Pattern Recognition Letters 98 (2017) 90–95

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Semi-supervised manifold regularization with adaptive graph construction

Yunyun Wang[a,b,*], Yan Meng[a], Yun Li[a,b], Songcan Chen[c], Zhenyong Fu[a], Hui Xue[d]

[a] Department of Computer Science and Engineering, Nanjing University of Posts & Telecommunications, Nanjing 210046, P.R. China
[b] Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing, Jiangsu 210023, China
[c] Department of Computer Science and Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, P.R. China
[d] School of Computer Science and Engineering, Southeast University, Nanjing 210096, P.R. China

ARTICLE INFO

Article history:
Received 8 September 2016
Available online 5 September 2017

Keywords:
Semi-supervised classification
Manifold regularization
Manifold graph
Graph construction

ABSTRACT

Manifold regularization (MR) provides a powerful framework for semi-supervised classification (SSC) learning. It imposes the smoothness constraint over a constructed manifold graph, and its performance largely depends on such graph. However, 1) The manifold graph is usually pre-constructed before classification, and fixed during the classification learning process. As a result, independent with the subsequent classification, the graph does not necessarily benefit the classification performance. 2) There are parameters needing tuning in the graph construction, while parameter selection in semi-supervised learning is still an open problem currently, which sets up another barrier for constructing a "well-performing" manifold graph benefiting the performance. To address those issues, we develop a novel semi-supervised manifold regularization with adaptive graph (AGMR for short) in this paper by integrating the graph construction and classification learning into a unified framework. In this way, the manifold graph along with its parameters will be optimized in learning rather than pre-defined, consequently, it will be adaptive to the classification, and benefit the performance. Further, by adopting the entropy and sparse constraints respectively for the graph weights, we derive two specific methods called AGMR_entropy and AGMR_sparse, respectively. Our empirical results show the competitiveness of those AGMRs compared to MR and some of its variants.

1. Introduction

In many real applications, unlabelled data can be easily and cheaply collected, while the acquisition of labelled data is usually quite expensive and time-consuming, especially involving manual effort, e.g., in web page recommendation and spam email detection. Consequently, semi-supervised classification, which exploits a large amount of unlabelled data jointly with the limited labelled data for classification learning, has attracted intensive attention during the past decades [7,25,26,28].

Generally, semi-supervised classification methods attempt to exploit the intrinsic data distribution information disclosed by the unlabeled data in learning. To exploit the unlabeled data, some assumption should be adopted for learning. Two common assumptions in semi-supervised classification are the cluster assumption and the manifold assumption [7,19,26]. The former assumes that similar instances are likely to share the same class label, thus guides the classification boundary passing through the low density region between clusters. The latter assumes that the data are resided on some low dimensional manifold represented by a Laplacian graph, and similar instances should share similar classification outputs according to the graph. Almost all off-the-shelf semi-supervised classification methods adopt one or both of those assumptions explicitly or implicitly [7,25]. For instance, the large margin semi-supervised classification methods, such as Transductive Support Vector Machine (TSVM) [15], semi-supervised SVM (S3VM) [11] and their variants [8,17], adopt the cluster assumption. The graph-based semi-supervised classification methods, such as label propagation [4,27], graph cuts [5] and manifold regularization (MR) [3], adopt the manifold assumption.

The graph-based semi-supervised classification methods are mainly transductive ones, except MR. Although transductive methods have specific applications, many real tasks need predicting unseen instances, thus need inductive methods. As a result, as an inductive graph-based semi-supervised classification method, MR has attracted much attention and applied in many learning tasks such as image retrieval [14] and web spam identification [1], etc. In this paper, we will concentrate on the MR framework [3].

* Corresponding author.
E-mail address: wangyunyun@njupt.edu.cn (Y. Wang).

http://dx.doi.org/10.1016/j.patrec.2017.09.004
0167-8655/

The learning process of MR includes two steps: First, a manifold graph is constructed to describe the manifold structure of instances, in which the graph nodes represent instances, and the weights represent the similarities between instances. Then, according to the manifold assumption, the smoothness constraint over the constructed graph is implemented in terms of regularization. The construction of manifold graph is critical for the performance of MR. Once a "well-performing" graph benefiting the subsequent classification is constructed, it can finally help boost the classification performance. Otherwise, it will not help the classification, or even hurt the performance. However, on one hand, the graph is usually defined in advance and kept fixed during the learning process. It is actually impossible for us to judge whether a graph is a "well-performing" graph in advance. As a result, it is really difficult to construct a "well-performing" graph before classification. On the other hand, there are parameters needing tuning in the manifold graph, whereas in semi-supervised learning with limited label information, the parameter selection is still an open problem with no effective solution yet. It sets up another barrier for graph constructing for MR in advance. As far as we know, the existing improvements of MR either attempt to select the regularization parameters [12], or try to improve the efficiency of MR [23,21], few researches have concentrated on graph construction up to now. Actually, the graph learning issue is considered as a separate topic under research currently, although the adaptive graph construction has been studied in GoLPP [24] for dimension reduction, MR and its improvements mainly adopt a pre-constructed graph.

To address the above two issues, we aim to develop a new MR framework for semi-supervised classification here by introducing an adaptive graph (AGMR for short). In AGMR, the construction of manifold graph is incorporated into the classification learning. In this way, the manifold graph along with its parameters can be automatically adjusted in learning rather than specified in advance. The graph construction and classification learning are combined together, thus can be more likely to benefit each other. Further, by adopting the entropy and sparse constraints for the graph weights, respectively, we derive two specific methods called AGMR_entropy and AGMR_sparse, respectively. The implementation follows an alternating iterative strategy to optimize the decision function and the manifold graph, respectively. Each step in the iteration results in a closed-form solution, and its iterative convergence can theoretically be guaranteed. Experiments on several real datasets show the competitive performance of AGMR compared with MR and its improvements with different graph constructed.

The rest of this paper is organized as follows. Section 2 introduces the related works, Section 3 presents the proposed graph-adaptive MR framework, Section 4 gives the empirical results, and some conclusions are drawn in Section 5.

2. Related works

Given labeled data $X_l = \{x_i\}_{i=1}^{l}$ with the corresponding labels $Y = \{y_i\}_{i=1}^{l}$, and unlabeled data $X_u = \{x_j\}_{j=l+1}^{n}$, where each $x_i \in R^d$ and $u = n\text{-}l$. $G = \{w_{ij}\}_{i,j=1}^{n}$ is a pre-specified Laplacian graph over the whole dataset, where each weight w_{ij} represents the similarity between the connected instances x_i and x_j. There are two ways for deciding whether x_i and x_j are connected. One is the k-nearest neighbor strategy, i.e., x_i and x_j are connected if x_i is in the k-nearest neighbor of x_j (or x_j is in the k-nearest neighbor of x_i). The other is the ε-ball nearest neighbor strategy, i.e., x_i and x_j are connected when $\|x_i - x_j\|^2 < \varepsilon$. The weights over the graph describe the similarities between the connected instances, and can be specified by several weighting strategies. For example, the 0–1 weighting, i.e., $w_{ij} = 1$ if x_i and x_j are connected by an edge over the graph, the heat kernel weighing with $w_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma}}$ if x_i and x_j are connected, or the dot-product weighting with $w_{ij} = x_i^T x_j$ if x_i and x_j are connected.

After the construction of the manifold graph, the framework of MR can be formulated as follows with a decision function $f(x)$,

$$\min_f \sum_{i,j=1}^{l+u} w_{ij}(f(x_i) - f(x_j))^2 + \gamma_1 \sum_{i=1}^{l} V(x_i, y_i, f) + \gamma_2 \|f\|_K^2 \tag{1}$$

where $V(x_i, y_i, f)$ is the loss function, such as the hinge loss max $\{0, 1\text{-}y_i f(x_i)\}$ for support vector machine (SVM), or the square loss $(y_i - f(x_i))^2$ for regularized least square classifier (RLSC), in this way, the MR framework naturally embodies the specific algorithms LapSVM and LapRLSC [3]. $\|f\|_K^2$ is a regularization term for smoothness in the Reproducing Kernel Hilbert Space (RKHS). The third term guarantees the prediction smoothness over the manifold graph, which can be further written as

$$\sum_{i,j=1}^{l+u} w_{ij}(f(x_i) - f(x_j))^2 = 2\mathbf{f}^T\mathbf{L}\mathbf{f} \tag{2}$$

where $\mathbf{f} = [f(x_1), \ldots, f(x_{l+u})]^T$, and $\mathbf{L}$ is the graph Laplacian given by $\mathbf{L} = \mathbf{D}\text{-}\mathbf{W}$, $\mathbf{W}$ is the weight matrix of graph G and $\mathbf{D}$ is a diagonal matrix with the diagonal component given by $\mathbf{D}_{ii} = \sum_{j=1}^{n} w_{ij}$. According to the Representer theorem [3], the minimizer of problem (1) has the form

$$f^*(x) = \sum_{i=1}^{l+u} \alpha_i K(x_i, x) \tag{3}$$

where $K: X \times X \to R$ is a Mercer kernel (the bias of the decision function can be omitted by augmenting each instance with a 1-valued element).

3. Semi-supervised manifold regularization with adaptive graph (AGMR)

3.1. Model formulation

Given labeled instances $X_l = \{x_i\}_{i=1}^{l}$ with the corresponding labels $Y = \{y_i\}_{i=1}^{l}$, and unlabelled instances $X_u = \{x_j\}_{j=l+1}^{n}$, where each $x_i \in R^d$ and $u = n\text{-}l$. The optimization problem of AGMR can be formulated as

$$\begin{aligned} &\min_{f, w_{ij}} \sum_{i,j=1}^{l+u} w_{ij}(f(x_i) - f(x_j))^2 \\ &+\gamma_1 \sum_{i=1}^{l} (f(x_i) - y_i)^2 + \gamma_2 \|f\|_K^2 + \eta R(w_{ij}) \\ &s.t. \sum_{j=1}^{u} w_{ij} = 1 \\ &w_{ij} \geq 0 \end{aligned} \tag{4}$$

The first three terms in the optimization function of (4) are the same with those in MR, $R(w_{ij})$ is some constraint on the graph weights, and η is the regularization parameter. Different from MR seeking the decision function in learning, AGMR seeks both the decision function and the weights for the manifold graph. From the optimization problem in (4), we can find that: 1) The manifold graph in MR is specified before classification, and fixed in the learning process. While in AGMR, the graph is actually optimized in the learning process along with its parameters; 2) In AGMR, we have $\sum_{j=1}^{u} w_{ij} = 1$ and $w_{ij} \geq 0$, in this way, each w_{ij} actually reflects the probability that x_i and x_j should be in the same class; 3) Without the constraint $R(w_{ij})$ on each w_{ij}, the solution for each w_{ij} will degenerate to a trivial one, in which only one element is 1, and the remainder are all 0.

Different constraints for the graph weights generate different models, thus yield different classification performance. In the following, we will respectively use the entropy constraint and the sparse constraint for examples to develop new AGMR methods within the above framework.