Discriminative Semi-Supervised Feature Selection Via Manifold Regularization

Zenglin Xu, *Member, IEEE,* Irwin King, *Senior Member, IEEE,* Michael Rung-Tsong Lyu, *Fellow, IEEE,* and Rong Jin

***Abstract*—Feature selection has attracted a huge amount of interest in both research and application communities of data mining. We consider the problem of semi-supervised feature selection, where we are given a small amount of labeled examples and a large amount of unlabeled examples. Since a small number of labeled samples are usually insufficient for identifying the relevant features, the critical problem arising from semi-supervised feature selection is how to take advantage of the information underneath the unlabeled data. To address this problem, we propose a novel discriminative semi-supervised feature selection method based on the idea of manifold regularization. The proposed approach selects features through maximizing the classification margin between different classes and simultaneously exploiting the geometry of the probability distribution that generates both labeled and unlabeled data. In comparison with previous semi-supervised feature selection algorithms, our proposed semi-supervised feature selection method is an embedded feature selection method and is able to find more discriminative features. We formulate the proposed feature selection method into a convex-concave optimization problem, where the saddle point corresponds to the optimal solution. To find the optimal solution, the level method, a fairly recent optimization method, is employed. We also present a theoretic proof of the convergence rate for the application of the level method to our problem. Empirical evaluation on several benchmark data sets demonstrates the effectiveness of the proposed semi-supervised feature selection method.**

***Index Terms*—Feature selection, level method, manifold regularization, multiple kernel learning, semi-supervised learning.**

I. Introduction

With the development of information technology, abundant features are produced to describe larger and more complex tasks, evolving in text processing, computer vision, bioinformatics, sensor networks, and so on. Extracting relevant information from a potentially overwhelming quantity of data becomes more and more important. Moreover, the abundance of features requires high computational ability and storage capability. Feature selection, which is known as a process of selecting relevant features and reducing dimensionality, has been playing an important role in both research and application [7], [21], [26]. It has been employed in a variety of real-world applications, such as natural language processing, image processing, and bioinformatics, where high dimensionality of data is usually observed. It is also used in distributed communication systems and sensor networks, where each mobile equipment or sensor has very limited computational power. Overall, feature selection is a very important method that is often applied to reduce the computational cost or to save storage space, for problems with either high dimensionality or limited resources.

Feature selection can be conducted in a supervised or unsupervised manner, in terms of whether the label information is utilized to guide the selection of relevant features [36]. Generally, supervised feature selection methods require a large amount of labeled training data. It, however, could fail to identify the relevant features that are discriminative to different classes, provided the number of labeled samples is small. On the other hand, while unsupervised feature selection methods could work well with unlabeled training data, they ignore the label information and therefore are often unable to identify the discriminative features. Given the high cost in manually labeling data, and at the same time abundant unlabeled data is often easily accessible, it is desirable to develop feature selection methods that are capable of exploiting both labeled and unlabeled data. This motivates us to introduce semi-supervised learning [9], [68] into the feature selection process.

Semi-supervised learning approaches can be roughly categorized into two major groups. The first group is based on the clustering assumption that most data examples, including both the labeled ones and the unlabeled ones, should be far away from the decision boundary of the target classes. The representative approaches in this category include tranductive support vector machine (SVM) and semi-supervised SVM [11], [12], [27], [54], [59], [60]. The second group is based on the manifold assumption that most data examples lie on a low-dimensional manifold in the input space. The well-known algorithms in this category include label propagation [67], harmonic function [69], graph cuts [6], spectral graph trans-

Manuscript received May 1, 2009; revised November 26, 2009 and February 23, 2010; accepted February 23, 2010. Date of publication June 21, 2010; date of current version July 8, 2010. This work was supported by the National Science Foundation under Grant IIS-0643494, the National Institute of Health under Grant 1R01GM079688-01, and the Research Grants Council of Hong Kong under Grants CUHK4158/08E and CUHK4128/08E. This work is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-Centric Computing and Interface Technologies.

Z. Xu is with Cluster of Excellence, Saarland University, Max Planck Institute for Informatics, Saarbruecken 66123, Germany (e-mail: zlxu@mpi-inf.mpg.de).

I. King and M. R. Lyu are with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Shatin, Hong Kong (e-mail: king@cse.cuhk.edu.hk; lyu@cse.cuhk.edu.hk).

R. Jin is with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48823 USA (e-mail: rongjin@cse.msu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2010.2047114

ducer [28], and manifold regularization [3]. A comprehensive study of semi-supervised learning techniques can be found in [9], [68]. Among these semi-supervised learning algorithms, the method of semi-supervised SVM with manifold regularization has demonstrated good performance [3], [45]. In this paper, we try to employ the idea of manifold regularization to semi-supervised feature selection.

Semi-supervised feature selection studies how to better identify the relevant features that are discriminative to different classes by effectively exploring the information underlying the huge amount of unlabeled data. In [65], a filter-based semi-supervised feature selection method was proposed, which ranks features via some information measure. As argued in [21], the filter-based feature selection could discard important features that are less informative by themselves but are informative when combined with other features. Moreover, it can also ignore the underlying learning algorithm that is used to train classifiers from labeled data. Therefore, it is hard to find features that are particularly useful to a given learning algorithm.

To avoid these disadvantages, we discuss in detail a novel semi-supervised feature selection method based on the idea of manifold regularization [57]. In the proposed method, an optimal subset of features is identified by maximizing a performance measure that combines classification margin with manifold regularization. Experiments on several benchmark data sets indicate the promising results of the proposed method in comparison with the state-of-the-art approaches for feature selection. We summarize the contributions of this paper in the following.

1) We propose a novel discriminative semi-supervised feature method based on the maximum margin principle and the manifold regularization. The feature selection process is embedded with the semi-supervised classifier, which distinguishes itself from the existing filter-based methods for semi-supervised feature selection methods. The maximum margin principle guarantees the discriminative ability of the selected features. We have theoretically shown that the proposed method is equivalent to the optimization over an mixed norm related to L_2 and L_0 regularization, which ensures the sparsity of selected features.
2) The manifold regularization in the proposed feature selection method assures that the decision function is smooth on the manifold constructed by the selected features of the unlabeled data. This, therefore, better exploits the underlying structural information of the unlabeled data.
3) We successfully formulate the presented semi-supervised feature selection method into a concave-convex problem, where the saddle point corresponds to the optimal solution. We then derive an extended level method [34], [56] for semi-supervised feature selection in order to efficiently find the optimal solution of the concave-convex problem. The proof of the convergence rate is also presented in this paper.

The rest of this paper is organized as follows. In Section II, we review the previous paper on feature selection. In Section III, we derive the discriminative semi-supervised feature selection model. We then employ the level method to solve the optimization problem for semi-supervised feature selection in Section IV. Section V presents the experimental evaluation of the proposed semi-supervised feature selection method on digit images and text data sets, under both of the transductive setting and the semi-supervised setting, followed by the conclusion in Section VI.

II. Related Work

Feature selection has been a fundamental research topic in data mining. The goal of feature selection is to choose from the input data a subset of features that maximizes a generalized performance criterion. Thus, it is different from feature extraction [2], [24], [30], [43], [44], [62], which maps the input data into a reduced representation set of features. Comparing with feature extraction, feature selection keeps the same space as the input data and thus has better interpretability for some specific applications. We focus on the review of recent paper for feature selection.

A number of performance criteria have been proposed for feature selection, including mutual information [19], [31], maximum margin [22], [52], kernel alignment [15], [37], graph-spectrum based measures [53], [66], construction errors in neural network [35], [42], worst case probability [4], [61], and so on. Among them, the maximum-margin-based criterion is probably one of the most widely used criteria for feature selection, due to its outstanding performance.

Generally, supervised feature selection algorithms can be classified into three categories: filters, wrappers, and embedded approaches, according to the degrees of the interaction between the feature selection method and the corresponding classification model [7], [21]. Among these feature selection methods, embedded feature selection methods based on the maximum margin principle have attracted a lot of research focus recently. A typical method is SVM-recursive feature elimination [22] where features with the smallest weights were removed iteratively. In [20], [38], L_1-norm of weights in SVM was suggested to replace L_2-norm for feature selection when learning an SVM model. Another feature selection model related to L_1-norm is lasso [50], which selects features by constraining the L_1-norm of weights. By varying L_1-norm of weights, a unique path of selected features can be obtained. A similar model is least angle regression [18], which can be regarded as unconstrained version of lasso. In addition, several studies [8], [51] explored L_0-norm when computing the weights of features. In [8], the authors proposed feature selection concave method that uses an approximate L_0-norm of the weights. It was improved in [37], [51] via an additional regularizer or a different approximation of L_0-norm. In [58], a non-monotonic feature selection method via direct optimization of feature indicators in the framework of multiple kernel learning can be regarded as a primal-form approximation of L_0-norm. Compared with supervised feature selection, unsupervised feature selection is more challenging in that there is no categorical information available. Indeed, the goal of unsupervised feature selection is to find a small

feature subset that best keeps the intrinsic clusters from data according to the specified clustering criterion [17], [53].

Extended from supervised feature selection and unsupervised feature selection, semi-supervised feature selection works on both the labeled data and the unlabeled data. In [65], the score obtained by combining the spectral and the mutual information is used to rank features in the semi-supervised setting. However, it suffers from both the weak interaction among features, and the weak interaction between the feature selection heuristics and the corresponding classifier. Some other heuristics, including Fisher score [64], forward search [41], and genetic search [23], suffer from the same problem when applied for feature selection. Instead, our proposed semi-supervised feature selection method works in an embedded way: the feature selection process is integrated to the semi-supervised classifier by taking advantage of manifold regularization. This, therefore, takes good care of the correlation among features and the integration between the features and the semi-supervised classifiers. Furthermore, the manifold regularization assists our proposed method to select the subset of features that captures the structural information underneath the unlabeled data.

III. Semi-Supervised Feature Selection Model

In this section, we present the semi-supervised feature selection model that is based on the maximum margin principle and the manifold regularization principle. The former principle guarantees that the selected features have a good discriminative ability, while the latter assures that the decision function is smooth on the manifold constructed from the unlabeled data. Before presenting the semi-supervised feature selection model, we first introduce the notations that will be used throughout this paper.

Let $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathcal{R}^{n\times d}$ denote the entire data set, which consists of n data points in d-dimensional space. The data set includes both the labeled examples and the unlabeled ones. We assume that the first l examples within $\mathbf{X}$ are labeled by $\mathbf{y} = (y_1, y_2, \ldots, y_l)$, where $y_i \in \{-1, +1\}$ represents the binary class label assigned to $\mathbf{x}_i$. For convenience, we also denote the collection of labeled examples by $\mathbf{X}_\ell = (\mathbf{x}_1, \ldots, \mathbf{x}_l)$, and the unlabeled examples by $\mathbf{X}_u$, such that $\mathbf{X} = (\mathbf{X}_\ell, \mathbf{X}_u)$.

The goal of semi-supervised feature selection is to find a set of m relevant features by using both the labeled examples and the unlabeled ones. It is important to note that determining the number of selected features is a model selection problem, which is beyond the scope of this paper. Following [48], we assume that the number of selected features, i.e., m, has been decided by an external oracle. It should also be noted that the number of required features usually is dependent on the objective of the task, and there is no single number of features that are optimal for all tasks.

A. Semi-Supervised SVM Based on Manifold Regularization

Following the framework of manifold regularization [3], a semi-supervised SVM can be obtained by penalizing a regularization term defined as

$$\|\mathbf{f}\|_I^2 = \sum_{i=1}^{n}\sum_{j=1}^{n}(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 W_{ij} = \mathbf{f}^\top \mathcal{L}\mathbf{f}$$

where W_{ij} are the edge weights defined on a pair of nodes $(\mathbf{x}_i, \mathbf{x}_j)$ of the adjacency graph. $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]$ denotes the decision function values over all data examples. The graph Laplacian $\mathcal{L}$ is defined as $\mathcal{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is a diagonal matrix and $D_{ii} = \sum_{j=1}^{n} W_{ij}$. According to [3], $\|\mathbf{f}\|_I^2$ indeed reflects the smoothness of the decision function with respect to the marginal distribution of $\mathbf{X}$.

Considering a linear SVM where the decision function can be represented as $f(\mathbf{x}_i) = \mathbf{w}^\top \mathbf{x}_i - b$, the manifold regularization term $\|\mathbf{f}\|_I^2$ is equal to $\mathbf{w}^\top \mathbf{X}^\top \mathcal{L}\mathbf{X}\mathbf{w}$. Note that the bias term b has no effect on calculating the regularization term. Then, the semi-supervised SVM can be represented as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{l}\xi_i + \frac{\rho}{2}\mathbf{w}^\top \mathbf{X}^\top \mathcal{L}\mathbf{X}\mathbf{w} \\ \text{s.t.} \quad & y_i(\mathbf{w}^\top \mathbf{x}_i - b) \geq 1 - \xi_i \; i = 1, \ldots, l \\ & \xi_i \geq 0 \; i = 1, \ldots, l \end{aligned} \tag{1}$$

where τ denotes the margin error and ρ is a tradeoff parameter between the two regularization terms of $\mathbf{w}$ satisfying $\rho \geq 0$.

In order to efficiently solve the optimization problem (1), we calculate its dual. We, therefore, introduce the following lemma.

Lemma 1: The dual problem of (1) can be written as

$$\begin{aligned} \max_{\alpha} \quad & \alpha^\top \mathbf{e} - \frac{1}{2}(\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell (\mathbf{I} + \rho \mathbf{X}^\top \mathcal{L}\mathbf{X})^{-1} \mathbf{X}_\ell^\top (\alpha \circ \mathbf{y}) \\ \text{s.t.} \quad & \alpha^\top \mathbf{y} = 0 \\ & 0 \leq \alpha \leq C \end{aligned}$$

where $\alpha \in \mathbb{R}^n$ is the dual variable, $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix, and $\circ$ is an operator of the element-wise product.

Lemma 1 can be easily verified using the Lagrange theory.

B. Semi-Supervised Feature Selection Model Based on Manifold Regularization

In the following, we will show how to derive the model of semi-supervised feature selection via manifold regularization.

First, we introduce the indicator variable $\mathbf{p}$, where $\mathbf{p} = (p_1, \ldots, p_d)^\top$ and $p_i \in \{0, 1\}$, $i = 1, \ldots, d$, to represent which features are selected. We further introduce a diagonal matrix $\mathrm{D}(\mathbf{p}) = \mathrm{diag}(p_1, \ldots, p_d)$. Then the input data are now represented as $\mathbf{X}\mathrm{D}(\mathbf{p})$. In order to indicate that m features are selected, we will have $\mathbf{p}^\top \mathbf{e} = m$. We then have Proposition 1 to describe the optimization problem with respect to the feature indicator and the decision function.

Proposition 1: The optimal feature subset for the optimization problem in (1) can be obtained by solving the following

combinatorial problem:

$$\begin{aligned}\min_{\mathbf{w},b,\xi,\mathbf{p}\in\{0,1\}^d} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{l}\xi_i \\ & +\frac{\rho}{2}\mathbf{w}^\top \mathrm{D}(\mathbf{p})\mathbf{X}^\top\mathcal{L}\mathbf{X}\mathrm{D}(\mathbf{p})\mathbf{w} \qquad (2)\\ \text{s.t.} \quad & y_i(\mathbf{w}^\top \mathrm{D}(\mathbf{p})\mathbf{x}_i - b) \geq 1-\xi_i \; i=1,\ldots,l \\ & \xi_i \geq 0 \; i=1,\ldots,l \\ & \mathbf{p}^\top\mathbf{e} = m.\end{aligned}$$

To simplify the presentation, we introduce a matrix $\mathbf{Z}$ as follows:

$$\mathbf{Z} = \mathbf{X}^\top\mathcal{L}\mathbf{X}. \qquad (3)$$

For the convenience of discussion, we assume matrix $\mathbf{Z}$ is non-singular, although the derivation that follows can be easily extended to the singular case by simply replacing matrix inverse with matrix pseudo inverse.

The following proposition reveals that the feature selection approach stated in (2) is equivalent to a mixture of L_2 and L_0 regularization. This, therefore, guarantees the sparsity of the obtained solution.

Proposition 2: The problem in (2) is equivalent to a mixture of L_2 and L_0 regularization, that is

$$\begin{aligned}\min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{l}\xi_i + \frac{\rho}{2}\mathbf{w}^\top\mathbf{Z}\mathbf{w} \qquad (4)\\ \text{s.t.} \quad & y_i(\mathbf{w}^\top\mathbf{x}_i - b) \geq 1-\xi_i \; i=1,\ldots,l \\ & \xi_i \geq 0 \; i=1,\ldots,l \\ & \|\mathbf{w}\|_0 = m.\end{aligned}$$

The equivalence between (2) and (4) can be easily verified by redefining $\mathbf{w}$ as $\mathbf{w}\mathrm{D}(\mathbf{p})$ and replacing constraint $\mathbf{p}^\top e = m$ with constraint $\|\mathbf{w}\|_0 = m$.

The theorem below shows that (2) can be reformulated into a min-max optimization, which is the key for speeding up the computation.

Theorem 1: The problem in (2) is equivalent to the following min-max optimization problem:

$$\min_{\mathbf{p}\in\tilde{P}} \max_{\alpha\in\mathcal{Q}} \phi(\mathbf{p},\alpha) \qquad (5)$$

where

$$\tilde{P} = \{\mathbf{p}\in\{0,1\}^d | \mathbf{p}^\top\mathbf{e} = m\}$$

$$\mathcal{Q} = \{\alpha\in[0,C]^l | \alpha^\top\mathbf{y} = 0\}$$

and $\phi(\mathbf{p},\alpha)$ is defined as

$$\begin{aligned}\phi(\mathbf{p},\alpha) \;=\; & \alpha^\top\mathbf{e} - \frac{1}{2\rho}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell \\ & \left(\mathbf{Z}^{-1} - \left[\mathbf{Z}+\rho\mathbf{Z}\mathrm{D}(\mathbf{p})\mathbf{Z}\right]^{-1}\right)\mathbf{X}_\ell\top(\alpha\circ\mathbf{y}).\end{aligned}$$

When ρ is very small (i.e., $\rho \ll 1$), $\phi(\mathbf{p},\alpha)$ is approximated as

$$\phi(\mathbf{p},\alpha) = \alpha^\top\mathbf{e} - \frac{1}{2}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell\mathrm{D}(\mathbf{p})\mathbf{X}_\ell^\top(\alpha\circ\mathbf{y}).$$

The proof of this theorem can be found in Appendix A. As indicated by the above theorem, when ρ is small, the manifold regularization term can essentially be ignored.

One of the major challenges in solving the optimization problem in (2), or the equivalence in (5), arises from the constraint that $\{p_i\}_{i=1}^d$ have to be binary variables. To avoid the combinatorial nature of the problem, we relax the binary variable $p_i \in \{0,1\}$ to a continuous one, i.e., $p_i \in [0,1]$, and convert the discrete optimization problem in (5) into the following continuous optimization problem:

$$\min_{\mathbf{p}\in\mathcal{P}} \max_{\alpha\in\mathcal{Q}} \phi(\mathbf{p},\alpha) \qquad (6)$$

where the domain $\mathbf{p}$ is modified as

$$\mathcal{P} = \{\mathbf{p}\in[0,1]^d | \mathbf{p}^\top\mathbf{e} = m\}.$$

Theorem 2: The problem in (6) is indeed a convex-concave optimization problem, and therefore its optimal solution is the saddle point of $\phi(\mathbf{p},\alpha)$.

The proof can be found in Appendix B. As indicated by the above theorem, the problem in (6) is essentially a convex problem and therefore its global optimal solution can be found via standard techniques.

Although (6) is a convex-concave optimization problem with a guarantee to find the global optimal solution, solving it efficiently is very challenging. To reduce the computational complexity, in the following proposition, we consider a variant of the min-max optimization problem for (6).

Proposition 3: Equation (6) is equivalent to the following min-max optimization problem:

$$\min_{\mathbf{p}\in\mathcal{P}} \max_{\alpha\in\mathcal{Q}} h(\mathbf{p},\alpha) \qquad (7)$$

where

$$h(\mathbf{p},\alpha) = \alpha^\top\mathbf{e} - \frac{1}{2}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell\Gamma\mathbf{X}_\ell^\top(\alpha\circ\mathbf{y}) \qquad (8)$$

and Γ is defined as

$$\Gamma = \mathrm{D}(\mathbf{p})\left(\mathbf{I}+\rho\mathrm{D}(\mathbf{p})\mathbf{Z}\mathrm{D}(\mathbf{p})\right)^{-1}\mathrm{D}(\mathbf{p}). \qquad (9)$$

The proof of Proposition 3 is similar to the proof of Theorem 1.

As the above optimization problem is hard to optimize due to the existence of the inverse, we then proceed to simplify Γ in $h(\mathbf{p},\alpha)$.

The following proposition provides a simple upper bound for Γ.

Proposition 4: We introduce the matrix $\mathbf{A}$ as

$$\mathbf{A} = (1-\tau)^2\mathrm{D}(\mathbf{p}) + \frac{\tau^2}{\rho}\mathbf{Z}^{-1} \qquad (10)$$

where τ is a trade-off parameter. We have $\mathbf{A} \succeq \Gamma$ for any $\tau\in[0,1]$.

The proof can be found in Appendix C. It is important to point out that the solution of $\mathbf{A}$ formulated in (29) is only one of the possible solutions to bound Γ. Finding the optimal $\mathbf{A}$ is also a challenging problem. However, given $\mathbf{p}$, one can search for an optimal value of τ to make the bound much tighter. As our final goal is to derive an approximated convex optimization

problem and the solution in (29) satisfies our goal, we will not go deeper to examine the optimality of $\mathbf{A}$.

Using the result in Proposition 4, we replace Γ with $\mathbf{A}$, which results in the following optimization problem:

$$\begin{aligned} \min_{\mathbf{p}} \max_{\alpha,\tau} \quad & \alpha^\top \mathbf{e} - \frac{1}{2}(\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell \mathbf{A} \mathbf{X}_\ell^\top (\alpha \circ \mathbf{y}) \\ \text{s.t.} \quad & \alpha^\top \mathbf{y} = 0, 0 \le \alpha \le C \\ & 0 \le \mathbf{p} \le 1 \; \mathbf{p}^\top \mathbf{e} = m \\ & 0 \le \tau \le 1. \end{aligned} \tag{11}$$

Because $\mathbf{A}$ is linear in $\mathbf{p}$, (11) is substantially simpler to solve than (7). In addition, since $\mathbf{A} \succeq \Gamma$, $(\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell \mathbf{A} \mathbf{X}_\ell^\top (\alpha \circ \mathbf{y}) \ge (\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell \Gamma \mathbf{X}_\ell^\top (\alpha \circ \mathbf{y})$ can be obtained. Therefore, for a given τ, the optimization problem in (11) defines a lower bound to the maximization problem in (7). However, it is desirable to bound the gap between the optimal solution and the approximated solution. We leave this for a future paper.

It is interesting to examine (11) with a fixed τ. When $\tau = 0$, the problem in (11) is reduced to a supervised feature selection algorithm. When $\tau = 1$, (11) leads to a semi-supervised classification approach utilizing all features. Now we can use (11) to approximate (7).

C. Optimization Method

Before introducing an optimization method to solve the optimization problem, we first discuss the relationship between the model of semi-supervised feature selection and multiple kernel learning [1], [33], [49], [56]. Note that for a linear kernel, the kernel matrix $\mathbf{K}$ can be written as

$$\mathbf{K} = \mathbf{X}_\ell \mathbf{X}_\ell^\top = \sum_{i=1}^{d} \mathbf{v}_i \mathbf{v}_i^\top = \sum_{i=1}^{d} \mathbf{K}_i$$

where $\mathbf{v}_i$ is the ith feature of $\mathbf{X}_\ell$. The term $\mathbf{K}_i = \mathbf{v}_i \mathbf{v}_i^\top$ can then be regarded as a base kernel which is calculated on a single feature. Therefore, the term $(1-\tau)^2 \mathbf{X}_\ell \mathrm{D}(\mathbf{p}) \mathbf{X}_\ell^\top$ can be written as $(1-\tau)^2 \sum_{i=1}^{d} p_i \mathbf{K}_i$. We further define $\mathbf{H} = \mathbf{X}_\ell (\mathbf{X}^\top \mathcal{L} \mathbf{X})^{-1} \mathbf{X}_\ell^\top$ which can be seen as a kernel matrix defined on the entire data set. The overall optimization problem can be formulated, by switching $\mathbf{p}$ and τ, $\max_{0 \le \tau \le 1} \psi(\tau)$, where $\psi(\tau)$ is defined as

$$\begin{aligned} \min_{\mathbf{p}} \max_{\alpha} \quad & \alpha^\top \mathbf{e} - \frac{1}{2}(\alpha \circ \mathbf{y})^\top \mathbf{M} (\alpha \circ \mathbf{y}) \\ \text{s.t.} \quad & \alpha^\top \mathbf{y} = 0, 0 \le \alpha \le C \\ & \mathbf{p}^\top \mathbf{e} = m \\ & 0 \le p_i \le 1, i = 1, \ldots, d \end{aligned} \tag{12}$$

where

$$\mathbf{M} = (1-\tau)^2 \sum_{i=1}^{d} p_i \mathbf{K}_i + \frac{\tau^2}{\rho} \mathbf{H}. \tag{13}$$

Therefore, the optimization problem in (12) is related to a kernel learning problem. According to [33], the dual problem of $\psi(\tau)$ can be formulated as a minimization problem, that is

$$\begin{aligned} \min_{\mathbf{p},t,\nu,\delta,\theta} \quad & t + 2C\delta^\top \mathbf{e} \\ \text{s.t.} \quad & \begin{pmatrix} \mathbf{M} \circ \mathbf{y}\mathbf{y}^\top & \mathbf{e} + \nu - \delta + \theta \mathbf{y} \\ (\mathbf{e} + \nu - \delta + \theta \mathbf{y})^\top & t \end{pmatrix} \succeq 0 \\ & \nu \ge 0 \; \delta \ge 0, \; \theta \ge 0 \\ & \mathbf{p}^\top \mathbf{e} = m, 0 \le \mathbf{p} \le 1. \end{aligned}$$

For a given τ, the above optimization problem is indeed a semi-definite programming (SDP) problem.

However, the SDP problem involves high computational and storage complexity. Hence, it is hard to be applied in large scale feature selection problems. Instead, we seek to employ efficient optimization techniques to solve the optimization problem in (12). Indeed, (12) can be regarded as a concave-convex problem, since (12) is concave in α and convex in $\mathbf{p}$. The saddle point of (12) corresponds to the optimal solution. According to the literatures of multiple kernel learning and convex optimization, we can formulate an alternating procedure to solve the concave-convex problem: in each step, the solution of α and that of $\mathbf{p}$ are alternatively optimized. More specifically, the procedure can be described in Algorithm 1.

Algorithm 1 A procedure to solve the concave-convex problem

1: Initialize $\mathbf{p}^0 = \frac{m}{d}\mathbf{e}$ and $i = 0$
2: **repeat**
3: Solve the dual of SVM with kernel $\mathbf{M} = (1-\tau)^2 \sum_{i=1}^{d} p_i \mathbf{K}_i + \frac{\tau^2}{\rho}\mathbf{H}$ and obtain optimal solution α^i
4: Solve the minimization problem related to $\mathbf{p}$ to obtain $\mathbf{p}^i$
5: Update $i = i + 1$ and calculate stopping criterion Δ^i
6: **until** $\Delta^i \le \varepsilon$

In Algorithm 1, Δ^i denotes the terminating condition, an example of which is the duality gap. Step 3 deals with the optimization problem related to SVM, where a number of efficient optimization packages could be employed. Step 4 minimizes the problem over $\mathbf{p}$. As the optimization problem (12) is a linear function of $\mathbf{p}$, a smooth optimization technique is needed in order to guarantee the fast convergence of the concave-convex procedure. In this sense, Step 4 plays a very important role for the overall optimization. Efficient methods to solve this step include the cutting plane method [49], the subgradient descent method [40], and the level method [56]. Among them, the level method has shown its significant improvements over the other two methods on the convergence speed [56]. Although some simplification methods of SDP could lead to a quadratic programming (QP) problem [13], they may not apply to our case, due to the constraints on $\mathbf{p}$. In the following, we discuss how to derive an extended level method to solve the concave-convex optimization problem related to semi-supervised feature selection.

IV. Level Method for Semi-Supervised Feature Selection

The level method [34] is from the family of bundle methods, which has recently been employed to efficiently solve

regularized risk minimization problems [47] and multiple kernel learning problems [56]. In this paper, we extend the level method to solve the semi-supervised feature selection problem.

A. Introduction to Level Method

The level method is an iterative approach designed for optimizing a non-smooth objective function. Let $f(x)$ denote the convex objective function to be minimized over a convex domain G.

In the ith iteration, the level method first constructs a lower bound for $f(x)$ by a cutting plane model, denoted by $g^i(x)$. The optimal solution, denoted by $\hat{x}^i$, that minimizes the cutting plane model $g^i(x)$ is then computed. An upper bound $\overline{f}^i$ and a lower bound $\underline{f}_i$ are computed for the optimal value of the target optimization problem based on $\hat{x}^i$.

Next, a level set for the cutting plane model $g^i(x)$ is constructed, denoted by

$$\mathcal{L}^i = \{x \in G : g^i(x) \leq \text{level}^i\}$$
$$\text{level}^i = \lambda \overline{f}^i + (1-\lambda)\underline{f}^i$$

where $\lambda \in (0, 1)$ is a tradeoff constant.

Finally, a new solution x^{i+1} is computed by projecting x^i onto the level set $\mathcal{L}^i$. It is important to note that the projection step, serving a similar purpose to the regularization term in subgradient descent methods, prevents the new solution x^{i+1} from being too far away from the old one x^i.

To demonstrate this point, consider a simple example

$$\min_x \{f(x) = [x]^2 : x \in [-4, 4]\}.$$

Assume $x^0 = -3$ is the initial solution. The cutting plane model at x^0 is

$$g^0(x) = h^0(x) = 9 - 6(x+3).$$

The optimal solution minimizing $g^0(x)$ is $\hat{x}^1 = 4$. If we directly take $\hat{x}^1$ as the new solution, as the cutting plane method does, we will find that it is significantly worse than x^0 in terms of $[x]^2$. The level method alleviates this problem by projecting $x^0 = -3$ to the level set

$$\mathcal{L}^0 = \{x : g^0(x) \leq 0.9[x^0]^2 + 0.1 g^0(\hat{x}^1), -4 \leq x \leq 4\}$$

where $\lambda = 0.9$. It is easy to verify that the projection of x^0 to $\mathcal{L}^0$ is $x^1 = -2.3$, which significantly reduces the objective function $f(x)$ compared with x^0. This is illustrated in Fig. 1. ∇ denotes the lower bound value in one iteration and $\square$ denotes the projected solution.

B. Level Method for Semi-Supervised Feature Selection

We now derive an extended level method for the concave-convex optimization problem (12) that is related to semi-supervised feature selection. To facilitate the description, we denote the objective function of (12) as follows:

$$\varphi(\mathbf{p}, \alpha) = \alpha^\top \mathbf{e} - \frac{1}{2}(\alpha \circ \mathbf{y})^\top \mathbf{M}(\alpha \circ \mathbf{y}). \tag{14}$$

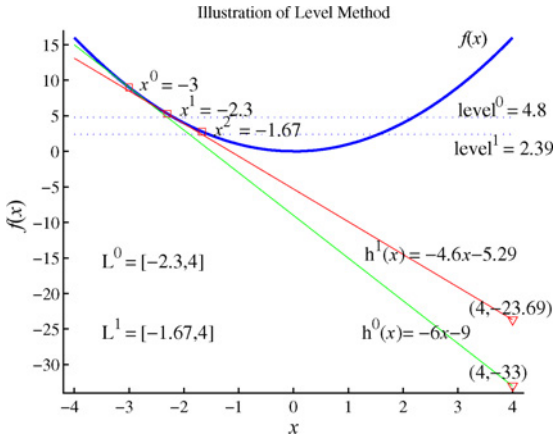


Fig. 1. Illustration of the Level method. We aim to minimize $f(x)$ over $[-4,4]$. With the help of the affine lower bound function $g_i(x)$, we are able to gradually approximate the optimal solution.

The optimization problems related to $\mathbf{p}$ and α are defined as follows:

$$\varphi_\alpha(\mathbf{p}) = \max_{\alpha \in \mathcal{Q}} \varphi(\mathbf{p}, \alpha) \tag{15}$$
$$\varphi_\mathbf{p}(\alpha) = \min_{\mathbf{p} \in \mathcal{P}} \varphi(\mathbf{p}, \alpha). \tag{16}$$

Since $\varphi(\mathbf{p}, \alpha)$ is convex in $\mathbf{p}$ and concave in α, according to the van Neuman lemma, for any optimal solution $(\mathbf{p}^*, \alpha^*)$, we have

$$\begin{aligned} \varphi(\mathbf{p}, \alpha^*) &= \varphi_\alpha(\mathbf{p}) \\ &\geq \varphi(\mathbf{p}^*, \alpha^*) \geq \varphi(\mathbf{p}^*, \alpha) = \varphi_\mathbf{p}(\alpha). \end{aligned}$$

This observation suggests that one can iteratively update both the lower and the upper bounds for $\varphi(\mathbf{p}, \alpha)$ in order to find the saddle point.

To obtain the bounds, we first construct the cutting plane model. Let $\{\mathbf{p}^j\}_{j=1}^i$ denote the solutions for $\mathbf{p}$ obtained in the last i iterations. Let $\alpha^j = \text{argmax}_{\alpha \in \mathcal{Q}} \varphi_{\mathbf{p}^j}(\alpha)$ denote the optimal solution that maximizes $\varphi(\mathbf{p}^j, \alpha)$. We calculate the gradient of $\varphi(\mathbf{p}, \alpha)$ over $\mathbf{p}$ as follows:

$$[\nabla_\mathbf{p} \varphi(\mathbf{p}, \alpha)]_i = -\frac{1}{2}(\alpha \circ \mathbf{y})^\top \mathbf{K}_i (\alpha \circ \mathbf{y}) \; i = 1, \ldots, m.$$

We then construct a cutting plane model $g^i(\mathbf{p})$

$$g^i(\mathbf{p}) = \max_{1 \leq j \leq i} \varphi(\mathbf{p}^j, \alpha^j) + (\mathbf{p} - \mathbf{p}^j)^\top \nabla_\mathbf{p}(\mathbf{p}^j, \alpha^j).$$

As $\varphi(\mathbf{p}, \alpha)$ is linear in $\mathbf{p}$, $g^i(\mathbf{p})$ can be simplified as follows:

$$g^i(\mathbf{p}) = \max_{1 \leq j \leq i} \varphi(\mathbf{p}, \alpha^j). \tag{17}$$

Remark: It is important to note that the cutting plane model in (17) utilizes the historical information of previous steps. As indicated in the illustration, the historical information is helpful to locate the solution. Using only the gradient information of the current iteration may introduce large oscillations. Due to the non-smoothness of the space $\mathcal{P}$, the historical information can help to improve the stability of solutions.

We derive the following proposition for the cutting plane model.

Proposition 5: For any $\mathbf{p} \in \mathcal{P}$, we have

$$g^{i+1}(\mathbf{p}) \geq g^i(\mathbf{p}), \tag{18}$$

$$g^i(\mathbf{p}) \leq \varphi_\alpha(\mathbf{p}). \tag{19}$$

Proposition 5 shows that the cutting plane model in (17) indeed defines a more and more accurate lower bound for the optimal value of $\phi(\mathbf{p}, \alpha)$ as the optimization progresses.

Then we can construct the lower and the upper bounds for the optimal value $\varphi(\mathbf{p}^*, \alpha^*)$. We define two quantities $\underline{\varphi}^i$ and $\overline{\varphi}^i$ as follows:

$$\underline{\varphi}^i = \min_{\mathbf{p}\in\mathcal{P}} g^i(\mathbf{p}) \tag{20}$$

$$\overline{\varphi}^i = \min_{1\leq j\leq i} \varphi(\mathbf{p}^j, \alpha^j). \tag{21}$$

The following theorem shows that $\{\underline{\varphi}^j\}_{j=1}^i$ and $\{\overline{\varphi}^j\}_{j=1}^i$ provide a series of increasingly tight bounds for $\varphi(\mathbf{p}^*, \alpha^*)$.

Theorem 3: We obtain the following properties for $\{\underline{\varphi}^j\}_{j=1}^i$ and $\{\overline{\varphi}^j\}_{j=1}^i$:

$$\underline{\varphi}^i \leq \varphi(\mathbf{p}^*\alpha^*) \leq \overline{\varphi}^i \tag{22}$$

$$\overline{\varphi}^1 \geq \overline{\varphi}^2 \geq \ldots \geq \overline{\varphi}^i \tag{23}$$

$$\underline{\varphi}^1 \leq \underline{\varphi}^2 \leq \ldots \leq \underline{\varphi}^i. \tag{24}$$

Proof: First, since $g^i(\mathbf{p}) \leq \max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}, \alpha)$ for any $\mathbf{p} \in \mathcal{P}$, we have

$$\underline{\varphi}^i = \min_{\mathbf{p}\in\mathcal{P}} g^i(\mathbf{p}) \leq \min_{\mathbf{p}\in\mathcal{P}} \max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}, \alpha).$$

Second, since $\varphi(\mathbf{p}^j, \alpha^j) = \max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}^j, \alpha)$, we have

$$\begin{aligned}
\overline{\varphi}^i &= \min_{1\leq j\leq i} \varphi(\mathbf{p}^j, \alpha^j) \\
&= \min_{\mathbf{p}\in\{\mathbf{p}_1,\ldots,\mathbf{p}_i\}} \max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}, \alpha) \\
&\geq \min_{\mathbf{p}\in\mathcal{P}} \max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}, \alpha) \\
&= \varphi(\mathbf{p}^*, \alpha^*).
\end{aligned}$$

Combining the above results, we obtain (22) in the theorem. It is easy to verify (23) and (24). ■

Theorem 3 shows that the series of lower bound values $(\underline{\varphi}^j)_{j=1}^i$ is non-decreasing and the series of upper bound values $(\overline{\varphi}^j)_{j=1}^i$ is non-increasing. Therefore, we can use these two quantities to bound the optimal value $\varphi(\mathbf{p}^*, \alpha^*)$.

We furthermore define the gap Δ^i as

$$\Delta^i = \overline{\varphi}^i - \underline{\varphi}^i. \tag{25}$$

The following corollary indicates that the gap Δ^i can be used to measure the sub-optimality for solution $\mathbf{p}^i$ and α^i.

Corollary 1: We have the following properties for Δ^i:

$$\begin{aligned}
&\Delta^j \geq 0 \, j = 1, \ldots, i \\
&\Delta^1 \geq \Delta^2 \geq \ldots \geq \Delta^i \\
&|\varphi(\mathbf{p}^j, \alpha^j) - \varphi(\mathbf{p}^*, \alpha^*)| \leq \Delta^i.
\end{aligned}$$

It is easy to verify these three properties of Δ^i in the above corollary using the results of Theorem 3. Corollary 1 shows that the difference of the current objective value to the optimal value is always bounded by the non-increasing gap Δ_i.

Algorithm 2 The level method for semi-supervised feature selection

1: Initialize $\mathbf{p}^0 = \frac{m}{d}\mathbf{e}$ and $i = 0$
2: **repeat**
3: Solve the dual problem of SVM with $\mathbf{M} = (1-\tau)^2\mathbf{X}_\ell \mathrm{D}(\mathbf{p}^i)\mathbf{X}_\ell^\top + \frac{\tau^2}{\rho}\mathbf{H}$ to obtain the optimal solution α^i
4: Construct the cutting plane model $g^i(\mathbf{p})$ in (17)
5: Calculate the lower bound $\underline{\varphi}^i$ and the upper bound $\overline{\varphi}^i$ in (21), and the gap Δ^i in (25)
6: Compute the projection of $\mathbf{p}^i$ onto the level set $\mathcal{L}^i$ by solving the optimization problem in (27)
7: Update $i = i + 1$
8: **until** $\Delta^i \leq \varepsilon$

In the third step, we define the current level as

$$\ell^i = \lambda\overline{\varphi}^i + (1-\lambda)\underline{\varphi}^i.$$

We then construct the level set $\mathcal{L}^i$ using the estimated bounds $\overline{\varphi}^i$ and $\underline{\varphi}^i$ as follows:

$$\mathcal{L}^i = \{\mathbf{p} \in \mathcal{P} : g^i(\mathbf{p}) \leq \ell^i\} \tag{26}$$

where $\lambda \in (0, 1)$ is a predefined constant. The new solution, denoted by $\mathbf{p}^{i+1}$, is computed as the projection of $\mathbf{p}^i$ onto the level set $\mathcal{L}^i$, which is equivalent to solving the following optimization problem:

$$\begin{aligned}
\min_{\mathbf{p}\in\mathcal{P}} \quad & \|\mathbf{p} - \mathbf{p}^i\|_2^2 \\
\text{s.t.} \quad & \varphi(\mathbf{p}, \alpha^j) \leq \ell^i \, j = 1, \ldots, i.
\end{aligned} \tag{27}$$

By means of the projection, we on the one hand aim to ensure that $\mathbf{p}^{i+1}$ is not very far away from $\mathbf{p}^i$, and on the other hand ensure that a significant progress is made when the solution is updated from $\mathbf{p}^i$ to $\mathbf{p}^{i+1}$.

As stated in [56], although the projection is regarded as a QP problem, it can often be solved efficiently because its solution is likely to be the projection onto one of the hyperplanes of polyhedron $\mathcal{L}^i$. In other words, only a few number of linear constraints of $\mathcal{L}$ are active, while most of others are inactive. This sparsity nature usually leads to significant speedup of QP, similar to the solver of SVM [39]. Moreover, with the optimization progresses, especially when it is near to the convergence, there are less changes the values of $\mathbf{p}$. Using warm-start techniques, we can obtain the new solution very efficiently.

We summarize the steps of the extended level method for semi-supervised feature selection in Algorithm 2.

Finally, we discuss the convergence behavior of the level method. In general, convergence is guaranteed because the gap Δ^i, which bounds the absolute difference between $\varphi(\mathbf{p}^*, \alpha^*)$ and $\varphi(\mathbf{p}^i, \alpha^i)$, monotonically decreases through iterations. Based on [56], the following theorem shows the convergence rate of the level method when applied to semi-supervised feature selection.

Theorem 4: To obtain a solution $\mathbf{p}$ that satisfies the stopping criterion, that is

$$|\max_{\alpha\in\mathcal{Q}} \varphi(\mathbf{p}, \alpha) - \varphi(\mathbf{p}^*, \alpha^*)| \leq \varepsilon$$

the maximum number of iterations N that the level method requires is bounded as follows:

$$N \leq \frac{2c(\lambda)L^2}{\varepsilon^2} \tag{28}$$

where

$$c(\lambda) = \frac{1}{(1-\lambda)^2\lambda(2-\lambda)}$$
$$L = \frac{1}{2}\sqrt{d}C^2 \max_{1\leq i\leq d} |\mathbf{v}_i|^2.$$

Note L is the Lipschitz constant of $\varphi(\mathbf{p}, \alpha)$. The convergence rate can be derived similarly as in [55].

We now analyze the overall complexity. In each iteration, the main complexity is bound by the complexity of an SVM solver, which is usually in the scale of $\mathcal{O}(n^{2.5})$, since the two subproblems of the level method are a linear programming and a simple quadratic programming, respectively. Therefore, in general, the overall complexity of the proposed algorithm can be bound by $\mathcal{O}(n^{2.5}/\varepsilon^2)$.

C. Extension to Multiclass Feature Selection

Before presenting how to extend the model of semi-supervised feature selection to multiclass problems, we first discus the approaches related to multiclass SVM. Generally, multiclass SVM can be solved by combining a series of binary classification problems. Standard approaches include the one-against-one approach and the one-against-others approach. For an overview on multiclass SVM, the readers can refer to [16], [25], where the performance of different implementations is compared. Multiclass SVM can also be regarded as a global optimization problem [14], a structured-output SVM [29], or a Bayesian inference problem [63]. Here we are not concerning the aspect of which approach is better. Instead, we will show the multiclass semi-supervised feature selection problem can also be as easily solved as the binary problem. In the following, we employ the one-against-others approach to implement the multiclass semi-supervised feature selection.

Consider a data set with N classes, denoted by $\{C_1, \ldots, C_N\}$. We denote the number of examples in class C_i as n_i. For the one-against-others approach, there are totally $t = N$ binary classification problems. In this case, we can write the multiclass feature selection problem in the following way:

$$\min_{\mathbf{p}\in\mathcal{P}} \max_{\hat{\alpha}\in\hat{\mathcal{Q}}} \hat{\alpha}^\top \mathbf{e} - \frac{1}{2}(\hat{\alpha}\circ\hat{\mathbf{y}})^\top \left((1-\tau)^2 \sum_{i=1}^{m} p_i \hat{\mathbf{K}}_i + \tau^2 \hat{\mathbf{H}}\right)(\hat{\alpha}\circ\hat{\mathbf{y}})$$

where

$$\hat{\alpha} = [\alpha^1, \ldots, \alpha^N] \in \mathbb{R}^{n\times N}$$
$$\hat{\mathbf{y}} = [\mathbf{y}^1, \ldots, \mathbf{y}^N] \in \mathbb{R}^{n\times N}$$
$$\hat{\mathbf{K}}_i = \mathrm{D}(\mathbf{S}_i^1, \ldots, \mathbf{S}_i^N) \in \mathbb{R}^{n\times N} \times \mathbb{R}^{n\times N}$$
$$\hat{\mathbf{H}} = \mathrm{D}(\mathbf{H}, \ldots, \mathbf{H}) \in \mathbb{R}^{n\times N} \times \mathbb{R}^{n\times N}.$$

In the above multiclass feature selection problem, we suppose the kth classifier is composed by the kth class and the rest classes. $\alpha^k \in \mathbb{R}^n$ (for $1 \leq k \leq N$) is the dual variable for the kth classifier. For the kth classifier, the class labels $\hat{\mathbf{y}}^k \in \{-1, +1\}^n$ satisfying $\hat{\mathbf{y}}_h^k = +1$ for $1 \leq h \leq n$ if $\mathbf{x}_h^k$ belongs to C_i and $\hat{\mathbf{y}}_h^k = -1$ otherwise. $\mathbf{S}_i^k \in \mathbb{R}^n \times \mathbb{R}^n$ denotes the kth block of the block-diagonal kernel matrix $\hat{\mathbf{K}}_i$ constructed by the ith kernel function for the kth classifier.

Now we can solve the semi-supervised multiclass feature selection problem using the Levle method. The only change is the calculation of the gradient. The gradient for each element of $\mathbf{p}$ can be calculated as

$$[\nabla_{\mathbf{p}} f(\mathbf{p}, \alpha)]_i = -\frac{1}{2}(\hat{\alpha}\circ\hat{\mathbf{y}})^\top \hat{\mathbf{K}}_i (\hat{\alpha}\circ\hat{\mathbf{y}}) \; i = 1, \ldots, m.$$

Correspondingly, the cutting plane model $g^i(\mathbf{p})$ in the ith iteration can be calculated as

$$g^i(\mathbf{p}) = \max_{1\leq j\leq i} \varphi(\mathbf{p}^j, \alpha^j) + (\mathbf{p} - \mathbf{p}^j)^\top \nabla_{\mathbf{p}} \varphi(\mathbf{p}^j, \alpha^j).$$

TABLE I
TEXT DATA SETS FOR EVALUATION

Corpus	Data Set	# Cat	# Doc	# Dim
20-NG	*auto versus motor*	2	2000	5341
	baseball versus hockey	2	2000	6311
	gun versus mideast	2	2000	7821
	news.rec	4	2000	8959
	news.sci	4	2000	9674
	news.talk	3	3000	9533
Reuters	*money versus trade*	2	1203	2498
	ship versus trade	2	772	2321

Cat, # Doc, and # Dim denote the number of categories, the number of documents, and the number of features, respectively.

V. EXPERIMENTS

In this section, we thoroughly compare the proposed semi-supervised feature selection method with previous state-of-the-art feature selection algorithms. In the following, we will introduce the data sets employed in the paper, the experimental setup and experimental results.

A. Data Description

We adopt two types of data sets: digit characters and text documents. For the data sets of digit characters, we select three tasks from the United States Postal Service (USPS) data set, i.e., *4 versus 7*, *2 versus 3*, and *3 versus 8*, to make the learning tasks more challenging. For each task, we randomly select 400 digit images to form a data set. Each digit image is a 16×16 gray scale image. For the data sets of text documents, five subsets of text documents are selected from two benchmark text corpora, i.e., 20-Newsgroups (20-NG) and Reuters-21578. Their detailed information is summarized in Table I.

B. Experimental Setup

We denote by FS-Manifold the proposed discriminative feature selection method based on manifold regularization. We compare our algorithm with the following state-of-the-art approaches for feature selection: Fisher [5], L_0-SVM [51]

TABLE II
CLASSIFICATION ACCURACY (%) ON BINARY TEXT DATA SETS

Data Set	#F	FS-Manifold	L_1-SVM	L_0-SVM	Fisher
auto versus motor	50	**82.9**±2.4	82.2±2.9	82.3±2.9	82.3±3.5
	100	**83.5**±2.2	82.9±2.6	**83.2**±2.6	**83.4**±2.6
baseball versus hockey	50	**89.7**±3.9	88.7±8.6	89.1±4.9	**89.8**±6.9
	100	**91.1**±3.4	**90.9**±5.8	90.3±3.7	90.3±5.6
gun versus mideast	50	**84.2**±4.3	82.0±4.4	82.9±4.3	81.3±4.7
	100	**85.8**±3.9	84.1±4.2	85.2±4.4	84.3±4.1
money versus trade	50	**90.1**±1.7	**89.4**±2.4	**90.0**±2.0	89.1±2.7
	100	**90.7**±1.6	89.7±2.2	**90.5**±1.5	**90.0**±2.5
ship versus trade	50	**95.4**±1.6	94.1±2.2	94.6±1.7	94.3±2.2
	100	**95.9**±1.3	95.0±1.7	**95.4**±1.6	**95.3**±1.5

#F denotes the number of selected features. The best result, and those not significantly worse than it (t-test with 95% confidence level), are highlighted.

and L_1-SVM [20]. The description of the selected comparison methods is as follows.

1) Fisher [5] calculates a Fisher/Correlation score for each feature.
2) L_0-SVM [51] approximates the L_0-norm by minimizing a logarithm function.
3) L_1-SVM [20] replaces L_2-norm of the weights $\mathbf{w}$ with L_1-norm in SVM and leads to a sparse solution.

For all the comparison methods, features with the largest scores are selected. SVM is used as the evaluation classifier since it is usually regarded as the state-of-the-art classification method.

It is important to note that we also compare the above methods with the semi-supervised feature selection method proposed in [65], which selects features according to the spectral and the normalized mutual information. However, given the small amount of training data used in our semi-supervised learning algorithm, it is usually difficult to tune parameters for the optimal setup. Furthermore, due to the weak interaction between features and the class labels, it is unstable in the scenario of small training samples and it usually performs significantly worse than L_0-SVM. Therefore, we do not include its results in this paper.

The regularization parameter C in all SVM-based feature selection methods is chosen from the set $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100, 1000\}$ by a 5-fold cross validation. The tradeoff parameter τ in our proposed FS-Manifold is also tuned by a 5-fold cross validation and selected from $\{0, 0.1, \ldots, 0.9, 1\}$. The parameter ρ is fixed to 10, since the tradeoff is naturally taken care of by the parameter τ. To calculate the Laplacian, a graph is constructed. We adopt the Cosine similarity measure and the binary weights. The number of neighbors is set to 20 for all cases. In addition, we set the parameter λ in the level method to 0.9, since a larger λ means more regularization from the previous solution and thus the solution is more stable especially when it is near to the optimal solution.

We adopt two settings for semi-supervised feature selection. One is the transductive setting: all the test data are used as unlabeled data during training. Another is semi-supervised setting: a part of the test data are employed as unlabeled data and the left data are regarded as new data and not involved in training.

TABLE III
CLASSIFICATION ACCURACY (%) ON MULTI-CATEGORY TEXT DATA SETS

Data Set	#F	FS-Manifold	L_0-SVM	Fisher
news.rec	200	**73.1**±3.3	**72.8**±2.0	72.1±2.0
news.sci	200	**67.4**±3.3	66.5±3.0	66.4±3.4
news.talk	200	**57.2**±2.4	55.4±2.1	54.7±2.4

The best result, and those not significantly worse than it (t-test with 95% confidence level), are highlighted.

In the following, we first present experiments on transductive settings for both the USPS digits recognition task and the text categorization task. Then we present experiments under semi-supervised setting.

C. Experiments on USPS Digits Recognition

In this experiment, the training examples are randomly selected such that each category has the same number of examples. The remaining examples are then employed as the test data. As the USPS data sets are engaged to examine how the property of features changes with the number of labeled examples, we vary the number of training examples within the set of $\{6, 10, 20, 30, 40\}$. For each setting of the training samples, the number of selected features is set to 10, 20, and 30, respectively. This is because a small number of features (pixels) are enough to identify the digits. In all cases, every experiment is repeated with 30 random trials.

In the following, we first examine the results of the USPS data sets. We plot the results on the USPS data sets averaged over 30 runs in Figs. 2–4. Fig. 2 shows the test accuracy of the feature selection algorithms when the number of required features is set to 10.

First, we analyze the experimental results from the perspective of embedding the feature selection process to the classification method. It can be observed that the maximum margin based methods (SVM-based methods) usually perform better in identifying the discriminative features comparing with the non-SVM based method, Fisher. The advantage is more significant for the proposed semi-supervised feature selection method, i.e., FS-Manifold. For example, for the task of *4 versus 7*, when the number of training samples is 30 and the number of required features is 10, the improvement of FS-

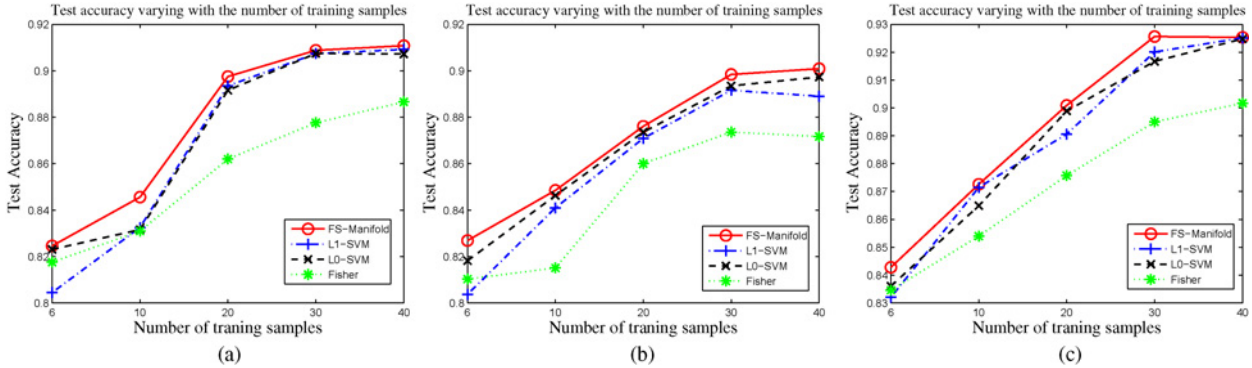


Fig. 2. Comparison among different feature selection algorithms when the number of selected features is equal to 10. The number of training samples is set as 6, 10, 20, 30, and 40, respectively. (a) 2 *versus* 3. (b) 3 *versus* 8. (c) 4 *versus* 7.

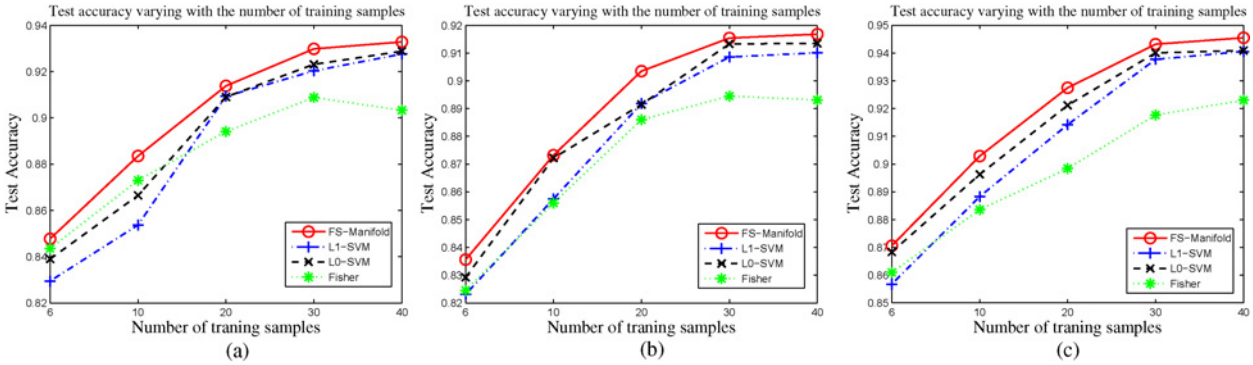


Fig. 3. Comparison among different feature selection algorithms when the number of selected features is equal to 20. The number of training samples is set as 6, 10, 20, 30, and 40, respectively. (a) 2 *versus* 3. (b) 3 *versus* 8. (c) 4 *versus* 7.

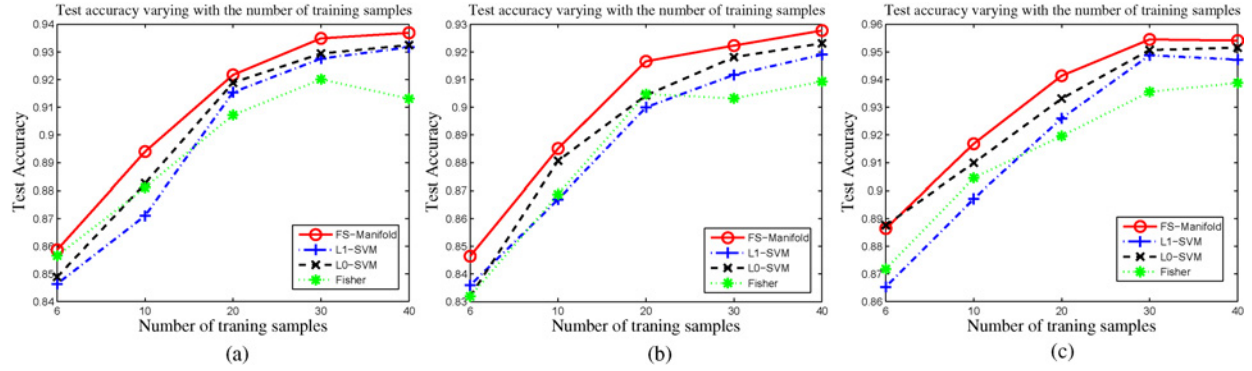


Fig. 4. Comparison among different feature selection algorithms when the number of selected features is equal to 30. The number of training samples is set as 6, 10, 20, 30, and 40, respectively. (a) 2 *versus* 3. (b) 3 *versus* 8. (c) 4 *versus* 7.

Manifold over Fisher is over 3%. This indicates the advantage of embedding the feature selection process to the classifier.

Second, we analyze the results from the perspective of whether the unlabeled data are employed. Compared with the supervised feature selection methods, FS-Manifold achieves promising test accuracy. In a number of cases, FS-Manifold outperforms the supervised feature selection methods. This is because the information supplied by the manifold structure of the unlabeled data helps to identify the global smooth features where the data lie in.

Figs. 3 and 4 show the test accuracy when the number of required features is set to 20 and 30, respectively. Consistent results are also observed. This, therefore, indicates the importance of the proposed semi-supervised feature selection method, which takes advantage of the maximum margin principle and the manifold regularization principle.

D. Experiments on Text Categorization

For the text data sets, we fix the number of training documents to be 50, since the scales of the text data sets are significantly larger than those of the USPS data sets. For each text data set, we consider two settings that the number of required features is equal to 50 and 100, respectively. It is interesting to note that the features (words) in the text data sets are very sparse and therefore more features are needed to represent the documents.

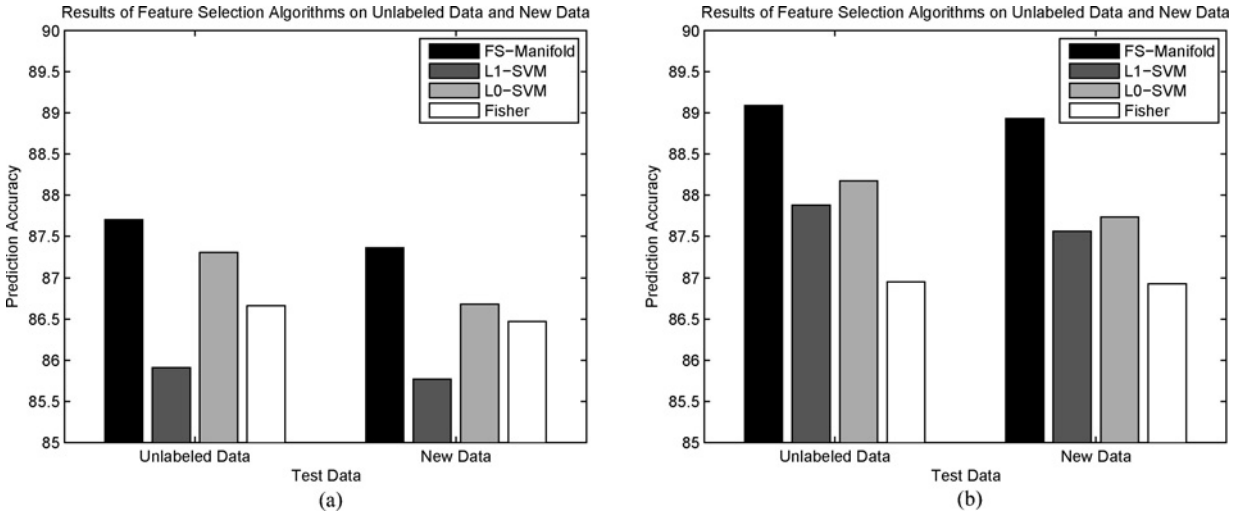


Fig. 5. Accuracy of different feature selection algorithms for unlabeled data and new test data, respectively. (a) *Baseball versus hockey*. The number of selected features is set to 50. (b) *Baseball versus hockey*. The number of selected features is set to 100.

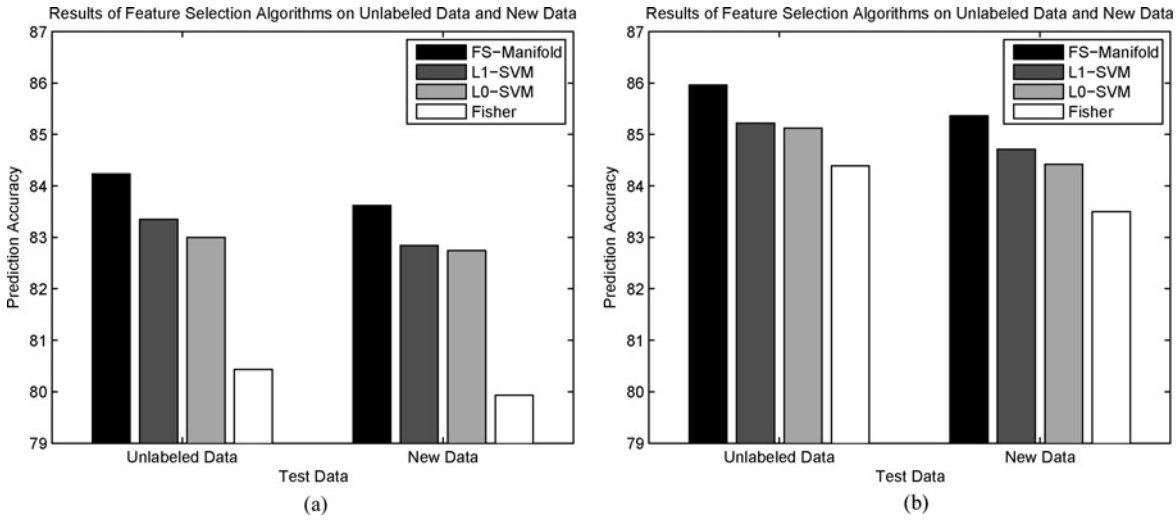


Fig. 6. Accuracy of different feature selection algorithms for unlabeled data and new test data, respectively. (a) *Gun versus mideast*. The number of selected features is set to 50. (b) *Gun versus mideast*. The number of selected features is set to 100.

We report the averaged prediction accuracy and the standard deviation on the text data sets in Table II. The best result, and those not significantly worse than it (t-test with 95% confidence level), are highlighted. We can observe that the proposed semi-supervised feature selection method performs better than other methods in lot of cases. For example, in the *gun versus mideast* data set, the improvement of FS-Manifold over Fisher is nearly 4% when the number of selected features is equal to 50. Furthermore, it is important to note that, for each data set, FS-Manifold achieves smaller deviation values than other feature selection methods. This phenomenon, which may be due to the global smoothness induced by the manifold regularization, suggests that FS-Manifold is more robust in selecting features.

We then conduct experiments for multiclass classification. For the multiclass data, the number of training examples is set to 100 and the number of selected features is set to 200. This is because the task of feature selection for multiclass text categorization becomes more challenging than binary classification. We show the results experiments on multiclass classification in Table III. Consistent with the results of binary classification, the proposed algorithm achieves better categorization accuracy than other algorithms.

E. Semi-Supervised Setting

To understand the generalization of the proposed feature selection algorithm, we conduct experiments on the semi-supervised setting, i.e., part of the test data are not employed in training. For clear of presentation, we regard this part of test data as new data. Under this setting, we divide the test data into two equal size parts: one for training, and the other for testing. The other settings remain the same as the transductive setting.

We show the results of the selected data sets, i.e., *baseball versus hockey* and *gun versus mideast*, in Figs. 5 and 6. Results on other data sets are consistent with the results of the

selected data sets. It can be observed that in all cases for both data sets, the proposed feature selection algorithm achieves better accuracy on both the unlabeled data and the new coming test data. For example, for *baseball versus hockey*, when the number of features is set to 100, as shown in Fig. 5(b), the improvement of the proposed FS-Manifold over Fisher is about 2%. It is important to note that the accuracy on the new data is consistent with the accuracy on the unlabeled training data.

In summary, the proposed semi-supervised feature selection algorithm improves the accuracy of supervised feature selection algorithms. The t-test with the confidence of 95% indicates that the improvement in a number of cases is significant, which is much clearer in text data. Unfortunately, we did not observe the improvement in all cases. It should be noted that semi-supervised learning is sensitive to data distributions and experimental settings. This is also observed in [10], where different algorithms seem to have clearly different performance. Furthermore, it is still unclear in what theoretical conditions semi-supervised learning would outperform supervised learning [32], [46]. As semi-supervised feature selection is a newly emerging and challenging topic, it is very deserving to find the theoretical conditions where it would improve supervised feature selection.

VI. Conclusion and Future Work

We have presented a discriminative semi-supervised feature selection method via manifold regularization. The proposed method selects features through maximizing the margin between different classes and at the same time exploiting the geometry of the probability distribution that generates the data. Comparing with other supervised and semi-supervised feature selection algorithms, our proposed semi-supervised feature selection method is an embedded feature selection method and is able to find more discriminative features. We successfully formulate the resulting semi-supervised feature selection method as a concave-convex optimization problem, where the saddle point corresponds to the optimal solution. We then derive an extended level method to find the optimal solution of the concave-convex problem. Empirical evaluation with several benchmark data sets demonstrates the effectiveness of our proposed feature selection method over the state-of-the-art feature selection methods.

This paper can be improved from several perspectives. One is to study how to efficiently solve large scale semi-supervised feature selection problems. Although extensive experiments are conducted to verify the efficacy of the proposed semi-supervised feature selection, our studies are mostly restricted to data set of modest size. In the next step, we plan to extend this paper to large data sets that consist of hundreds of thousands or even millions of training examples. The main computational difficulty of this paper is in calculating the regularization term related to manifold regularization. We will explore other optimization techniques to accelerate the calculation. Another perspective to improve this paper is to study the robustness of the selected features produced by the proposed algorithm when noisy features appear. Since the proposed algorithm employs an L_1-norm constraint for kernel/feature combination coefficients, intuitively, the L_1-norm constraint is robust to noisy features. In the future paper, we plan to verify the robustness of the proposed algorithm by testing it on toy or real-world data sets. Furthermore, we could also study how to employ robust optimization techniques on our model to further improve the robustness.

Appendix A
Proof of Theorem 1

Proof: First, using Lemma 1, it is straightforward to verify the problem in (2) is equivalent to the following min-max optimization problem:

$$\min_{\mathbf{p}\in\{0,1\}^d} \max_{\alpha\in\mathcal{Q}} h(\mathbf{p},\alpha)$$

where $h(\mathbf{p},\alpha)$ is defined as

$$h(\mathbf{p},\alpha) = \alpha^\top \mathbf{e} - \frac{1}{2}(\alpha\circ\mathbf{y})^\top \mathbf{X}_\ell \mathbf{Z}^{-1}\mathbf{W}\mathbf{Z}^{-1}\mathbf{X}_l^\top(\alpha\circ\mathbf{y})$$
$$\mathbf{W} = \mathbf{Z}\mathrm{D}(\mathbf{p})(\mathbf{I}+\rho\mathrm{D}(\mathbf{p})\mathbf{Z}\mathrm{D}(\mathbf{p}))^{-1}\mathrm{D}(\mathbf{p})\mathbf{Z}.$$

Using the matrix inverse lemma, we have

$$\begin{aligned}&\left(\frac{\mathbf{Z}^{-1}}{\rho}+\mathrm{D}(\mathbf{p})\mathrm{D}(\mathbf{p})\right)^{-1}\\ =\;& \rho\mathbf{Z}-\rho^2\mathbf{Z}\mathrm{D}(\mathbf{p})(\mathbf{I}+\rho\mathrm{D}(\mathbf{p})\mathbf{Z}\mathrm{D}(\mathbf{p}))^{-1}\mathrm{D}(\mathbf{p})\mathbf{Z}\\ =\;& \rho\mathbf{Z}-\rho^2\mathbf{W}.\end{aligned}$$

Hence, $h(\mathbf{p},\alpha)$ is written as

$$\begin{aligned}h(\mathbf{p},\alpha) \;=\;& \alpha^\top\mathbf{e}-\frac{1}{2\rho}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell\\ &\left(\mathbf{Z}^{-1}-\left[\mathbf{Z}+\rho\mathbf{Z}D(\mathbf{p})\mathbf{Z}\right]^{-1}\right)\mathbf{X}_\ell^\top(\alpha\circ\mathbf{y}).\end{aligned}$$

The above derivation is based on the fact that $\mathbf{p}$ is a binary vector and therefore $\mathrm{D}^2(\mathbf{p}) = \mathrm{D}(\mathbf{p})$. When $\rho \ll 1$, we could approximate $\mathbf{Z}^{-1} - \left[\mathbf{Z}+\rho\mathbf{Z}D(\mathbf{p})\mathbf{Z}\right]^{-1}$ by using the first order expansion of $\mathbf{Z}^{-1}$, i.e., $(\mathbf{Z}+\Delta)^{-1} \approx \mathbf{Z}^{-1} - \mathbf{Z}^{-1}\Delta\mathbf{Z}^{-1}$. This results in the following approximation of $h(\mathbf{p},\alpha)$:

$$h(\mathbf{p},\alpha) = \alpha^\top\mathbf{e} - \frac{1}{2}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell\mathrm{D}(\mathbf{p})\mathbf{X}_\ell^\top(\alpha\circ\mathbf{y}).$$

■

Appendix B
Proof of Theorem 2

Proof: It is easy to verify that $\phi(\mathbf{p},\alpha)$ is a concave function in terms of α. This is because $\nabla_\alpha^2\phi(\mathbf{p},\alpha) = -\frac{1}{\rho}\mathrm{D}(\mathbf{y})\mathbf{X}_\ell\left(\mathbf{Z}^{-1}-[\mathbf{Z}+\rho\mathbf{Z}\mathrm{D}(\mathbf{p})\mathbf{Z}]^{-1}\right)\mathbf{X}_\ell\mathrm{D}(\mathbf{y})$.

It is clear that $\nabla_\alpha^2\phi(\mathbf{p},\alpha) \preceq 0$, and therefore $\phi(\mathbf{p},\alpha)$ is concave in α. To show that the minimization of $\phi(\mathbf{p},\alpha)$ with respect to $\mathbf{p}$ is a convex optimization problem, we first extract the two terms in $\phi(\mathbf{p},\alpha)$ that are dependent on $\mathbf{p}$, and denote their sum by $\psi(\mathbf{p},\alpha)$, i.e., $\psi(\mathbf{p},\alpha) = \alpha^\top\mathbf{e} + \frac{1}{2\rho}(\alpha\circ\mathbf{y})^\top\mathbf{X}_\ell\left(\mathbf{Z}+\rho\mathbf{Z}\mathrm{D}(\mathbf{p})\mathbf{Z}\right)^{-1}\mathbf{X}_\ell(\alpha\circ\mathbf{y})$.

We thus need to show

$$\min_{\mathbf{p}\in\mathcal{P}} \psi(\mathbf{p},\alpha)$$

is a convex optimization problem. To this end, we introduce a slack variable t to bound the second term in $\psi(\mathbf{p}, \alpha)$, that is

$$2\rho t \geq (\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell \left(\mathbf{Z} + \rho \mathbf{Z} \mathrm{D}(\mathbf{p}) \mathbf{Z}\right)^{-1} \mathbf{X}_\ell (\alpha \circ \mathbf{y}).$$

Using the Schur complement, the above inequality constraint is converted into the following linear matrix inequality:

$$\begin{pmatrix} \mathbf{Z} + \rho \mathbf{Z} \mathrm{D}(\mathbf{p}) \mathbf{Z} & \mathbf{X}_l^\top (\alpha \circ \mathbf{y}) \\ (\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell & 2\rho t \end{pmatrix} \succeq 0.$$

As a result, the minimization of $\psi(\mathbf{p}, \alpha)$ with respect to $\mathbf{p}$ is rewritten into the following semi-definite programming problem, which is a standard convex optimization problem:

$$\begin{aligned} \min_{\mathbf{p} \in \mathcal{P}} \quad & \alpha^\top \mathbf{e} + t \\ \text{s.t.} \quad & \begin{pmatrix} \mathbf{Z} + \rho \mathbf{Z} \mathrm{D}(\mathbf{p}) \mathbf{Z} & \mathbf{X}_l^\top (\alpha \circ \mathbf{y}) \\ (\alpha \circ \mathbf{y})^\top \mathbf{X}_\ell & 2\rho t \end{pmatrix} \succeq 0. \end{aligned}$$

■

APPENDIX C
PROOF OF PROPOSITION 4

Proof: According to the Schur complement, the condition $\mathbf{A} \succeq \Gamma$ is equivalent to the following constraint:

$$\begin{pmatrix} \mathbf{A} & \mathrm{D}(\mathbf{p}) \\ \mathrm{D}(\mathbf{p}) & \mathbf{I} + \rho \mathrm{D}(\mathbf{p}) \mathbf{Z} \mathrm{D}(\mathbf{p}) \end{pmatrix} \succeq 0. \tag{29}$$

The necessary condition for the condition (29) to hold is that $\mathbf{A} = \mathbf{A}_1 + \mathbf{A}_2$ such that

$$\begin{pmatrix} \mathbf{A}_1 & (1-\tau)\mathrm{D}(\mathbf{p}) \\ (1-\tau)\mathrm{D}(\mathbf{p}) & \mathbf{I} \end{pmatrix} \succeq 0 \tag{30}$$

$$\begin{pmatrix} \mathbf{A}_2 & \tau \mathrm{D}(\mathbf{p}) \\ \tau \mathrm{D}(\mathbf{p}) & \rho \mathrm{D}(\mathbf{p}) \mathbf{Z} \mathrm{D}(\mathbf{p}) \end{pmatrix} \succeq 0 \tag{31}$$

where $0 \leq \tau \leq 1$. Add (30) to (31), we therefore have

$$\mathbf{A} = (1-\tau)^2 \mathrm{D}(\mathbf{p}) + \frac{\tau^2}{\rho} \mathbf{Z}^{-1}.$$

■

REFERENCES

[1] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm," in *Proc. 21th Int. Conf. Mach. Learning (ICML)*, 2004, pp. 41–48.

[2] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems*, vol. 14. Cambridge, MA: MIT Press, 2002, pp. 585–591.

[3] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learning Res.*, vol. 7, pp. 2399–2434, Nov. 2006.

[4] C. Bhattacharyya, "Second order cone programming formulation for feature selection," *J. Mach. Learning Res.*, vol. 5, pp. 1417–1433, Dec. 2004.

[5] C. Bishop, *Neural Networks for Pattern Recognition*. London, U.K.: Oxford Univ. Press, 1995.

[6] A. Blum and S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," in *Proc. 18th Int. Conf. Mach. Learning (ICML)*, 2001, pp. 19–26.

[7] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 245–271, 1997.

[8] P. S. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proc. 15th Int. Conf. Mach. Learning (ICML)*, 1998, pp. 82–90.

[9] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006.

[10] O. Chapelle, V. Sindhwani, and S. S. Keerthi, "Optimization techniques for semi-supervised support vector machines," *J. Mach. Learning Res.*, vol. 9, pp. 203–233, Feb. 2008.

[11] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," in *Proc. 10th Int. Workshop Artif. Intell. Statist.*, 2005, pp. 57–64.

[12] R. Collobert, F. Sinz, J. Weston, and L. Bottou, "Large scale transductive SVMs," *J. Mach. Learning Res.*, vol. 7, pp. 1687–1712, Dec. 2006.

[13] C. Cortes, M. Mohri, and A. Rostamizadeh, "Learning sequence kernels," in *Proc. IEEE Workshop Mach. Learning Signal Process.*, 2008, pp. 2–8.

[14] K. Crammer, "On the learnability and design of output codes for multiclass problems," in *Proc. 13th Annu. Conf. Comput. Learning Theory (COLT)*, 2000, pp. 35–46.

[15] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. S. Kandola, "On kernel-target alignment," in *Proc. 13th Neural Inform. Process. Syst. (NIPS)*, 2001, pp. 367–373.

[16] K. Duan and S. S. Keerthi, "Which is the best multiclass SVM method? An empirical study," in *Proc. 6th Int. Workshop Multiple Classifier Syst.*, 2005, pp. 278–285.

[17] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *J. Mach. Learning Res.*, vol. 5, pp. 845–889, Dec. 2004.

[18] B. Efron, T. Hastie, L. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.

[19] P. Estevez, M. Tesmer, C. Perez, and J. Zurada, "Normalized mutual information feature selection," *IEEE Trans. Neural Netw.*, vol. 20, no. 2, pp. 189–201, Feb. 2009.

[20] G. Fung and O. L. Mangasarian, "Data selection for support vector machine classifiers," in *Proc. 6th Assoc. Comput. Machinery Special Interest Group Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2000, pp. 64–70.

[21] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learning Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[22] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learning*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[23] J. Handl and J. Knowles, "Semi-supervised feature selection via multiobjective optimization," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2006, pp. 3319–3326.

[24] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, vol. 16. Cambridge, MA: MIT Press, 2003.

[25] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, 2002.

[26] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, Feb. 1997.

[27] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th Int. Conf. Mach. Learning (ICML)*, 1999, pp. 200–209.

[28] T. Joachims, "Transductive learning via spectral graph partitioning," in *Proc. 20th Int. Conf. Mach. Learning (ICML)*, 2003, pp. 290–297.

[29] T. Joachims, "Training linear SVMS in linear time," in *Proc. 12th Assoc. Comput. Machinery Special Interest Group Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2006, pp. 217–226.

[30] I. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 1986.

[31] D. Koller and M. Sahami, "Toward optimal feature selection," in *Proc. 24th Int. Conf. Mach. Learning (ICML)*, 1996, pp. 284–292.

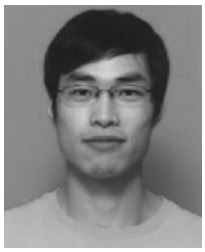
[32] J. Lafferty and L. Wasserman, "Statistical analysis of semi-supervised regression," in *Advances in Neural Information Processing Systems*, vol. 20, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, pp. 801–808.

[33] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *J. Mach. Learning Res.*, vol. 5, pp. 27–72, Dec. 2004.

[34] C. Lemaréchal, A. Nemirovski, and Y. Nesterov, "New variants of bundle methods," *Math. Programming*, vol. 69, no. 1, pp. 111–147, 1995.

[35] J. Li, M. T. Manry, P. L. Narasimha, and C. Yu, "Feature selection using a piecewise linear network," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1101–1115, Sep. 2006.

[36] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005.
[37] J. Neumann, C. Schnörr, and G. Steidl, "Combined SVM-based feature selection and classification," *Mach. Learning*, vol. 61, nos. 1–3, pp. 129–150, 2005.
[38] A. Y. Ng, "Feature selection, L1 versus L2 regularization, and rotational invariance," in *Proc. 21st Int. Conf. Mach. Learning (ICML)*, 2004, pp. 78–86.
[39] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA: MIT Press, 1999, pp. 185–208.
[40] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "More efficiency in multiple kernel learning," in *Proc. 24th Int. Conf. Mach. Learning (ICML)*, 2007, pp. 775–782.
[41] J. Ren, Z. Qiu, W. Fan, H. Cheng, and P. S. Yu, "Forward semi-supervised feature selection," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining (PAKDD)*, 2008, pp. 970–976.
[42] E. Romero and J. Sopena, "Performing feature selection with multilayer perceptrons," *IEEE Trans. Neural Netw.*, vol. 19, no. 3, pp. 431–441, Mar. 2008.
[43] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
[44] B. Scholkopf, A. Smola, and K. R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.
[45] V. Sindhwani, P. Niyogi, and M. Belkin, "Beyond the point cloud: From transductive to semi-supervised learning," in *Proc. Int. Conf. Mach. Learning*, 2005, pp. 824–831.
[46] A. Singh, R. Nowak, and X. Zhu, "Unlabeled data: Now it helps, now it doesn't," in *Advances in Neural Information Processing Systems*, vol. 21, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2009, pp. 1513–1520.
[47] A. Smola, S. V. N. Vishwanathan, and Q. Le, "Bundle methods for machine learning," in *Advances in Neural Information Processing Systems*, vol. 20, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, pp. 1377–1384.
[48] L. Song, A. Smola, A. Gretton, K. M. Borgwardt, and J. Bedo, "Supervised feature selection via dependence estimation," in *Proc. 24th Int. Conf. Mach. Learning (ICML)*, 2007, pp. 823–830.
[49] S. Sonnenburg, G. Rätsch, C. Schäfer, and B. Schölkopf, "Large scale multiple kernel learning," *J. Mach. Learning Res.*, vol. 7, pp. 1531–1565, Jul. 2006.
[50] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.
[51] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping, "Use of the zero norm with linear models and kernel methods," *J. Mach. Learning Res.*, vol. 3, pp. 1439–1461, Mar. 2003.
[52] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for SVMs," in *Advances in Neural Information Processing Systems*, vol. 13. Cambridge, MA: MIT Press, 2000, pp. 668–674.
[53] L. Wolf and A. Shashua, "Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach," *J. Mach. Learning Res.*, vol. 6, pp. 1855–1887, Dec. 2005.
[54] L. Xu and D. Schuurmans, "Unsupervised and semi-supervised multi-class support vector machines," in *Proc. Assoc. Adv. Artif. Intell.*, 2005, pp. 904–910.
[55] Z. Xu, "Learning with unlabeled data," Ph.D. dissertation, Dept. Comput. Sci., Chinese Univ. Hong Kong, Shatin, Hong Kong, 2009.
[56] Z. Xu, R. Jin, I. King, and M. Lyu, "An extended level method for efficient multiple kernel learning," in *Advances in Neural Information Processing Systems*, vol. 21, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. 2009, pp. 1825–1832.
[57] Z. Xu, R. Jin, J. Ye, M. R. Lyu, and I. King, "Discriminative semi-supervised feature selection via manifold regularization," in *Proc. 21th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2009, pp. 1303–1308.
[58] Z. Xu, R. Jin, J. Ye, M. R. Lyu, and I. King, "Non-monotonic feature selection," in *Proc. 26th Annu. Int. Conf. Mach. Learning (ICML)*, 2009, pp. 1145–1152.
[59] Z. Xu, R. Jin, J. Zhu, I. King, M. Lyu, and Z. Yang, "Adaptive regularization for transductive support vector machine," in *Advances in Neural Information Processing Systems*, vol. 22, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, Eds. Cambridge, MA: MIT Press, 2009, pp. 2125–2133.
[60] Z. Xu, R. Jin, J. Zhu, I. King, and M. R. Lyu, "Efficient convex relaxation for transductive support vector machine," in *Advances in Neural Information Processing Systems*, vol. 20, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, pp. 1641–1648.
[61] Z. Xu, I. King, and M. R. Lyu, "Feature selection based on minimum error minimax probability machine," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 21, no. 8, pp. 1–14, 2007.
[62] X. Yang, H. Fu, H. Zha, and J. L. Barlow, "Semi-supervised nonlinear dimensionality reduction," in *Proc. 23rd Int. Conf. Mach. Learning (ICML)*, 2006, pp. 1065–1072.
[63] Z. Zhang and M. I. Jordan, "Bayesian multicategory support vector machines," in *Proc. 22nd Conf. Uncertainty Artif. Intell. (UAI)*, 2006.
[64] J. Zhao, K. Lu, and X. He, "Locality sensitive semi-supervised feature selection," *Neurocomputing*, vol. 71, nos. 10–12, pp. 1842–1849, 2008.
[65] Z. Zhao and H. Liu, "Semi-supervised feature selection via spectral analysis," in *Proc. 7th SIAM Data Mining Conf. (SDM)*, 2007, pp. 641–646.
[66] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proc. 24th Int. Conf. Mach. Learning (ICML)*, 2007, pp. 1151–1157.
[67] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems*, vol. 16, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004.
[68] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, Tech. Rep. 1530, 2005.
[69] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proc. 20th Int. Conf. Mach. Learning (ICML)*, 2003, pp. 912–919.



Zenglin Xu (M'07) received the B.S. degree in computer science and technology from Xi'an Polynomial University, Xi'an, China, the M.S. degree in computer software and theory from Xi'an Jiaotong University, Xi'an, China, and the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Shatin, Hong Kong.

In 2007 and 2008, he was a Visiting Student of Professor R. Jin at Michigan State University, East Lansing, where he worked on the problem of semi-supervised learning and kernel learning. He is currently a Post-Doctoral Researcher in Cluster of Excellence, Saarland University, Max Planck Institute for Informatics, Saarbrucken, Germany.

Irwin King (SM'08) received the B.S. degree in engineering and applied science from the California Institute of Technology, Pasadena, in 1984, and the M.S. and Ph.D. degrees, both in computer science, from the University of Southern California, Los Angeles, in 1986 and 1993, respectively.

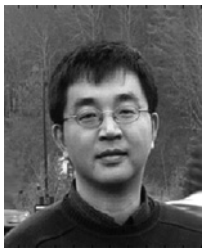
Since 1993, he has been with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Shatin, Hong Kong. He is currently an Associate Professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong. In his research areas, he has over 200 technical publications in journals and conferences. In addition, he has contributed over 20 book chapters and edited volumes. He has over 30 research and applied grants. His current research interests include machine learning, web intelligence, social computing, data mining, and multimedia information processing.

Dr. King is an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS and the IEEE COMPUTATIONAL INTELLIGENCE MAGAZINE. He is a Member of the Association for Computing Machinery, the International Neural Network Society (INNS), and the Asian Pacific Neural Network Assembly (APNNA). Currently, he is serving the Neural Network Technical Committee and the Data Mining Technical Committee under the IEEE Computational Intelligence Society (formerly the IEEE Neural Network Society). He is also a Member of the Board of Governors of INNS, and a Vice-President and Governing Board Member of APNNA. He is serving or has served as a Program and/or Organizing Member in numerous top international conferences and workshops. He has also served as a Reviewer for international conferences as well as journals.

Michael R. Lyu (F'04) received the B.S. degree in electrical engineering from National Taiwan University, Taipei City, Taiwan, in 1981, the M.S. degree in computer science from the University of California, Santa Barbara, in 1985, and the Ph.D. degree in computer science from the University of California, Los Angeles, in 1988.

From 1988 to 1990, he was a Technical Staff Member with the Jet Propulsion Laboratory, Pasadena, CA. From 1990 to 1992, he was an Assistant Professor with the Department of Electrical and Computer Engineering, University of Iowa, Iowa City. From 1992 to 1995, he was a Member of the Technical Staff in the applied research area of Bell Communications Research (Bellcore), Piscataway, NJ. From 1995 to 1997, he was a Research Member of the Technical Staff with Bell Laboratories, which was first part of AT&T, Dallas, TX, and later became part of Lucent Technologies, Inc., Murray Hill, NJ. He is currently a Professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Shatin, Hong Kong. He has been frequently invited as a Keynote or Tutorial Speaker to conferences and workshops in the U.S., Europe, and Asia. He has published over 300 refereed journal and conference papers in his research areas. He initiated the first International Symposium on Software Reliability Engineering in 1990. He is the Editor for two book volumes: *Software Fault Tolerance* (New York: Wiley, 1995) and *Handbook of Software Reliability Engineering* (New York: IEEE and McGraw-Hill, 1996). These books have received an overwhelming response from both the academia and the industry. His current research interests include software reliability engineering, distributed systems, fault-tolerant computing, web technologies, mobile networks, digital video library, multimedia processing, and video searching and delivery. He has participated in more than 30 industrial projects in these areas, and helped to develop many commercial systems and software tools.

Dr. Lyu was the Program Chair for the International Symposium on Software Reliability Engineering in 1996, the Program Co-Chair for the International World Wide Web Conference in 2010, the General Chair for the International Symposium on Software Reliability Engineering in 2001, the General Co-Chair for the Pacific Rim International Symposium on Dependable Computing in 2005, and has served in program committees for many conferences. He was an Associate Editor of the IEEE TRANSACTIONS ON RELIABILITY, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, and the *Journal of Information Science and Engineering*. He is currently on the Editorial Board of the Wiley *Software Testing, Verification and Reliability Journal*. He was elected an IEEE Fellow in 2004 and an American Association for the Advancement of Science Fellow in 2007 for his contributions to software reliability engineering and software fault tolerance. He was also named a Croucher Senior Research Fellow in 2008.



Rong Jin received the B.A. degree in engineering from Tianjin University, Tianjin, China, in 1993, the M.S. degree in physics from Beijing University, Beijing, China, in 1996, and the M.S. and Ph.D. degrees in computer science from Carnegie Mellon University, Pittsburgh, PA, in 2000 and 2003, respectively.

He is currently an Associate Professor with the Department of Computer Science and Engineering, Michigan State University, East Lansing. He is working on the areas of statistical machine learning and its application to information retrieval. He has published more than 80 conference and journal articles on related topics.

Dr. Jin received the U.S. National Science Foundation Career Award in 2006.