Neurocomputing 129 (2014) 265–272

Contents lists available at ScienceDirect

Neurocomputing

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

Sparse semi-supervised learning on low-rank kernel

Kai Zhang [a,*], Qiaojun Wang [b], Liang Lan [c], Yu Sun [d], Ivan Marsic [b]

[a] NEC Laboratories America, Inc., 4 Independence Way, Princeton, NJ, United States
[b] Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ, United States
[c] Huawei Noah's Ark Laboratory, Hong Kong
[d] Siemens Corporate Research, 755 College Road East, Princeton, NJ, United States

ARTICLE INFO

Article history:
Received 21 April 2013
Received in revised form
8 September 2013
Accepted 9 September 2013
Communicated by Shiliang Sun
Available online 23 October 2013

Keywords:
Semi-supervised learning
Regularized least squares
Manifold regularization
Graph Laplacian
Sparse regression
Low-rank approximation

ABSTRACT

Advances of modern science and engineering lead to unprecedented amount of data for information processing. Of particular interest is the semi-supervised learning, where very few training samples are available among large volumes of unlabeled data. Graph-based algorithms using Laplacian regularization have achieved state-of-the-art performance, but can induce huge memory and computational costs. In this paper, we introduce L_1-norm penalization on the low-rank factorized kernel for efficient, globally optimal model selection in graph-based semi-supervised learning. An important novelty is that our formulation can be transformed to a standard LASSO regression. On one hand, this makes it possible to employ advanced sparse solvers to handle large scale problems; on the other hand, a globally optimal subset of basis can be chosen adaptively given desired strength of penalizing model complexity, in contrast to some current endeavors that pre-determine the basis without coupling it with the learning task. Our algorithm performs competitively with state-of-the-art algorithms on a variety of benchmark data sets. In particular, it is orders of magnitude faster than exact algorithms and achieves a good trade-off between accuracy and scalability.

1. Introduction

Advances of modern science and engineering in various domains have created unprecedented amount of data for information processing. Of particular interest is the semi-supervised learning scenario, where very few training labels are available due to the high cost of human interventions. How to utilize unlabeled data together with a small amount of labeled examples to boost learning performance while guaranteeing the algorithm efficiency has been a continued research interest. Enormous efforts have been devoted to semi-supervised learning, including transductive SVM [6,13], cotraining [3], label propagation [34], graph-based methods [1,20,36], semi-supervised kernel learning [4,7,15]. See a detailed survey in [35].

In this paper, we focus on a graph-based algorithm for semi-supervised learning. Assume that we use a positive semi-definite (PSD) kernel function $\kappa(\cdot,\cdot)$, and the $n\times n$ kernel/similarity matrix K such that $K_{ij}=\kappa(\mathbf{x}_i,\mathbf{x}_j)$. Define the graph Laplacian matrix as $\mathcal{L}=D-K$, where $D\in\mathbb{R}^{n\times n}$ is a (diagonal) degree matrix such that $D_{ii}=\sum_{j=1}^{n}K_{ij}$. The normalized graph Laplacian is defined as $\tilde{L}=\mathbf{I}-D^{-1/2}KD^{-1/2}$, where $\mathbf{I}$ is the identity matrix of proper size. The (normalized) graph Laplacian matrix imposes important smoothness constraints. To see this, suppose a prediction function $f(\cdot)$ is evaluated on $\{\mathbf{x}_i\}_{i=1}^{n}$, and the prediction is represented as $\mathbf{f}\in\mathbb{R}^{n\times 1}$ where $f_i=f(\mathbf{x}_i)$. Then the smoothness of $\mathbf{f}$ with regard to the graph is given by [1,37]

$$\sum_{i,j=1}^{n}\left(\frac{f_i}{\sqrt{D_{ii}}}-\frac{f_j}{\sqrt{D_{jj}}}\right)^2K_{ij}=\mathbf{f}'\tilde{L}\mathbf{f},$$

whose minimization is called the Laplacian regularization. It enforces a geometric, data-dependent constraint that the prediction should be sufficiently smooth with regard to the manifold structure of the data. Suppose that we are given a set of labeled data $\{\mathbf{x}_i\}_{i=1}^{l}$ and a large amount of unlabeled data $\{\mathbf{x}_i\}_{i=l+1}^{n}$, where $u=n-l$. By using a loss function $V(y,f(\mathbf{x}))$, Laplacian regularized semi-supervised learning can be formulated as [1]

$$\min_{f}\sum_{i=1}^{l}V(y_i,f(\mathbf{x}_i))+\gamma_A\|f\|_K^2+\gamma_I\frac{1}{n^2}\mathbf{f}'L\mathbf{f} \tag{1}$$

here $\|f\|_K$ is the Reproducing Kernel Hilbert Space (RKHS) norm of the prediction function, γ_A is the associated regularization parameter, and γ_I is the regularization parameter for the Laplacian smoothness term. The minimizer of this optimization problem admits the expansion form:

$$f^*(\mathbf{x})=\sum_{i=1}^{l+u}\boldsymbol{\alpha}_iK(\mathbf{x}_i,\mathbf{x}), \tag{2}$$

* Corresponding author.
E-mail address: kzhang@nec-labs.com (K. Zhang).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.09.033

where α_i's are the kernel expansion coefficients. Eq. (2) is called the representer theorem [1].

The Laplacian regularization has been proven as an effective way for semi-supervised learning [1,36]. One practical concern, however, is the need to manipulate the $n \times n$ kernel matrix which is the computational bottleneck. On the other hand, as the representer theorem (2) shows, the decision function is potentially spanned by all the labeled and unlabeled samples, leading to a dense model and slow testing.

Various attempts have been made to alleviate the computational cost of graph-based semi-supervised learning. One direction is to use low-rank approximation to scale up the optimization [10–12,26]. These algorithms are typically transductive, and the low-rank approximation does not consider label information which can be otherwise beneficial. Another direction is to span the model by only a small set of basis vectors [18,30], which will lead to fast training and testing. However, the selection of the basis is independent of the learning task. Also the training time scales quadratically with the model size, which is less efficient if a complex model is needed for difficult tasks.

Recently, the L_1-regularized linear regression, also known as the LASSO [24], has drawn considerable interest. It achieves simultaneous prediction and globally optimal model selection via penalizing the L_1-norm of the model coefficients. Inspired by it, we apply the L_1-norm penalization on the expansion coefficients of the low-rank factorized kernel in graph-based semi-supervised learning. To the best of our knowledge, applying the L_1-penalization with the low-rank kernel decomposition for semi-supervised learning is new. Our formulation not only ensures effective manifold regularization but also enjoys the globally optimal model selection. We also propose an efficient solution by approximately transforming our formulation to a standard LASSO, which is quite scalable to large data. Our algorithm competes favorably with exact, state-of-the-art algorithms such as Laplacian-RLS [1] and local and global consistency [34], while at the same time being orders of magnitudes faster. Compared with several fast semi-supervised learning algorithms [8,12,30], the accuracy of our algorithm is quite promising, though only a few times slower. Overall, our algorithm achieves a good trade-off between accuracy and scalability.

The rest of the paper is organized as follows. Section 2 introduces the proposed algorithm. In Section 3, we discuss related algorithms. Section 4 reports experimental results. The last section concludes the paper.

2. Proposed method

This section details our algorithm. First, we propose the mathematical formulation, i.e., L_1-penalization on low-rank kernel expansion in Laplacian-Regularized Least-Squares (Section 2.1). The resultant, sparse QP problem can be expensive to compute. So we propose to apply the Nyström low-rank approximation to the kernel matrix (Section 2.2), which allows us to transform the sparse QP to a standard LASSO (Section 2.3) that can be solved very efficiently.

2.1. L_1-penalization of Laplacian-regularized least squares

Given a set of training data $\{\mathbf{x}_i\}_{i=1}^{l}$ and unlabeled data $\{\mathbf{x}_i\}_{i=l+1}^{n}$, we can obtain kernel matrix K, degree matrix D, the graph Laplacian L and normalized graph Laplacian $\tilde{L}$ as in the previous section. For notational simplicity, we define $K_l \in \mathbb{R}^{l\times n}$ as the rows in the kernel matrix corresponding to the labeled samples. Note that this can also be written as $K_l = e_l K$ where $e_l = [\mathbf{I}_{l\times l}\ \mathbf{0}_{l\times u}]$.

By using (2), we assume that the classifier is spanned by all the labeled and unlabeled samples, i.e., $\mathbf{f} = K\boldsymbol{\alpha}$, where $\boldsymbol{\alpha} \in \mathbb{R}^{n\times 1}$ is the kernel expansion coefficient.[1] We also require the model coefficients to be reasonably sparse considering the training and testing speed. Therefore we use an L_1-norm penalization on the model coefficients $\boldsymbol{\alpha}$ to control the model complexity. On the other hand, we require that the estimated labels, $K\alpha$, to be smooth with regard to the graph structure of the data similar to (1), and skipped the term $\|f\|_K$ for simplicity. Then we have the following problem:

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{n\times 1}} \lambda_1 \|K_l\boldsymbol{\alpha} - \mathbf{y}_0\|^2 + (K\boldsymbol{\alpha})' L (K\boldsymbol{\alpha}) + \lambda_2 |\boldsymbol{\alpha}|_1 \tag{3}$$

here $\mathbf{y}_0 \in \mathbb{R}^{l\times 1}$ is the class labels for the labeled samples. The first term is a loss function that measures the discrepancy between the true and estimated labels on the labeled samples ($\mathbf{x}_l$). The second term enforces the smoothness constraint of $K\boldsymbol{\alpha}$. The third term $|\alpha|_1 = \sum_i |\alpha_i|$ is a regularization term, which encourages zero entries in the model coefficients α, thereby improving the efficiency of the testing phase.

The objective function (3) can be written equivalently as

$$\begin{aligned} &\min_{\boldsymbol{\alpha}\in\mathbb{R}^{n\times 1}} \quad \boldsymbol{\alpha}' Q \boldsymbol{\alpha} - 2c'\boldsymbol{\alpha} + \lambda_2 |\boldsymbol{\alpha}|_1 \\ &\text{where} \quad Q = K'LK + \lambda_1 K' e_l' e_l K \\ &c = K_l' \mathbf{y}_0 \end{aligned} \tag{4}$$

Formulation (4) is a quadratic programming problem with a sparsity constraint, which can be solved in different ways. For example, it can be re-formulated as a standard QP by decomposing α_i's as the difference of two non-negative terms a_i and b_i, which gives a standard QP with $2n$ variables with $2n$ non-negative constraints, and typically has a polynomial time complexity which is expensive for large n. Another possibility is to resort to an existing optimization technique like the alternating direction method [29]. This can provide exact optimal solution for (4).

In this paper, we are interested in obtaining an approximate, computationally more efficient solution of (4). An interesting observation here is that by fully exploiting the low-rank structure of the kernel matrix, problem (4) can be transformed to a standard LASSO regression, which can then be solved extremely efficiently due to the availability of various sparse solvers.

2.2. Low-rank approximation in semi-supervised setting

Low-rank matrix approximation is a useful tool for handling large matrices and reducing the dimensionality. Besides, it also has applications in dynamic systems [9,21]. In this paper, we are interested in the low-rank approximation of symmetric, positive semi-definite matrices (such as the kernel matrix) $K \in \mathbb{R}^{n\times n}$ in the form of

$$K \approx GG', \quad G \in \mathbb{R}^{n\times m}, \ m \ll n \tag{5}$$

here GG' is called the rank-m approximation of K. It has been found that in many learning problems, the kernel matrix typically has a fast decaying spectrum [27], justifying the use of the low-rank approximation technique in reducing the memory and computational cost. The optimal rank-m approximation is provided by the eigenvalue decomposition, which can be very expensive. So we will resort to a popular, sampling-based method called the Nyström method [11,10,26,31]. For an $n \times n$ kernel matrix, the Nyström method chooses a subset of m rows/columns $K_{nm} \in \mathbb{R}^{n\times m}$, compute an $m \times m$ eigenvalue decomposition on the intersection of selected rows and columns $K_{mm} \in \mathbb{R}^{m\times m}$, and then approximate

[1] In the case of multiple c classes, $\boldsymbol{\alpha} \in \mathbb{R}^{n\times c}$, which is a simple extension of the binary class formulation.