Neural Networks 146 (2022) 174–180

Contents lists available at ScienceDirect

Neural Networks

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

Neural Networks Letter

Multiple-view flexible semi-supervised classification through consistent graph construction and label propagation

Najmeh Ziraki [d], Fadi Dornaika [a,b,c,*], Alireza Bosaghzadeh [d]

[a] School of Computer and Information Engineering, Henan University, Kaifeng, China
[b] University of the Basque Country UPV/EHU, San Sebastian, Spain
[c] IKERBASQUE, Basque Foundation for Science, Bilbao, Spain
[d] Shahid Rajaee Teacher Training University, Tehran, Iran

ARTICLE INFO

Article history:
Received 27 June 2021
Received in revised form 4 October 2021
Accepted 11 November 2021
Available online 18 November 2021

Keywords:
Multi-view semi-supervised classification
Information fusion
Graph-based data smoothness
Graph construction

ABSTRACT

Graph construction plays an essential role in graph-based label propagation since graphs give some information on the structure of the data manifold. While most graph construction methods rely on predefined distance calculation, recent algorithms merge the task of label propagation and graph construction in a single process. Moreover, the use of several descriptors is proved to outperform a single descriptor in representing the relation between the nodes. In this article, we propose a Multiple-View Consistent Graph construction and Label propagation algorithm (MVCGL) that simultaneously constructs a consistent graph based on several descriptors and performs label propagation over unlabeled samples. Furthermore, it provides a mapping function from the feature space to the label space with which we estimate the label of unseen samples via a linear projection. The constructed graph does not rely on a predefined similarity function and exploits data and label smoothness. Experiments conducted on three face and one handwritten digit databases show that the proposed method can gain better performance compared to other graph construction and label propagation methods.

1. Introduction

Graph-based classification algorithms have received a lot of attention due to their capability in revealing the underlying structure of the data manifold (Wang, Mezlini, Demir, Fiume, Tu, Brudno, Haibe-Kains, & Goldenberg, 2014; Wang, Yang, Liu, & Fujita, 2019; Zheng, Liu, Chen, An, & Zhang, 2020). Moreover, while there exists a huge amount of data, very few of them may have labels, and the majority of the available data samples are without any label which got excluded from the training phase when we adopt supervised learning. Hence, the use of semi-supervised learning algorithms that can simultaneously use labeled and unlabeled samples is of more interest compared to supervised learning methods which only adopt labeled samples Dornaika and Bosaghzadeh (2015), Dornaika, Dahbi, Bosaghzadeh, and Ruichek (2017), Karasuyama and Mamitsuka (2013). As a result, graph-based semi-supervised learning algorithms can overcome the above-mentioned limitations (An, Chen, & Yang, 2017; Liu, Lai, Ou, Zhang, & Zheng, 2020; Tong, Gray, Gao, Chen, & Rueckert, 2017).

While most of these approaches rely on a single feature (consequently, they construct a single graph and use it for label propagation) (Dornaika & Bosaghzadeh, 2015; Dornaika, Bosaghzadeh, & Raducanu, 2013; Dornaika, Dahbi et al., 2017; Zhou, Bousquet, Lal, Weston, & Schölkopf, 2004), recent techniques use several sources of information to extract different features and consequently, better estimate the underlying manifold structure (An et al., 2017; Angelou, Solachidis, Vretos, & Daras, 2019; Wang et al., 2014; Zheng et al., 2020) .

Most of the previous graph-based learning methods consider the graph construction and the learning (classification, regression, clustering, etc.) as two separate tasks (Bahrami, Bosaghzadeh, & Dornaika, 2019; Tong et al., 2017) where in the first phase, one or several graphs are constructed (and merged) and then the label propagation is performed. However, recent methods do not consider them as two separate tasks and instead, they combine the graph construction and the post-learning task in a single framework (Huang, Kang, Tsang, & Xu, 2019; Lin, Liao, Sun, Chen, & Zhao, 2017; Nie, Cai, & Li, 2017; Wang & Tsotsos, 2016). The experimental results proved that the fusion of these two steps can enhance the learning performance.

One of the major drawbacks in most of the single-view and multi-view graph-based learning algorithms is how they estimate the label of unseen samples. In this case, one has to reconstruct

* Corresponding author at: University of the Basque Country UPV/EHU, San Sebastian, Spain.
E-mail addresses: najmeh.ziraki@gmail.com (N. Ziraki), fadi.dornaika@ehu.eus (F. Dornaika), a.bosaghzadeh@sru.ac.ir (A. Bosaghzadeh).

https://doi.org/10.1016/j.neunet.2021.11.015

the affinity graph using the new samples and perform the label propagation on the new samples (Bosaghzadeh & Dornaika, 2020). However, when we have a stream of samples coming in sequence, this solution becomes less possible since repeating the graph construction and label propagation is a time-consuming task. This is due to the fact that these algorithms which are called transductive, cannot directly estimate the label of test samples and for the new samples, one has to repeat the whole process. On the other hand, inductive algorithms learn an explicit mapping to estimate the label of unseen samples; hence, for the new samples, it is not necessary to reconstruct a new graph and perform label propagation, but the labels can be calculated via a linear mapping (Bahrami et al., 2019; Nie, Xu, Tsang, & Zhang, 2010).

In multi-view learning, since there exist several features, one has to look for a fusion algorithm to merge these sources of information. Most multi-view graph-based algorithms adopt several pre-constructed graphs (e.g., KNN graph), and then merge them to obtain a unified consistent graph (An et al., 2017; Bahrami et al., 2019). However, recent algorithms directly estimate the affinity graph using either a single feature or several descriptors. There are two main groups of methods for inferring the graph from data: (i) methods that rely on data-self representativeness (Dornaika & Bosaghzadeh, 2015; Dornaika, Kejani, & Bosaghzadeh, 2017; Kang, Shi, Huang, Chen, Pu, Zhou, & Xu, 2020), and (ii) methods that rely on data smoothness (Nie et al., 2017; Wang et al., 2019).

In summary, existing multi-view graph-based semi-supervised learning methods which estimate a flexible non-linear embedding cannot overcome the drawbacks mentioned above.

In this article, we propose a multi-view graph-based semi-supervised learning algorithm that can overcome the above-mentioned limitations. The proposed algorithm has the following characteristics:

- It combines the label propagation and multi-view graph construction in a unified framework.
- It jointly estimates the projection matrix, the labels of unlabeled samples, the consistent graph, and the views' weights
- It constructs a single consistent graph considering the information in different views.
- It uses several features and fuses the information available in different views to enhance accuracy.
- For unseen samples, the proposed method can directly estimate the labels via a linear transformation which makes it suitable for large-scale label propagation.

The remaining of this article is organized as follows: Section 2 introduces the graph and basic assumptions in graph-based label propagation. It also briefly reviews some related works. Section 3 is dedicated to the proposed method. Section 4 reports the experimental results. Finally, we conclude the article in Section 5.

2. Related work

2.1. Graph-based label propagation

Consider $\mathbf{X} \in \mathbb{R}^{d\times N}$ as the data matrix containing $N = l + u$ samples each with the dimensionality of d, where l samples have labels and u samples are unlabeled. Without loss of any generality, it is assumed that the labeled samples are represented by the first l columns of the matrix $\mathbf{X}$. There exist a respective binary label matrix $\mathbf{Y} = [\mathbf{Y}_l, \mathbf{Y}_u] = [\mathbf{y}_1; \mathbf{y}_2; ....; \mathbf{y}_l; \mathbf{y}_{l+1}; ...; \mathbf{y}_{l+u}] \in \mathbb{R}^{N\times C}$ where $\mathbf{y}_i$ is a C dimensional vector that determines the class for the samples $\mathbf{x}_i$. For an arbitrary sample $\mathbf{x}_i$ that belongs to the class c, the cth element of $\mathbf{y}_i$ is set to one and the rest are zero. For unlabeled samples (i.e., $\mathbf{Y}_u$), $y_{ij} = 0| \ \forall j$. Moreover, we define a soft label matrix $\mathbf{F} = [\mathbf{f}_1; \mathbf{f}_2; ....; \mathbf{f}_N] \in \mathbb{R}^{N\times C}$ where F_{ic} represents the probability of sample $\mathbf{x}_i$ belonging to the class c.

We define an undirected weighted graph as $\mathbf{G} = \{\mathbf{X}, \mathbf{W}\}$ where the data matrix $\mathbf{X}$ contains the nodes and the symmetric affinity matrix $\mathbf{W}$ represents the pairwise similarity between the nodes. Moreover, we define $\mathbf{L} = \mathbf{D} - \mathbf{W}$ as the unnormalized Laplacian matrix where $\mathbf{D}$ is the diagonal degree matrix where $D_{ii} = \sum_{j=1}^{N} W_{ij}$ shows the degree for the node $\mathbf{x}_i$.

2.2. Review of single and multi view graph-based learning methods

The basic assumption in graph-based learning is that close samples share similar labels.

Gaussian Field and Harmonic Functions (GFHF) algorithm (Zhu, Ghahramani, & Lafferty, 2003) exploits this assumption and attempts to estimate the label matrix, $\mathbf{F}$, by minimizing the following criterion:

$$\min_{\mathbf{F}} \sum_{i=1}^{N}\sum_{j=1}^{N} \|\mathbf{f}_i - \mathbf{f}_j\|^2 W_{ij} = \min_{\mathbf{F}} Tr(\mathbf{F}^T \mathbf{L}\, \mathbf{F}) \tag{1}$$

where $Tr(.)$ denotes the trace of a matrix. It states that for the two samples $\mathbf{x}_i$ and $\mathbf{x}_j$ with corresponding soft labels $\mathbf{f}_i$ and $\mathbf{f}_j$ and the similarity W_{ij}, if the similarity is high (i.e., large W_{ij}) the difference between the corresponding soft labels of the two samples should be low.

LGC (Zhou et al., 2004) adds the label fitness term to Eq. (1). Its objective function is:

$$\min_{\mathbf{F}} \sum_{i,j=1}^{N} \| \frac{1}{\sqrt{D_{ii}}} \mathbf{f}_i - \frac{1}{\sqrt{D_{jj}}} \mathbf{f}_j \|^2 W_{ij} + \mu \sum_{i=1}^{l} \|\mathbf{f}_i - \mathbf{y}_i\|^2 \tag{2}$$

where μ is the balance parameter between the label smoothness and label fitness terms.

As explained in the introduction, the adoption of several features for graph construction and label propagation has absorbed attention due to the complementary information in different views; hence different graphs. However, a drawback in adopting several features is the possible noisy features and graphs. In Karasuyama and Mamitsuka (2013) the authors adopted the weighted fusion of several graphs, while at the same time maintaining the sparsity for the adopted graphs. Their mathematical objective function was

$$\begin{aligned} &\min_{\mathbf{F},\lambda} \sum_{v=1}^{V} \lambda_v (\mathbf{F}^T \mathbf{L}_v \mathbf{F}) + \mu_1 \sum_{i=1}^{l} \|\mathbf{f}_i - \mathbf{y}_i\|^2 + \mu_2 \|\lambda\|_2^2 \\ &s.t.\ \lambda^T \mathbf{1} = 1,\ \boldsymbol{\lambda} \geq 0 \end{aligned} \tag{3}$$

where $\mathbf{L}_v$ is the Laplacian of the graph constructed from the vth view and λ_v is contribution of the vth view. Recent fusion algorithms insert the label space as a source of information to further enhance the performance. The authors in Wang and Tsotsos (2016) divide their work into Kernel Fusion and Kernel Diffusion steps, where in the Kernel Fusion step, they combine the information in the label space and the data space, hoping that label information can boost the accuracy of label propagation. Similarly, in Lin et al. (2017), along with several features, the authors used the information in the label space in their optimization to further enhance the accuracy.

While above mentioned algorithms adopt graph construction algorithms like KNN graph with Gaussian kernel, recent algorithms automatically construct the graph via some predefined criterion. For instance, Wang et al. (2019) used manifold learning and sparse representation to construct a graph for each view. Their adopted criterion is

$$\begin{aligned} &\min_{\mathbf{S}^v} \sum_{i,j=1}^{n} \|\mathbf{x}_i^v - \mathbf{x}_j^v\|_2^2 s_{ij}^v + \alpha \sum_{i=1}^{n} \|\mathbf{s}_i^v\|_2^2 \\ &s.t.\ s_{ii}^v = 0,\ s_{ij}^v \geq 0,\ \mathbf{1}^T \mathbf{s}_i^v = 1 \end{aligned} \tag{4}$$

where s_{ij}^v represents the calculated similarity between the pair $\mathbf{x}_i$ and $\mathbf{x}_j$ in the vth view. Another recent approach is to construct the graph based on the data Self-expressiveness property which states that each data sample can be expressed as a linear combination of other samples (Kang et al., 2020). The mathematical form of its multi-view version can be written as

$$\min_{\mathbf{Z}^v} \sum_{v=1}^{V} \|\mathbf{X}^v - \mathbf{X}^v\mathbf{S}^v\|_F^2 + \alpha\|\mathbf{S}^v\|^2 \quad \mathbf{S}^v \geq 0 \tag{5}$$

where $\mathbf{S}^v$ is the similarity matrix in the vth view. In the fusion step, both Kang et al. (2020) and Wang et al. (2019) use the idea of consensus graph, which states that the graph of any view is a perturbation of the consensus graph. The consensus graph criterion can be written in its simple form as follows.

$$\min \sum_{v=1}^{V} \|\mathbf{S}^v - \mathbf{U}\|^2 \tag{6}$$

where $\mathbf{U}$ is the consensus graph between the graphs of the different views.

2.3. Review of flexible manifold embedding

While Zhu et al. (2003) and Zhou et al. (2004) work only on available samples (labeled and unlabeled samples), they cannot handle o unseen samples. The authors of Flexible manifold embedding (Nie et al., 2010) additionally adopted a projection matrix that maps data to the label space (inductive model). The objective function of FME is given by:

$$\min_{\mathbf{F},\mathbf{Q},\mathbf{b}} Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + Tr((\mathbf{F}-\mathbf{Y})^T\mathbf{U}(\mathbf{F}-\mathbf{Y})) + \mu(\|\mathbf{Q}\|^2 + \gamma\|\mathbf{X}^T\mathbf{Q} + \mathbf{1}\mathbf{b}^T - \mathbf{F}\|^2) \tag{7}$$

where the first term is the label smoothness, the second one is the label fitness term, and the third term is the regularization of the projection matrix plus the error of label fitting using the projection matrix. $\mathbf{Q}$ is the projection matrix, $\mathbf{b}$ is the bias vector, μ and γ are balancing parameters and $\|.\|^2$ denotes the ℓ_2 norm of a matrix. The diagonal matrix $\mathbf{U}$ has nonzero values for the labeled samples and zero otherwise. The solution to the optimization problem (7) gives $\mathbf{F}$, $\mathbf{Q}$, and $\mathbf{b}$ via the following equations (see Nie et al. (2010)):

$$\mathbf{Q} = \gamma[\gamma\,\mathbf{X}\mathbf{H}_c\mathbf{X}^T + \mathbf{I}]^{-1}\mathbf{X}\mathbf{H}_c\mathbf{F} \tag{8}$$

$$\mathbf{b} = \frac{1}{l+u}[\mathbf{F}^T\mathbf{1} - \mathbf{W}^T\mathbf{X}\mathbf{1}] \tag{9}$$

$$\mathbf{F} = (\mathbf{U} + \mathbf{L} + \mu\,\gamma\,\mathbf{H}_c - \mu\,\gamma^2\,\mathbf{M})^{-1}\mathbf{U}\,\mathbf{Y} \tag{10}$$

where $\mathbf{M} = \mathbf{X}_c^T\mathbf{X}_c(\gamma\,\mathbf{X}_c^T\mathbf{X}_c + \mathbf{I})^{-1}$, $\mathbf{X}_c$, and $\mathbf{H}_c = \mathbf{I} - \frac{1}{l+u}\mathbf{1}\mathbf{1}^T$ are the centered data matrix and the centering matrix, respectively.

3. Proposed method

We propose a Multiple-View Consistent Graph construction and Label propagation algorithm (MVCGL) that unifies the multi-view graph construction and label propagation in a single framework. Unlike previous works (Bahrami et al., 2019; Wang & Tsotsos, 2016), we do not use some previously constructed graphs but estimate a unified graph based on the information in different views. Thus, the targeted task is more complicated than the tasks addressed in the previous works.

Since the graph matrix is considered as an input in the FME framework, the data smoothness term is not taken into account in the FME framework. In our case, one of the sub-tasks is the estimation of a unified consistent graph over the multiple views. Thus, data and label smoothness will be used in our objective function to estimate this unified graph.

Data smoothness or manifold smoothness assumes that far samples should have low similarity (Similarity is encoded in the elements of the $\mathbf{S}$ matrix). Mathematically, the graph matrix $\mathbf{S}$ should minimize the following smoothness term:

$$\min_{\mathbf{S}} \frac{1}{2}\sum_{i,j=1}^{N}\|\mathbf{x}_i - \mathbf{x}_j\|^2 S_{ij} = \min_{\mathbf{S}} Tr(\mathbf{X}\mathbf{L}\mathbf{X}^T) \tag{11}$$

where $\mathbf{L}$ is the Laplacian matrix of the graph $\mathbf{S}$. We define the data smoothness term in multi-view learning as

$$\min_{\mathbf{S}} \sum_{v=1}^{V}\beta_v^p \sum_{i,j=1}^{N}\|\mathbf{x}_{vi} - \mathbf{x}_{vj}\|^2 S_{ij} = \min_{\mathbf{S}} \sum_{v=1}^{V}\beta_v^p\, Tr(\mathbf{X}_v\mathbf{L}\mathbf{X}_v^T) \tag{12}$$

where $\mathbf{x}_{vj}$ is the jth sample in the vth view, β_v is an unknown weight, $p > 1$ is a hyperparameter that avoids the trivial solution of choosing only one view, and V denotes the total number of views.

By adding Eq. (12) to the FME criterion (i.e., Eq. (7)), we have:

$$\begin{aligned}\min_{\beta,\mathbf{S},\mathbf{F},\mathbf{Q},\mathbf{b}} [\sum_{v=1}^{V}\beta_v^p\, Tr(\mathbf{X}_v\mathbf{L}\mathbf{X}_v^T) + \lambda\, Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + Tr((\mathbf{F}-\mathbf{Y})^T\mathbf{U}(\mathbf{F}-\mathbf{Y}))\\ +\gamma\,\|\mathbf{S}\|^2 + \mu(\|\mathbf{Q}\|^2 + \alpha\,\|\mathbf{X}_c^T\mathbf{Q} + \mathbf{1}\mathbf{b}^T - \mathbf{F}\|^2)]\\ s.t.\quad 0 \leq S_{ij} \leq 1,\quad \sum_{j=1}^{N} S_{ij} = 1\end{aligned} \tag{13}$$

where $\mathbf{S}$ is the unified similarity matrix and β_v is the weight of view v. By optimizing the above objective function, we can simultaneously estimate $\boldsymbol{\beta}$, $\mathbf{S}$, $\mathbf{F}$, $\mathbf{Q}$, and $\mathbf{b}$. Since in multi-view learning we have different features, to estimate the label of test samples one should use all these features. Consequently, we concatenate the features of all views such that they form a single descriptor. Hence, the data matrix $\mathbf{X}_c$ and the projection matrix $\mathbf{Q}$ have the dimensionality of $(\sum_{v=1}^{V} d^v) \times N$ and $(\sum_{v=1}^{V} d^v) \times C$, respectively.

To solve the problem presented in Eq. (13), we propose an iterative alternate solution. We proceed as follows.

Fix S and $\boldsymbol{\beta}$ and calculate Q, b, and F The minimization problem of Eq. (13) is reduced to:

$$\begin{aligned}\min_{\mathbf{F},\mathbf{Q},\mathbf{b}} \quad & \lambda\, Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + Tr((\mathbf{F}-\mathbf{Y})\mathbf{U}(\mathbf{F}-\mathbf{Y})^T)\\ & + \mu(\|\mathbf{Q}\|^2 + \alpha\,\|\mathbf{X}_c^T\mathbf{Q} + \mathbf{1}\mathbf{b}^T - \mathbf{F}\|^2)\end{aligned} \tag{14}$$

The above is the FME formulation in Eq. (7). Consequently, Eqs. (8), (9), and (10) can be adopted to calculate $\mathbf{Q}$, $\mathbf{b}$, and $\mathbf{F}$, respectively.

Fix F, Q, b, and $\boldsymbol{\beta}$ and calculate S The proposed objective function in Eq. (13) can be simplified to

$$\begin{aligned}\min_{\mathbf{S}} \quad & \sum_{v=1}^{V}\beta_v^p\, Tr(\mathbf{X}_v\mathbf{L}\mathbf{X}_v^T) + \lambda\, Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \gamma\,\|\mathbf{S}\|^2\\ s.t.\quad & 0 \leq S_{ij} \leq 1,\quad \sum_{j=1}^{N} S_{ij} = 1\end{aligned} \tag{15}$$

which can be written as

$$\begin{aligned}&\min_{\mathbf{S}} \sum_{v=1}^{V}\beta_v^p\sum_{i,j=1}^{N}\|\mathbf{x}_i^v - \mathbf{x}_j^v\|_2^2 S_{ij} + \lambda\sum_{i,j=1}^{N}\|\mathbf{f}_i - \mathbf{f}_j\|_2^2 S_{ij} + \gamma\,\|\mathbf{S}\|_2^2\\ &\equiv \min_{\mathbf{S}} \sum_{i,j=1}^{N} S_{ij}\left(\sum_{v=1}^{V}\beta_v^p\,\|\mathbf{x}_i^v - \mathbf{x}_j^v\|_2^2 + \lambda\,\|\mathbf{f}_i - \mathbf{f}_j\|_2^2\right) + \gamma\,\|\mathbf{S}\|_2^2\end{aligned}$$

$$\equiv \min_{\mathbf{s}} \sum_{i,j=1}^{N} [S_{ij}\, d_{ij} + \gamma\, S_{ij}^2] \quad s.t. \quad 0 \le S_{ij} \le 1, \quad \sum_{j=1}^{N} S_{ij} = 1 \tag{16}$$

where $d_{ij} = \sum_{v=1}^{V} \|\mathbf{x}_i^v - \mathbf{x}_j^v\|_2^2 + \lambda \|\mathbf{f}_i - \mathbf{f}_j\|_2^2$. Since Eq. (16) is independent between different i, we can deal with it individually for each row $\mathbf{s}_i$ as (the vector $\mathbf{d}_i$ is set to $(d_{i1}, \ldots, d_{iN})^T$):

$$\min_{s_i} \sum_{j=1}^{N} S_{ij}\, d_{ij} + \gamma\, S_{ij}^2 \equiv \min_{s_i} \|\mathbf{s}_i + \frac{1}{2\gamma}\mathbf{d}_i\|_2^2$$
$$s.t. \quad 0 \le S_{ij} \le 1, \quad \sum_{j=1}^{N} S_{ij} = 1 \tag{17}$$

A closed-form solution for $\mathbf{s}_i$ ($i = 1, \ldots, N$) can be found in Nie et al. (2017).

Fix F, Q, b, and S and calculate β Eq. (13) can be simplified as

$$\min_{\beta} \sum_{v=1}^{V} \beta_v^p\, C_v + \eta\,(\sum_{v=1}^{V} \beta_v - 1) \tag{18}$$

where $C_v = Tr(\mathbf{X}_v^T \mathbf{L} \mathbf{X}_v)$ and η is the Lagrangian multiplier.

To estimate the weight coefficients (i.e., β_v, $v = 1, \ldots, V$), we vanish the derivative of Eq. (18) w.r.t. β_v and obtain

$$p\beta_v^{p-1} C_v + \eta = 0 \Rightarrow \beta_v = (\frac{-\eta}{pC_v})^{\frac{1}{p-1}} \tag{19}$$

Expanding ($\sum_{v=1}^{V} \beta_v = 1$), we have

$$(\frac{-\eta}{pC_1})^{\frac{1}{p-1}} + (\frac{-\eta}{pC_2})^{\frac{1}{p-1}} + \cdots + (\frac{-\eta}{pC_V})^{\frac{1}{p-1}} = 1$$
$$\Rightarrow, \ (\frac{-\eta}{p})^{\frac{1}{p-1}} = \frac{1}{\sum_{v=1}^{V} \frac{1}{C_v}^{\frac{1}{p-1}}} \tag{20}$$

Inserting Eq. (20) in Eq. (19), we get:

$$\beta_v = \frac{\frac{1}{C_v}^{\frac{1}{p-1}}}{\sum_{v=1}^{V} (\frac{1}{C_v})^{\frac{1}{p-1}}} \tag{21}$$

Algorithm 1 summarizes the main steps for solving the problem (13). Once the model is estimated (i.e., **F**, **Q**, **b**, and **S** are recovered), the label (row vector) of any test sample can be predicted via the following mapping:

$$\mathbf{f} = \mathbf{x}_c^T \mathbf{Q} + \mathbf{b}^T \tag{22}$$

where $\mathbf{x}_c$ is the vector constructed by concatenating the features of the views V views.

Input: V data matrices $\mathbf{X}_c = [\mathbf{X}_1; \ldots; \mathbf{X}_V]$, Binary label of labeled samples $\mathbf{Y}_l$, Parameters α, μ, λ, and p
Output: Predicted soft label matrix **F**, Projection matrix **Q** and bias vector **b**, the coefficients β_v, $v = 1, \ldots, V$

1. Initialize the coefficient of each view β_v to $\frac{1}{V}$
2. Initialize the soft label matrix **F** = **Y**
3. Initialize the affinity graph **S**
   repeat
   Update the affinity matrix **S** by the solution in Nie et al. (2017).
   Update **F**, **Q** and **b** by Eqs. (10), (8), and (9).
   Update β by Eq. (21).
   until $|\mathbf{S}^t - \mathbf{S}^{t-1}| < threshold$;
4. The **F** matrix shows the label of unlabeled samples.
5. Estimate the label of unseen samples via Eq. (22).

Algorithm 1: Multiple-View Consistent Graph construction and Label propagation algorithm

4. Experiments

In this section, we evaluate the performance of the proposed method and compare it with that of several algorithms. The competing methods are single feature, feature concatenation, SNF (Wang et al., 2014), SMGI (Karasuyama & Mamitsuka, 2013), DGFLP (Lin et al., 2017) MLGC (An et al., 2017), and AMGL (Nie, Li, & Li, 2016).

4.1. Experimental setup

For the evaluation of the proposed method, we use five image databases, three are small face databases namely PF01[1], PIE (Sim, Baker, & Bsat, 2002), FERET (Phillips, Moon, Rizvi, & J. Rauss, 2000), COVIDx Dataset of chest X-ray images[2] and one large handwritten digit database namely MNIST (LeCun & Cortes, 2010). This database contains 60,000 train and 10,000 test images of handwritten digits from 0∼9 which constructs 10 classes.

Since a multi-view algorithm requires several features, from the adopted databases explained above, we extract several features. For face databases, we use LBP image (Ahonen, Hadid, & Pietikainen, 2006), Gabor (Shen & Bai, 2006) and Covariance descriptor (Cov) (Tuzel, Porikli, & Meer, 2006). The dimensions of these descriptors are respectively 900, 2560, and 405. The Gabor filter adopted five scales and eight orientations. The Cov descriptor adopted nine low-level feature maps with 3×3 blocks in the image.

The COVIDx dataset contains 13 892 chest X-ray images in three classes, namely: COVID19, normal and pneumonia. Although this dataset is commonly used for supervised classification, we use it to test the proposed semi-supervised method. For each X-ray image, we extract two image descriptors given by two different deep CNNs: ResNet50 and ResNet101 trained on the ImageNet dataset. These nets produce an image descriptor of 2048 elements. For the MNIST database, we used two deep convolutional neural networks, namely VGG-16 and VGG19 (Simonyan & Zisserman, 2014) that both were trained on the ImageNet database (Deng, Dong, Socher, Li, Li, & Fei-Fei, 2009). These nets produce an image descriptor of 4096 elements.

PCA is used for reducing the dimensionality of features such that 90% of the energy is preserved. After that, zero-mean and unit-variance are applied. We separately report the results for small and large databases. For small databases, we select l samples as labeled and the rest are used as unlabeled and the whole data is used in the training phase. In the large databases (i.e., Covidx and MNIST datasets, we split the data into training and test parts. The training samples are then treated similarly to the small datasets (l samples as labeled and the rest of the training data as unlabeled).

Since the accuracy depends on the data configuration, we create 10 splits for labeled and unlabeled samples and perform the experiments on these splits and then report the average and standard deviation of the accuracy. For each database, we

[1] nova.postech.ac.kr/special/imdb/imdb.html.

[2] https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md.

Table 1
Average accuracy and standard deviation of a single feature and the proposed method on PIE, FERET and PF01 databases. The optimal parameters are shown in the parenthesis $(k, \lambda, \mu, \alpha)$

PIE					
lab./class	Feature				
	Cov	Gabor	LBP	Concatenation	MVCGL
15	80.93 ± 3.71 (3,20,100,2)	91.79 ± 3.32 (5,20,100,2)	87.75 ± 3.9 (3,10,5,5)	94.31 ± 2.88 (3,0.0001,1,1000)	**94.39 ± 2.86** (3,10,1,1000)
17	81.93 ± 4.06 (3,10,2,5)	91.09 ± 3.08 (3,0.0001,1,100)	88.11 ± 2.99 (3,10,2,5)	94.54 ± 2.51 5,0.0001,1,1000)	**94.84 ± 2.47** (3,20,0.1,1000)
FERET					
lab./class	Feature				
	Cov	Gabor	LBP	Concatenation	MVCGL
3	59.17 ± 9.53 (5,20,100,100)	79.8 ± 8.96 (3,20,2,100)	72.01 ± 10.3 (5,20,1.,2)	83.98 ± 8.06 (3,10,0.1,100	**84.66 ± 7.74** (3,20,0.1,100
5	67,27 ± 18.67 (5,20,5,1)	86 ± 15.38 (3,10,0.1,10)	85.15 ± 17.66 (3,20,2,2)	90.72 ± 12.98 (3,0.0001,0.1,100)	**91.45 ± 12.95** (3,20,0.1,100)
PF01					
lab./class	Feature				
	Cov	Gabor	LBP	Concatenation	MVCGL
5	70.24 ± 3.34 (20,20,1000,5)	84.26 ± 3.76 (20,10,1000,100)	79.57 ± 2.55 (3,20,1,100)	91.3 ± 2.03 (15,0.0001,1,100)	**92.61 ± 1.74** (10,20,5,100)
7	71.82 ± 5.93 (15,20,10,100)	85.81 ± 4.93 (20,20,100,1)	80.71 ± 4.51 (20,20,1000,2)	93.22 ± 2.39 (20,20,1,100)	**93.7 ± 2.4** (20,10,2,100)

choose different numbers of labeled samples (i.e., l). For each l, we vary the parameters and calculate the accuracy and then select the parameters which give the highest accuracy as the best parameter. We then fix these parameters for the rest of the splits. The parameters of the proposed method are the balance parameters λ, γ, μ, and α. Also, we have the p parameter that should be greater than one to avoid trivial solution (i.e., one view is selected) and experimentally, we set it to two. For the parameters μ, α, and λ, we select a combination of labeled/ unlabeled data (i.e., split) as evaluation and performed a grid search to find the parameters which gives the highest accuracy. We then fix the obtained parameters for the remaining splits in this experiment. More specifically, for the number of neighbors parameter used in graph optimization (i.e., k), we take the set $\{3, 5, 7, 10, 15, 20\}$, for the regression model parameters (i.e., μ and α), we take the set $\{0.0001, 0.001, 0.01, 0.1, 2, 5, 10, 100, 1000\}$ and for the label smoothing parameter (i.e., λ), we use the $\{0.0001, 0.001, 0.01, 0.1, 2, 5, 10, 20\}$. The parameter γ is set according to Nie et al. (2017).

4.2. Experimental results

Small databases. For the datasets with less than 2000 images, we split the data into labeled and unlabeled samples and use them for training. We randomly select l samples as labeled and the rest are left as unlabeled samples. This configuration is adopted for small databases since the number of samples is small in these databases. For initialization, we construct the graph using Eq. (17) because the $\mathbf{f}$ vectors are set to zero. Moreover, the initial coefficient vector β is set to $(\frac{1}{V}, \frac{1}{V}, \ldots, \frac{1}{V})$. For the single features, we construct a single graph and insert it into the proposed method.

We calculate the average accuracy and standard deviation on 10 splits of labeled/unlabeled samples and report them in Table 1 for the PIE, FERET, and PF01 databases. As we can see, the proposed method obtained higher accuracy compared to the use of a single feature and moreover, has a lower standard deviation compared to them. The numbers in the parenthesis show the adopted parameters.

Table 2
Average accuracy and standard deviation of the single feature and the fusion algorithm on COVIDx database. (Top) Results on the unlabeled samples (i.e., transductive setting). (Bottom) Results on the unseen (test) samples (i.e., inductive setting).

lab./class	Feature		
	View1	View2	MVCGL
40	60.13 ± 2.17 3,20,10,0.1	54.60 ± 2.56 20,0.1,0.001,1000	61.70 ± 1.38 5,20,10,0.1
120	64.71 ± 1.64 7,2,5,0.1	61.42 ± 1.16 3,5,5,2	65.59 ± 1.68 3,2,2,10
200	66.85 ± 1.39 5,0.0001,5,0.1	61.91 ± 1.09 3,0.1,10,0.1	67.58 ± 1.42 7,2,5,0.1

lab./class	Feature		
	View1	View2	MVCGL
40	60.86+2.53 5,20,10,0.1	51.77+7.84 20,0.1,0.001,1000	61.55+2.72 5,20,10,0.1
120	64.28 ± 1.50 7,2,5,0.1	60.96 ± 2.18 3,5,5,2	66.02 ± 1.42 3,2,2,10
200	64.99 ± 0.8 5,0.0001,5,0.1	61.90 ± 1.74 3,0.1,10,0.1	67.00 ± 0.8 7,2,5,0.1

Large databases. In the previous experiment, we consider that the whole data is available in the training phase; however, this is not the case in most real-world applications. In this experiment, we adopt a large database and divide the data into a training part (containing labeled and unlabeled samples) and a test part. We do not use test data in the training phase. We assume that the training data (i.e., labeled and unlabeled samples) are available in the training phase and are used by the proposed algorithm.

The COVIDx database contains chest X-ray images of 13 892 patients where 468 has 'COVID-19', 7966 are normal and 5458 have 'pneumonia'. From each class, we select 400 samples for training (among them l are labeled) and the rest are used as test. We report the average accuracy and standard deviation for the single feature and the proposed fusion algorithm for unlabeled and test samples in Table 2. As we observe, the proposed fusion algorithm is able to enhance the accuracy compared to the use

Table 3
The average and standard deviation calculated on the MNIST dataset adopting different labeled sample sizes. (Top) Results on the unlabeled samples (i.e., transductive setting). (Bottom) Results on the unseen (test) samples (i.e., inductive setting).

MNIST (Accuracy on Unlabeled data)			
Scheme	*lab./class*		
	20	30	40
VGG16	94.53 ± 0.3	95.17 ± 0.3	95.59 ± 0.3
VGG19	94.35 ± 0.5	94.99 ± 0.5	95.35 ± 0.4
Concatenation	95.87 ± 0.4	96.28 ± 0.4	**96.66 ± 0.3**
SNF (Wang et al., 2014)	94.53 ± 0.6	95.81 ± 0.4	96.49 ± 0.3
SMGI (Karasuyama & Mamitsuka, 2013)	88.40 ± 0.9	89.66 ± 0.7	89.84 ± 0.7
DGFLP (Lin et al., 2017)	92.14 ± 0.5	92.70 ± 0.5	93.06 ± 0.3
MLGC (An et al., 2017)	88.20 ± 1.4	89.45 ± 1.0	90.27 ± 0.7
AMGL (Nie et al., 2016)	94.99 ± 0.5	95.39 ± 0.3	95.67 ± 0.3
MVCGL (ours)	**95.89 ± 0.4**	**96.38 ± 0.4**	**96.66 ± 0.3**

MNIST (Accuracy on Test data)			
Scheme	*lab./class*		
	20	30	40
VGG16	94.98 ± 0.5	95.39 ± 0.3	95.81 ± 0.3
VGG19	95.05 ± 0.5	95.64 ± 0.4	95.97 ± 0.4
Concatenation	95.47 ± 0.5	**96.79 ± 0.5**	96.21 ± 0.4
SNF (Wang et al., 2014)	94.79 ± 0.6	95.97 ± 0.3	96.64 ± 0.2
MVCGL (ours)	**96.48 ± 0.4**	96.75 ± 0.3	**97.05 ± 0.3**

of a single feature which proves that the fusion algorithm can benefit from the complementary information in both views.

In the MNIST database, for each class, we randomly select 1000 samples for the training set and the rest for the test set. We select l samples from the training set as the labeled data and the rest as unlabeled samples. We report the average accuracy along with the standard deviation over 10 different combinations of labeled/unlabeled samples in Table 3. Bold values correspond to the highest accuracy among competing methods. We compared the proposed method with the following algorithms, SNF (Wang et al., 2014), SMGI (Karasuyama & Mamitsuka, 2013), DGFLP (Lin et al., 2017), MLGC (An et al., 2017), AMGL (Nie et al., 2016), and MMCL (Gong, Tao, Maybank, Liu, Kang, & Yang, 2016).

Table 3(a) shows the accuracy on unlabeled samples. The proposed method shows better results in terms of average accuracy and standard deviation. In other words, the average accuracy of the proposed method is higher while its standard deviation is lower compared to the use of a single feature and other competing algorithms. It shows that the proposed method can extract more discriminative information in comparison to other methods and hence obtains better performance.

Moreover, compared with other fusion algorithms, the proposed method had better accuracy. Moreover, for the test samples, we report the accuracy in Table 3(b). It is clear that SMGI, DGFLP, MLGC, and AMGL which cannot predict the label of test samples are not reported. For the proposed method, we calculate the label of the test data using Eq. (10). Similarly, we observe that the proposed method has better accuracy than the single feature. The only algorithm which gains slightly better accuracy is the feature concatenation on test data with only 0.04% when we have 30 labeled samples per class. The obtained accuracy on the test samples is similar to that obtained on the unlabeled samples, which shows that the proposed method can well determine the underlying subspace of data.

4.3. Computational complexity

It is worth noting that the label of an unseen sample can be estimated in two ways. The first is to include it in the training set as an unlabeled sample and repeat the whole training process. The second solution is to find a mapping from the feature space to the label space and thus estimate the label of an unseen sample directly. The algorithms with inductive capability adopt the second solution. The positive of the inductive capability is that one does not need to repeat the training process, but uses a mapping that is usually faster to estimate the label for the unseen samples. In the following, we evaluate the computational complexity of the proposed method in estimating the label of unlabeled and test samples. In the transductive setting, our proposed method estimates both the labels of unlabeled samples and a linear transformation for mapping the feature space to the label space. The method is iterative, so its complexity is linear with the number of iterations (i.e., T). The transductive part of the proposed method has a computational complexity of $O(TN^3)$. The computational complexity of the DGFLP algorithm is $O(T(D^{3.5}C + DN^2))$ (Lin et al., 2017). That of SMGI and MLGC is $O(N^3)$ and that of AMGL is $O(u^3l^2C)$, where D is the dimensionality of the features, C is the number of classes, $N = l+u$ is the total number of training samples, u is the number of unlabeled samples and l is the number of labeled samples.

We note that the proposed method has a higher computational cost than some other methods in the training phase (transductive setting), but has a higher accuracy compared to the competing methods. For the inductive part of the proposed method, which is used to estimate the label of N_{test} test samples, the inference has complexity $O(N_{test}D^2C)$ because this inference requires matrix multiplication $\mathbf{X}^T\mathbf{Q}$, where the size of $\mathbf{X}^T$ is $N_{test} \times D$ and that of $\mathbf{Q}$ is $D \times C$. It should be mentioned that in order to estimate the labeling of test samples in transductive algorithms, one needs to repeat the training phase, while in the proposed method it is obtained by linear projection (matrix multiplication). Moreover, the projection used to estimate the label of test samples can be done in parallel when we have many samples.

5. Conclusion

In this letter, we propose a unified algorithm for constructing a consistent multi-view graph and label propagation that uses smoothness of data and labels in its criteria and combines label propagation and graph estimation into a single criterion. From the application point of view, the proposed method has several advantages. First, it can use few labeled samples, which is beneficial for many applications (e.g., medical diagnosis) where labeled

samples are difficult to obtain. Second, it merges information from multiple views to propagate the labels to unlabeled samples. Third, it estimates a linear transformation to predict the labeling of unseen samples, which can be useful for many practical tasks (e.g., online classification, real-time and large-scale labeling estimation). Fourth, it does not require individual graphs for each view, but constructs a unified graph within its optimization task, which offer a lot of simplicity in real-world tasks. The evaluations performed on several databases show that the proposed multi-view method has higher accuracy than using a single feature and also has better accuracy compared to other fusion techniques. The use of different hand-crafted and deep features shows that the proposed method can work with different types of features.

Although the proposed method can easily estimate the label for a large number of unseen samples, one of the main drawbacks of the proposed algorithm is that it cannot handle large training sizes since it uses the entire training samples in the learning phase. In our future work, we will focus on using anchors as a solution to reduce the computational complexity of the algorithm in the training phase. Moreover, the assumption of a linear mapping between the feature space and the label space is a naive assumption that may not be correct for complicated tasks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was partially funded by the Spanish Ministerio de Ciencia, Innovación y Universidades, Spain, Programa Estatal de I+D+i Orientada a los Retos de la Sociedad, RTI2018-101045-B-C21, and the University of the Basque Country, GIU19/027.

References

Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(12), 2037–2041. http://dx.doi.org/10.1109/TPAMI.2006.244.

An, L., Chen, X., & Yang, S. (2017). Multi-graph feature level fusion for person re-identification. *Neurocomputing*, *259*, 39–45. http://dx.doi.org/10.1016/j.neucom.2016.08.127, Multimodal Media Data Understanding and Analytics.

Angelou, M., Solachidis, V., Vretos, N., & Daras, P. (2019). Graph-based multimodal fusion with metric learning for multimodal classification. *Pattern Recognition*, *95*, 296–307.

Bahrami, S., Bosaghzadeh, A., & Dornaika, F. (2019). Multi similarity metric fusion in graph-based semi-supervised learning. *Computation*, *7*(1), http://dx.doi.org/10.3390/computation7010015.

Bosaghzadeh, A., & Dornaika, F. (2020). Incremental and dynamic graph construction with application to image classification. *Expert Systems with Applications*, *144*, Article 113117. http://dx.doi.org/10.1016/j.eswa.2019.113117.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A Large-scale hierarchical image database. In *CVPR09*.

Dornaika, F., & Bosaghzadeh, A. (2015). Adaptive graph construction using data self-representativeness for pattern classification. *Information Sciences*, *325*, 118–139.

Dornaika, F., Bosaghzadeh, A., & Raducanu, B. (2013). Efficient graph construction for label propagation based multi-observation face recognition. In A. A. Salah, H. Hung, O. Aran, & H. Gunes (Eds.), *Human Behavior Understanding* (pp. 124–135). Cham: Springer International Publishing.

Dornaika, F., Dahbi, R., Bosaghzadeh, A., & Ruichek, Y. (2017). Efficient dynamic graph construction for inductive semi-supervised learning. *Neural Networks*, *94*, 192–203. http://dx.doi.org/10.1016/j.neunet.2017.07.006.

Dornaika, F., Kejani, M. T., & Bosaghzadeh, A. (2017). Graph construction using adaptive Local Hybrid Coding scheme. *Neural Networks*, *95*, 91–101.

Gong, C., Tao, D., Maybank, S. J., Liu, W., Kang, G., & Yang, J. (2016). Multi-modal curriculum learning for semi-supervised image classification. *IEEE Transactions on Image Processing*, *25*(7), 3249–3260. http://dx.doi.org/10.1109/TIP.2016.2563981.

Huang, S., Kang, Z., Tsang, I. W., & Xu, Z. (2019). Auto-weighted multi-view clustering via kernelized graph learning. *Pattern Recognition*, *88*, 174–184. http://dx.doi.org/10.1016/j.patcog.2018.11.007.

Kang, Z., Shi, G., Huang, S., Chen, W., Pu, X., Zhou, J. T., et al. (2020). Multi-graph fusion for multi-view spectral clustering. *Knowledge-Based Systems*, *189*, Article 105102. http://dx.doi.org/10.1016/j.knosys.2019.105102.

Karasuyama, M., & Mamitsuka, H. (2013). Multiple graph label propagation by sparse integration. *IEEE Transactions on Neural Networks and Learning Systems*, *24*(12), 1999–2012. http://dx.doi.org/10.1109/TNNLS.2013.2271327.

LeCun, Y., & Cortes, C. (2010). MNIST handwritten digit database.

Lin, G., Liao, K., Sun, B., Chen, Y., & Zhao, F. (2017). Dynamic graph fusion label propagation for semi-supervised multi-modality classification. *Pattern Recognition*, *68*, 14–23. http://dx.doi.org/10.1016/j.patcog.2017.03.014.

Liu, Z., Lai, Z., Ou, W., Zhang, K., & Zheng, R. (2020). Structured optimal graph based sparse feature extraction for semi-supervised learning. *Signal Processing*, *170*, Article 107456.

Nie, F., Cai, G., & Li, X. (2017). Multi-view clustering and semi-supervised classification with adaptive neighbours. In *Thirty-first AAAI conference on artificial intelligence*.

Nie, F., Li, J., & Li, X. (2016). Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence* (pp. 1881–1887). AAAI Press.

Nie, F., Xu, D., Tsang, I. W., & Zhang, C. (2010). Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction. *IEEE Transactions on Image Processing*, *19*(7), 1921–1932. http://dx.doi.org/10.1109/TIP.2010.2044958.

Phillips, P. J., Moon, H., Rizvi, S., & J. Rauss, P. (2000). The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 1090–1104. http://dx.doi.org/10.1109/34.879790.

Shen, L., & Bai, L. (2006). A review on Gabor wavelets for face recognition. *Pattern Analysis and Applications*, *9*(2–3), 273–292.

Sim, T., Baker, S., & Bsat, M. (2002). The CMU pose, illumination, and expression (PIE) database. In *Proceedings of the 5th IEEE international conference* (pp. 46–51).

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556.

Tong, T., Gray, K., Gao, Q., Chen, L., & Rueckert, D. (2017). Multi-modal classification of Alzheimer's disease using nonlinear graph fusion. *Pattern Recognition*, *63*, 171–181. http://dx.doi.org/10.1016/j.patcog.2016.10.009.

Tuzel, O., Porikli, F., & Meer, P. (2006). Region covariance: A fast descriptor for detection and classification. In A. Leonardis, H. Bischof, & A. Pinz (Eds.), *Computer vision* (pp. 589–600). Berlin, Heidelberg: Springer Berlin Heidelberg.

Wang, B., Mezlini, A. M., Demir, F., Fiume, M., Tu, Z., Brudno, M., et al. (2014). Similarity network fusion for aggregating data types on a genomic scale. *Nature Methods*, *11*, 333–337.

Wang, B., & Tsotsos, J. (2016). Dynamic label propagation for semi-supervised multi-class multi-label classification. *Pattern Recognition*, *52*, 75–84. http://dx.doi.org/10.1016/j.patcog.2015.10.006.

Wang, H., Yang, Y., Liu, B., & Fujita, H. (2019). A study of graph-based system for multi-view clustering. *Knowledge-Based Systems*, *163*, 1009–1019.

Zheng, F., Liu, Z., Chen, Y., An, J., & Zhang, Y. (2020). A novel adaptive multi-view non-negative graph semi-supervised ELM. *IEEE Access*, *8*, 116350–116362.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2004). Learning with local and global consistency. In S. Thrun, L. K. Saul, & B. Schölkopf (Eds.), *Advances in Neural Information Processing Systems 16* (pp. 321–328). MIT Press.

Zhu, X., Ghahramani, Z., & Lafferty, J. D. (2003). Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th international conference on machine learning* (pp. 912–919).