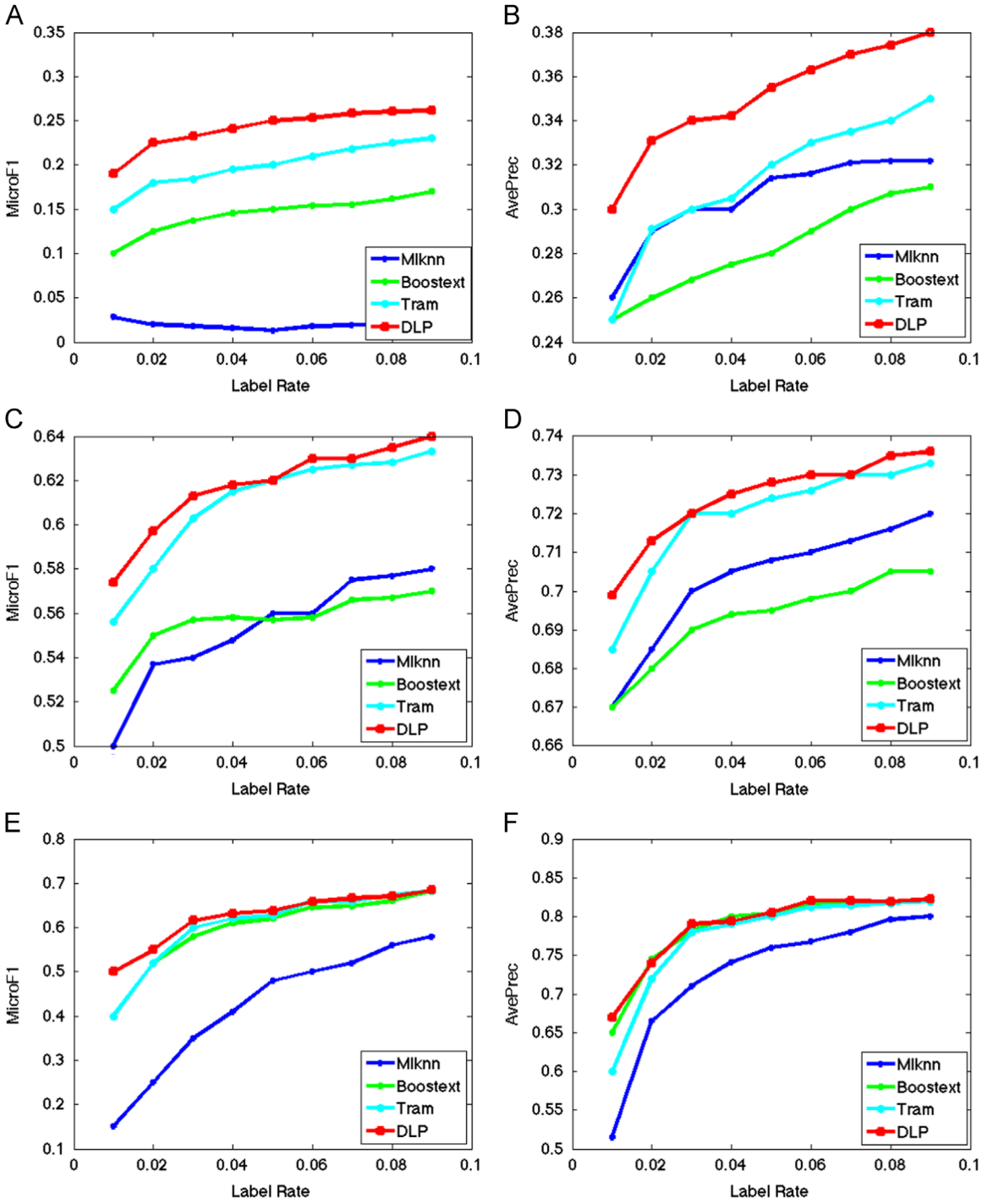

Fig. 7. The comparisons in three multilabel datasets. (a) MicroF1 on Corel. (b) Average Precesion on Corel. (c) MicroF1 in Yeast. (d) Average Precesion on Yeast. (e) MicroF1 on Scene. (f) Average Precesion on Scene.

given to describe the same images. The proposed DLP can be naturally generalized to multi-modality learning. We compare MDLP with both LP and DLP on a single similarity graph and one of the existing competing alternatives: Co-Tran [4] which combines both co-training and label propagation together.

We test our multi-modality extension of DLP on two datasets. One is a popular shape dataset MPEG7, which consists of 70 classes of shapes with 20 images per class. We use two different shape descriptors: Shape Context [6] and Inner Distance [29]. Shape Context describes a shape by spatial histogram of the contours while Inner Distance gives a representation from the skeletons of a shape. We randomly sample k shapes from each class and use them as labeled examples. The rest are treated as unlabeled data. For each number of k, we sample 100 different times and record the average accuracy. The comparison is given in Fig. 11(A). We can see that our MDLP outperforms the existing alternatives significantly. To provide a vivid sense of why our MDLP works better, we show the original similarity networks and the learned similarity by MDLP in Fig. 8. We can see that MDLP can generate a much cleaner similarity graph with stronger block structures therefore gaining significant improvement. Sensitivity test on MPEG7 is also performed (Fig. 9). We vary the hyper-parameters

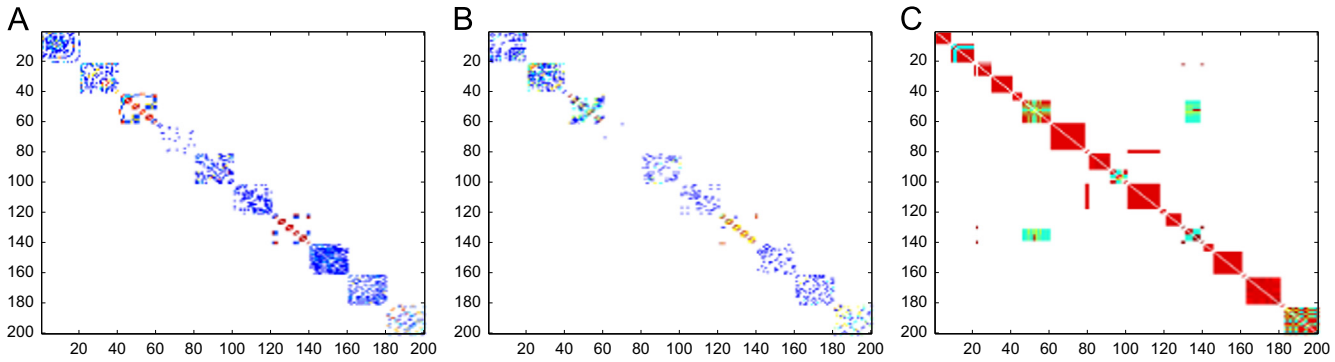

Fig. 8. Similarity graphs on MPEG7 dataset. We randomly choose 10 classes of shapes from MPEG7 dataset, and show their similarity heatmaps. (A)–(B) show the similarity by Shape Context [6] and Inner Distance [29], respectively. (C) shows the fused similarity obtained from our DLP. Enhanced and cleaner block structures are observed in this new similarity, which leads to higher classification accuracy.

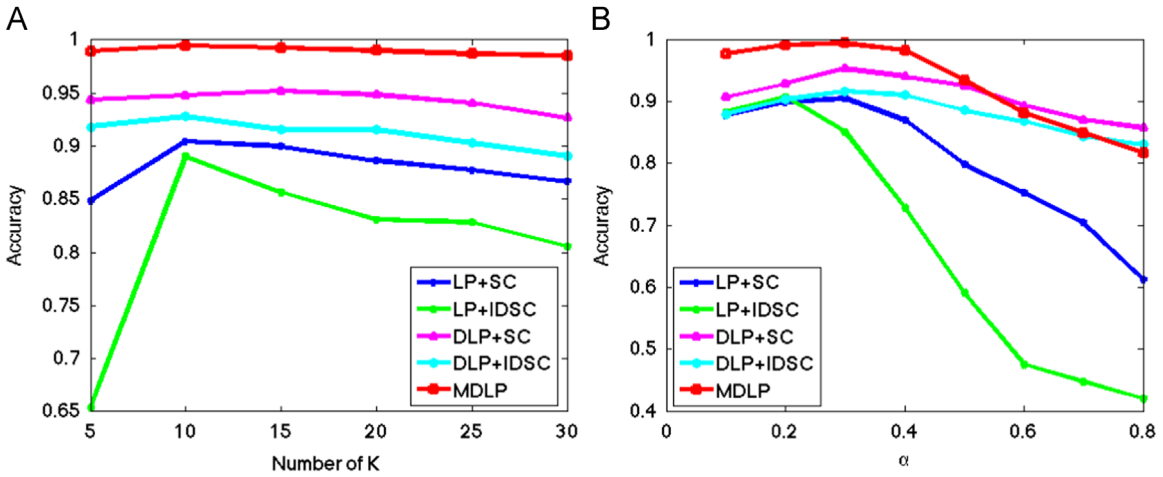

Fig. 9. Sensitivity test on MPEG7 dataset with MDLP. (A) shows the accuracies with different choices of K. (B) shows the accuracies with different choices of α.

Fig. 10. Sample images chosen from Caltech 101.

used in our DLP and report the corresponding results. It shows that the proposed DLP is robust to the choices of hyper-parameters (Fig. 9).

We also test our algorithm on the well-known Caltech-101 dataset [16] which consists of 101 classes and a collection of background images. We selected 12 classes (including animals, faces, buildings) from Caltech-101, which contains a total of 2788 images. These classes are chosen due to the relatively large number of available images within the category. The number of images per category varies from 41 to 800, most of which are medium resolution, i.e. about 300×200 pixels. Fig. 10 shows some samples of the subset.

We use two kinds of variants of SIFT feature: SIFT with locality-constrained linear coding (siftLLC) [48] and SIFT with Spatial Pyramid Matching (siftSPM) [26]. The SIFT features are both extracted from 16×16 pixel patches on a grid with step size of 8 pixels. The codebooks are obtained by standard K-means clustering with the codebook size 2048. The distance between two images is obtained by the χ^2 distance between two feature vectors. We vary the proportion of labels data and report the results of prediction accuracy in Fig. 11(B).

It is observed that our MDLP can well integrate multiple descriptions of image objects and outperform label propagation on either single view or multiple views. Why is the proposed dynamic label propagation (DLP) capable of improving multi-class/multi-label/multi-modality semi-supervised classifications? The key idea is that when you have noisy data and labels, two things can be exploited to better the performance. First, local structures in a graph provide a more trustworthy tool especially to transductive methods. It prevents high-order error propagation and also improves transitive similarities. Second, label information, even if not perfect, can provide correlations between the label space. This preliminary label information contains some global structural information of the graph and therefore can be well incorporated into the graph in turn to help improve the graph structures.

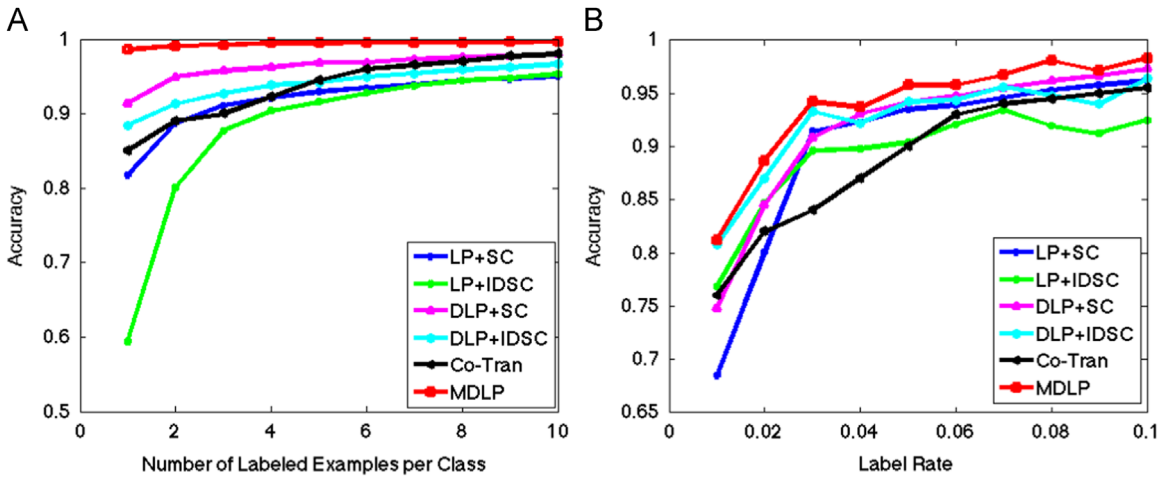

Fig. 11. Results of MDLP with different label rates. (A) shows the accuracies on MPEG7. (B) shows the accuracies on Caltech101.

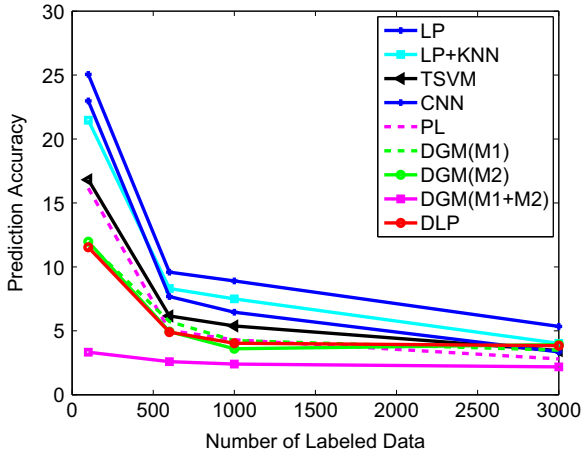

Fig. 12. Results of DLP on MNIST dataset with different label rates.

5.4. DLP on large dataset

We also demonstrate the scalability of DLP on large scale dataset, with an additional comparison with recent deep learning models. First, we compare with one of the most widely-used deep learning models in computer vision: Convolutional Neural Network (CNN) [24]. This model is served as a baseline because it only uses labeled data in training. Second, we compare with two recent semi-supervised deep learning models [28,22]. Pseudo-Label [28] (PL) trains deep network using both unlabeled and labeled data by utilizing the labels with maximum predicted probability. Another more sophisticated work called deep generative model (DGM) [22] proposed two models: Model One (M1) aims to construct a latent feature embedding which allows for higher moments of the data to be captured by the density model. Another model (M2) describes the data as being generated by a latent class variable in addition to a continuous latent variable. The data is explained by the generative process, which can also be seen as a hybrid continuous–discrete mixture model where the different mixture components share parameters. Classification power can be greatly improved by stacking these two models together (M1+M2). Details are referred in [22].

These deep learning models need a large dataset to perform properly. We chose a widely used benchmark dataset MNIST [27], which consists of 60,000 digit images ranging from 0 to 9. Among the 50,000 training data, we selected different number of labeled images and report the accuracy on the test data. Results are shown in Fig. 12. We can observe that our DLP can still achieve comparable results with these recent advances using deep learning models while enjoying a much smaller complexity. Particularly, our DLP achieves a decent performance when the number of labeled data is small. Note that these deep learning models are not able to deal with multi-modality and multi-label classification problems.

6. Conclusion

In this paper, we have proposed a novel classification method named dynamic label propagation (DLP), which improves the discriminative power in multi-class/multi-label problems in the framework of semi-supervised learning. Our method explores the effect of labeled information and local structure in improving the transition matrix in semi-supervised learning. The significant performance improvement on toy data and some popular natural object images has demonstrated the effectiveness of DLP for multi-class/multi-label classification. Our future work will focus on providing deeper theoretical understanding of the approach.

Conflict of interest

None declared.

Acknowledgment

This research was funded by the Canada Research Chairs Program through a Tier I Chair award (950-219525) to the second author, and by the Natural Sciences and Engineering Research Council of Canada via a grant (RPGIN/4557-2011) to the second author.

References

[1] E. Allwlein, R. Schapire, Y. Singer, Reducing multiclass to binary: a unifying approach for margin classifier, J. Mach. Learn. Res. (2000) 113–141.
[2] A. Azran, The rendezvous algorithm: multiclass semi-supervised learning with Markov random walks, in: Proceedings of ICML, 2007, pp. 49–56.
[3] P. Bachman, O. Alsharif, D. Precup, Learning with pseudo-ensembles, in: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Weinberger (Eds.), Advances in Neural Information Processing Systems, vol. 27, Curran Associates, Inc., Montreal, Canada, 2014, pp. 3365–3373, URL: ⟨http://papers.nips.cc/paper/5487-learning-with-pseudo-ensembles.pdf⟩.

[4] X. Bai, B. Wang, C. Yao, W. Liu, Z. Tu, Co-transduction for shape retrieval, IEEE Trans. Image Process. 21 (2012) 2747–2757.
[5] M. Belkin, P. Niyogi, V. Sindhwani, On manifold regularization, in: Proceedings of AISTAT, 2005.
[6] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 509–522.
[7] A.M. Bensaid, L.O. Hall, J.C. Bezdek, L.P. Clarke, Partially supervised clustering for image segmentation, Pattern Recognit. 29 (1996) 859–871.
[8] M.B. Blaschko, C.H. Lampert, A. Gretton, Semi-supervised Laplacian regularization of kernel canonical correlation analysis, in: Proceedings of ECMLKDD, 2008, pp. 133–145.
[9] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in: Proc. of COLT, 1998.
[10] M.R. Boutell, J. Luo, X. Shen, C.M. Brown, Learning multi-label scene classification, Pattern Recognit. 37 (2004) 1757–1771.
[11] H. Chang, D.-Y. Yeung, Locally linear metric adaptation with application to semi-supervised clustering and image retrieval, Pattern Recognit. 39 (2006) 1253–1264.
[12] G. Chen, Y. Song, F. Wang, Semi-supervised multi-label learning by solving a Sylvester equation, Proc. SDM 6 (2008) 28–30.
[13] K. Crammer, Y. Singer, A new family of online algorithms for category ranking, in: Proceedings of SIGIR, 2002.
[14] P. Duygulu, K. Barnard, J.F. de Freitas, D.A. Forsyth, Object recognition as machine translation: learning a lexicon for a fixed image vocabulary, in: Computer Vision ECCV 2002, Springer, Copenhagen, Denmark, 2002, pp. 97–112.
[15] A. Elisseeff, J. Weston, A kernel method for multi-labeled classification, in: NIPS, vol. 14, 2001, pp. 681–687.
[16] L. Fei-Fei, R. Fergus, P. Perona, One-shot learning of object categories, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 594–611.
[17] A. Goldberg, X.J. Zhu, B. Recht, J. Xu, R. Nowak, Transduction with matrix completion: three birds with one stone, in: Advances in Neural Information Processing Systems, vol. 23, 2010, pp. 757–765.
[18] N. Grira, M. Crucianu, N. Boujemaa, Active semi-supervised fuzzy clustering, Pattern Recognit. 41 (2008) 1834–1844.
[19] S. Har-peled, D. Roth, D. Zimak, Constraint classification for multiclass classification and ranking, in: Proceedings of NIPS, MIT Press, Vancouver, BC, 2003, pp. 785–792.
[20] J. Jiang, B. Wang, Z. Tu, Unsupervised metric learning by self-smoothing operator, in: ICCV, 2011, pp. 794–801.
[21] F. Kang, R. Jin, R. Sukthankar, Correlated label propagation with application to multi-label learning, in: Proceedings of CVPR, 2006, pp. 1719–1726.
[22] D.P. Kingma, S. Mohamed, D.J. Rezende, M. Welling, Semi-supervised learning with deep generative models, in: Advances in Neural Information Processing Systems, 2014, pp. 3581–3589.
[23] X. Kong, M.K. Ng, Z.-H. Zhou, Transductive multi-label learning via label set propagation, TKDE 99 (2011).
[24] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[25] S. Lafon, A.B. Lee, Diffusion maps and coarse-graining: a unified framework for dimensionality reduction, graph partitioning, and data set parameterization, IEEE Trans. PAMI 28 (2006) 1393–1403.
[26] S. Lazebnik, C. Schmid, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proceedings of CVPR, 2006.
[27] Y. Lecun, C. Cortes, The MNIST database of handwritten digits, ⟨http://yann.lecun.com/exdb/mnist/⟩.
[28] D.-H. Lee, Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks, in: Workshop on Challenges in Representation Learning, Proceedings of ICML, 2013.
[29] H. Ling, D.W. Jacobs, Shape classification using the inner-distance, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 286–299.
[30] Y. Liu, R. Jin, L. Yang, Semi-supervised multi-label learning by constrained non-negative matrix factorization, in: Proceedings of AAAI, 2006, pp. 421–426.
[31] Y. Liu, D. Zhang, G. Lu, W.-Y. Ma, A survey of content-based image retrieval with high-level semantics, Pattern Recognit. 40 (2007) 262–282.
[32] K. Lu, J. Zhao, D. Cai, An algorithm for semi-supervised learning in image retrieval, Pattern Recognit. 39 (2006) 717–720.
[33] L. Qiao, S. Chen, X. Tan, Sparsity preserving projections with applications to face recognition, Pattern Recognit. 43 (2010) 331–341.
[34] S. Rogers, M. Girolami, Multi-class semi-supervised learning with the e-truncted multinomial probit Gaussian process, J. Mach. Learn. Res. (2007) 17–32.
[35] J. Rousu, C. Saunders, S. Szedmak, J. Shawe-Taylor, On maximum margin hierarchical multi-label classification, in: Proceedings of NIPS Workshop on Learning With Structured Outputs, 2004.
[36] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.
[37] A. Saffari, C. Leistner, H. Bischof, Regularized multi-class semi-supervised boosting, in: CVPR, 2009.
[38] R.E. Schapire, Y. Singer, Boostexter: a boosting-based system for text categorization, Mach. Learn. 39 (2000) 135–168.
[39] V. Sindhwani, S.S. Keerthi, Large scale semi-supervised linear svms, in: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2006, pp. 477–484.
[40] Y. Song, F. Nie, C. Zhang, S. Xiang, A unified framework for semi-supervised dimensionality reduction, Pattern Recognit. 41 (2008) 2789–2799.
[41] Y. Song, C. Zhang, J. Lee, Graph based multi-class semi-supervised learning using Gaussian process, in: IAPR Workshop on Structural, Syntactic, and Statistical Pattern Recognition, 2006, pp. 450–458.
[42] A. Subramanya, S. Petrov, F. Pereira, Efficient graph-based semi-supervised learning of structured tagging models, in: Proceedings of EMNLP, 2010, pp. 167–176.
[43] J.B. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2322.
[44] G. Tsoumakas, I. Katakis, Multi-label classification: an overview, Int. J. Data Wareh. Min. (IJDWM) 3 (2007) 1–13.
[45] H. Valizadegan, R. Jin, A.K. Jain, Semi-supervised boosting for multi-class classification, in: Proceedings of ECML PKDD, 2008, pp. 522–537.
[46] B. Wang, J. Jiang, W. Wang, Z.-H. Zhou, Z. Tu, Unsupervised metric fusion by cross diffusion, in: CVPR, 2012, pp. 2997–3004.
[47] B. Wang, Z. Tu, Affinity learning via self-diffusion for image segmentation and clustering, in: CVPR, 2012, pp. 2312–2319.
[48] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: Proceedings of CVPR, 2010.
[49] K.Q. Weinberger, J. Blitzer, L.K. Saul, Distance metric learning for large margin nearest neighbor classification, in: Proceedings of NIPS, 2006, pp. 1473–1480.
[50] S. Xiang, F. Nie, C. Zhang, Learning a Mahalanobis distance metric for data clustering and classification, Pattern Recognit. 41 (2008) 3600–3612.
[51] E.P. Xing, A.Y. Ng, M.I. Jordan, S.J. Russell, Distance metric learning with application to clustering with side-information, in: Proceedings of NIPS, 2002, pp. 505–512.
[52] L. Xu, D. Schuurmans, Unsupervised and semi-supervised multi-class support vector machines, in: Proceedings of AAAI, 2005.
[53] X. Yang, X. Bai, L. Latecki, Z Tu, Improving shape retrieval by learning graph transduction, in: Proceedings of ECCV, 2008.
[54] Z.-J. Zha, T. Mei, J. Wang, Z. Wang, X.-S. Hua, Graph-based semi-supervised learning with multiple labels, J. Vis. Commun. Image Represent. 20 (2009) 97–103.
[55] M.-L. Zhang, Z.-H. Zhou, Ml-knn: a lazy learning approach to multi-label learning, Pattern Recognit. 40 (2007) 2038–2048.
[56] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, B. Schölkopf, Learning with local and global consistency, in: Proceedings of NIPS, MIT Press, Vancouver, BC, 2004, pp. 321–328.
[57] S. Zhu, X. Ji, W. Xu, Y. Gong, Multi-labeled classification using maximum entropy method, in: Proceedings of SIGIR, 2005, pp. 274–281.
[58] X. Zhu, Semi-supervised Learning with Graphs (Doctoral thesis), Department of Computer Science, Carnegie Mellon University, 2005.

Bo Wang received the B.E. degree in electronic information engineering from Huazhong University of Technology and Science, Wuhan, China, in 2010. He then received his master degree in the Department of Computer Science, University of Toronto, Toronto, ON, Canada. Now he is currently working on his Ph.D. degree in the Department of Computer Science at Stanford University, USA. His research interests include computer vision, machine learning, numerical analysis and computational bioinformatics.

John Tsotsos is Distinguished Research Professor of Vision Science at York University. He is Director of the Centre for Innovation in Computing at Lassonde, Canada Research Chair in Computational Vision, and is a Fellow of the Royal Society of Canada. He received his doctorate in Computer Science from the University of Toronto. He did a postdoctoral fellowship in Cardiology at Toronto General Hospital and then joined the University of Toronto on faculty in both Computer Science and in Medicine, where he stayed for 20 years, 10 of which were as a Fellow of the Canadian Institute for Advanced Research. He then moved to York University serving as Director of the Centre for Vision Research for 7 years. Visiting positions were held at the University of Hamburg, Polytechnical University of Crete, Center for Advanced Studies at IBM Canada, INRIA Sophia-Antipolis, and the Massachusetts Institute of Technology. He has published in computer science, neuroscience, psychology, robotics and biomedicine. Current research has a main focus in developing a comprehensive theory of visual attention in humans. A practical outlet for this theory forms a second focus, embodying elements of the theory into the vision systems of mobile robots.