2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Kernel-based Semi-supervised Learning for Novelty Detection

Van Nguyen, Trung Le, Thien Pham, Mi Dinh, and Thai Hoang Le

***Abstract*—One-class Support Vector Machine (OCSVM) is a well-known method for novelty detection. However, *OCSVM* regards all negative data samples as a common symbol and thereby not being able to utilize the information carried by them. Furthermore, *OCSVM* requires a fully labeled data set and cannot work efficiently with data set with both labeled and unlabeled data samples which is very popular nowadays. In this paper, we first extend the model of *OCSVM* to enable efficiently using the negative data samples. We then propose two methods to integrate the semi-supervised learning paradigm to the extended model for novelty detection purpose.**

***Index Terms*—Semi-supervised Learning, Novelty Detection, Kernel Method, One-class Classification.**

I. Introduction

In many applications of machine learning, abundant amounts of data can be cheaply and automatically collected. However, manual labeling for the purposes of training learning algorithms is often a slow, expensive, and error-prone process [1]. As a result, the collected data sets frequently consist of a collection of labeled data and a larger collection of unlabeled data. Semi-supervised learning involves employing the larger collection of unlabeled data jointly with smaller one of labeled data for improving generalization performance.

Support Vector Machine (SVM) [2, 3] has become the state-of-the-art classifier. *SVM* has its root in Statistical Learning Theory [4]. It is proven that the optimal hyperplane with the maximal margin maximizes the generalization ability of the linear classifier [4, 5]. The original *SVM* requires the fully labeling data sets. The idea of applying semi-supervised learning paradigm to *SVM* was first introduced by Vapnik and Sterin in 1977 [6]. However, it really attracted much concern of the machine-learning community after the work of Joachims [7]. So far, there have been many studies on *semi-supervised SVM* [8, 9, 10, 11, 12, 13, 14, 15].

SVM has been proven very successful for balanced data sets. However, it may not render the good performances for imbalanced data sets where one of two classes is under-sampled, or only data samples of one class are available for training [16]. To accommodate this issue, *One-class Support Vector Machine (OCSVM)* [17] was introduced. *OCSVM* aims at constructing an optimal hyperplane that can separate the origin and the normal data samples such that the margin, the distance from the origin to the hyperplane, is maximized. Although *OCSVM* offers the good performance for one-class classification problem, its obvious drawback is that *OCSVM* regards the origin as a common symbol for all abnormal data samples and may not efficiently utilize the information carried by them. Yet another successful kernel-based one-class classification method is *Support Vector Data Description (SVDD)* [18], which targets building an optimal hypersphere in the feature space which includes only normal (positive) data samples and excludes all abnormal (negative) data samples with tolerances. It appears that if an iso-metric kernel function, e.g., RBF Kernel, is used, *OCSVM* and *SVDD* are equivalent [17]. Both *OCSVM* and *SVDD* require the fully labeling data sets which rarely occurs in the application domains.

In this paper, we first extend the model of *OCSVM* to enable using of the abnormal data samples for classifying data. More concretely, an optimal hyperplane is constructed such that the margin, the distance from the closest negative data sample to the hyperplane rather than that of the origin, is maximized. Based on the extended model, we then present how to apply the semi-supervised learning paradigm to *OCSVM* to utilize the unlabeled data for increasing its generalization performance. Actually, in this paper, we have proposed two semi-supervised learning methods for novelty detection. The first proposed method is inspired from the *Transductive Support Vector Machine* of Joachims [7] whereas two temporary labels of two unlabeled data samples are swapped in a row to really decrease the objective function. This method is proven to gradually become better and to converge after a finite number of iterations to its local minima. The second proposed method associates each unlabeled data sample with a fuzzy membership which represents the possibility to assign it to the positive class. The temperature variable T is led to 0 to drive the objective function to its global minima. The experiment conducted on 14 *benchmark data sets of UCI repository* shows the superiority for the proposed methods.

II. One-class Support Vector Machine

OCSVM [17] aims at constructing an optimal hyperplane, such that the margin, the distance from the origin to the hyperplane, is maximized. Given the training set including l normal data samples $X = \{(x_1, y_1), ..., (x_l, y_l)\}$ where $y_i = 1, i = 1, ..., l$. The optimization problem of *OCSVM* is as follows:

$$\min_{w,\rho} \left(\frac{1}{2} \|w\|^2 - \rho + \frac{1}{\nu l} \sum_{i=1}^{l} \xi_i \right) \quad (1)$$

Van Nguyen, Trung Le, Thien Pham, and Mi Dinh are with the Faculty of Information Technology, the HCMc University of Pedagogy, Hochiminh city, Vietnam (email: {vannk, trunglm, thienph, and midtt}@hcmup.edu.vn).

Thai Hoang Le is with the Faculty of Information Technology, the HCMc University of Science, Hochiminh city, Vietnam (email:lhthai@fit.hcmus.edu.vn).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

subject to:

$$\begin{array}{l} w^T\phi(x_i) \geq \rho - \xi_i,\ i = 1, ..., l \\ \xi_i \geq 0,\ i = 1, ..., l \end{array} \quad (2)$$

where w is the normal vector of the hyperplane, ρ is the bias, ν is the trade-off parameter, $\phi(.)$ is a transformation from the input space to the feature space, and $\xi = [\xi_i]_{i=1,...,l}$ is vector of slack variables.

III. Large Margin One-class Support Vector Machine (LM-OCSVM)

A. The Idea of Large Margin One-class Support Vector Machine

Given the training set $X = \{(x_1, y_1), ..., (x_p, y_p), (x_{p+1}, y_{p+1}), ..., (x_{p+m}, y_{p+m})\}$ including both normal and abnormal data samples where $y_i = 1, i = 1, ..., p$ and $y_i = -1, i = p+1, ..., l$ with $l = p + m$, to decrease the chance of accepting abnormal as normal data, it desires to learn an optimal hyperplane that can separate the positive and negative data samples such that the margin, the distance from the closest negative data sample to the hyperplane, is maximized. This optimization problem is formulated as follows:

$$\underset{w,\rho}{max} \left(\underset{y_i=-1}{min} \left(\frac{y_i(w^T\phi(x_i) - \rho)}{\|w\|} \right) \right) \quad (3)$$

subject to

$$y_i(w^T\phi(x_i) - \rho) \geq 0,\ i = 1, ..., l \quad (4)$$

It occurs that the margin is invariant if we scale (w, ρ) by a factor k. Hence, without loss of generality, we can assume that: $\underset{y_i=-1}{min} (y_i(w^T\phi(x_i) - \rho)) = 1$. The above optimization is rewritten as follows:

$$\underset{w,\rho}{min}(\frac{1}{2}\|w\|^2) \quad (5)$$

subject to

$$\begin{array}{l} y_i(w^T\phi(x_i) - \rho) \geq 0,\ i = 1, ..., p \\ y_i(w^T\phi(x_i) - \rho) \geq 1,\ i = p+1, ..., l \end{array} \quad (6)$$

We refer the above model as hard model of *LM-OCSVM*. To derive the soft model, we extend the optimization problem in Eq. (5) by using the slack variables as follows:

$$\underset{w,\rho}{min}(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i) \quad (7)$$

subject to

$$\begin{array}{l} y_i(w^T\phi(x_i) - \rho) \geq -\xi_i,\ i = 1, ..., p \\ y_i(w^T\phi(x_i) - \rho) \geq 1 - \xi_i,\ i = p+1, ..., l \\ \xi_i \geq 0,\ i = 1, ..., l \end{array} \quad (8)$$

B. The Solution

We apply Karush-Kuhn-Tucker (KKT) theorem to derive the solution of the optimization problem in Eq. (7). The Lagrange function is of the following form:

$$\begin{array}{l} L(w, \rho, \xi, \alpha, \beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i \\ -\sum_{i=1}^{l}\alpha_i(y_i(w^T\phi(x_i) - \rho) - \theta_i + \xi_i) - \sum_{i=1}^{l}\beta_i\xi_i \end{array} \quad (9)$$

where $\theta_i = (1 - y_i)/2$. Setting the derivatives to 0, we gain:

$$\begin{array}{l} \frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=1}^{l} y_i\alpha_i\phi(x_i) \\ \frac{\partial L}{\partial \rho} = 0 \rightarrow \sum_{i=1}^{l} y_i\alpha_i = 0 \\ \frac{\partial L}{\partial \xi_i} = 0 \rightarrow \alpha_i + \beta_i = C,\ i = 1, ..., l \end{array} \quad (10)$$

Substituting Eq. (10) to the Lagrange function, we obtain the following optimization problem:

$$\underset{\alpha}{min} \left(\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_iy_jK(x_i, x_j)\alpha_i\alpha_j - \sum_{i=1}^{l}\alpha_i\theta_i \right) \quad (11)$$

subject to

$$\sum_{i=1}^{l}\alpha_iy_i = 0\ and\ 0 \leq \alpha_i \leq C,\ i = 1, ..., l \quad (12)$$

To calculate ρ, let us denote $I = \{i : 0 < \alpha_i < C\}$. For every $i \in I$, according to KKT condition, we have:

$$y_i(w^T\phi(x_i) - \rho) - \theta_i = 0 \leftrightarrow \rho = \sum_{j=1}^{l} y_j\alpha_jK(x_i, x_j) - y_i\theta_i \quad (13)$$

In practice, to avoid favoring any data sample, we take average of all right hand sides of Eq. (13) for calculating ρ.

C. Visual Explanation for LM-OCSVM

To visually demonstrate the performance of *LM-OCSVM*, we designed the experiments on two synthesized toy data sets. In the first experiment, we show that *LM-OCSVM* can take advantage from the negative data samples. The 2D toy data set was generated as in Figure 1 and the linear kernel was used. As seen in Figure 1, the margin, which is the distance from the closest negative data sample to the hyperplane, is maximized. This maximization really implies the maximal reducing of accepting abnormal as normal. In the second experiment, we generated a data set being the mixture of three Gaussian distributions of 200 data samples together with some negative data samples as shown in Figure 2 and RBF Kernel was employed. It is observed from Figure 2 that *LM-OCSVM* can perfectly recognize three Gaussian distributions and this shows the generalization ability of *LM-OCSVM*.

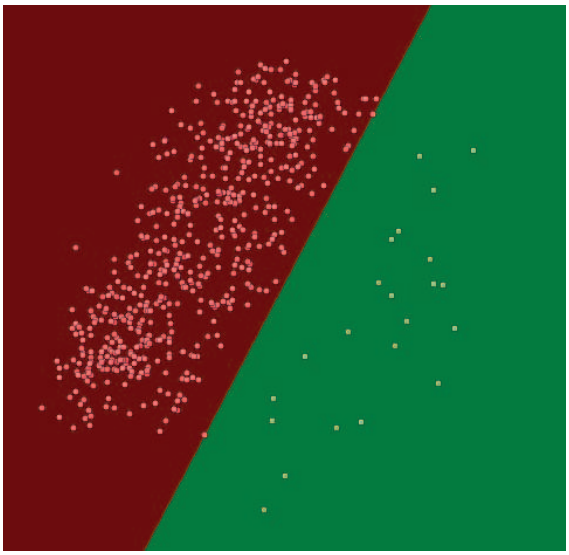

Fig. 1. LM-OCSVM can take advantage from the negative data samples.

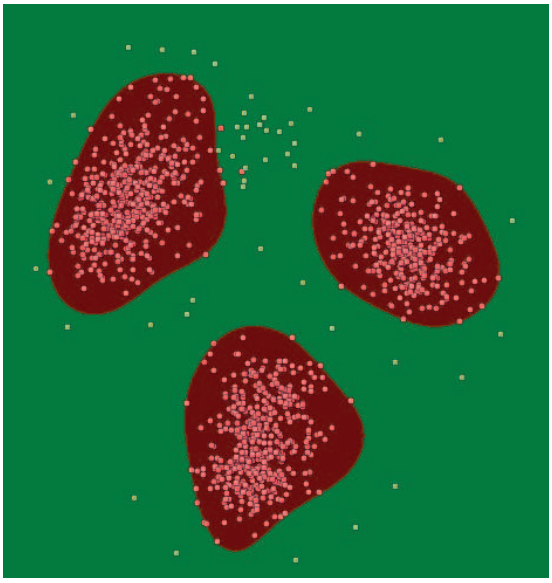

Fig. 2. LM-OCSVM can recognize three Gaussian distributions.

IV. Semi-supervised Large Margin One-class Support Vector Machine (S2LM-OCSVM)

A. The Problem Statement

Given a training set $X = X_l \cup X_u$ where $X_l = \{(x_1, y_1), (x_2, y_2), ..., (x_l, y_l)\}$ and $X_u = \{x_{l+1}, ..., x_n\}$ where $n = l + u$ are the labeled and unlabeled training set, respectively. It requires to use the unlabeled training set to enhance the generalization performance of classifier. We need to not only find out the optimal hyperplane but also assign the labels to the data samples of X_u . Therefore, it requires to solve the following optimization problem:

$$\min_{w,\rho,Y_u} \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + C'\sum_{i=l+1}^{n}\xi_i' \right) \quad (14)$$

subject to

$$\begin{aligned} &y_i(w^T\phi(x_i) - \rho) \geq \theta_i - \xi_i,\ i = 1, ..., l \\ &y_i(w^T\phi(x_i) - \rho) \geq \theta_i - \xi_i',\ i = l+1, ..., n \end{aligned} \quad (15)$$

where $Y_u = \{y_{l+1}, ..., y_n\}$ is a labeling assignment and $\theta_i = (1 - y_i)/2,\ i = l + 1, ..., n$.

B. The Algorithm for S2LM-OCSVM

To make *LM-OCSVM* and *S2LM-OCSVM* efficient for dealing with the imbalanced data sets, we use the different trade-off parameters for the negative and positive unlabeled data samples. The objective function of the optimization problem in Eq. (14) becomes the one in Eq. (16) while the constrains are still the same.

$$\min_{w,\rho,Y_U} \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + C'_+\sum_{y_i=1}\xi_i' + C'_-\sum_{y_i=-1}\xi_i' \right) \quad (16)$$

We now propose *Algorithm* 1 for *S2LM-OCSVM*.

Algorithm 1 Algorithm for S2LM-OCSVM

Input:
$X = X_l \cup X_u$
Parameters:
The trade-off parameters C, C'
num_+: number of unlabeled data samples assigned to the positive class
Output:
The predicted labels $\{y_{l+1}, ..., y_n\}$ for $\{x_{l+1}, ..., x_n\}$
Algorithm:
$(w, \rho, \xi) = solve\_lmocsvm(X_l, C, 0, 0)$
Calculate the output values $o_i = w^T\phi(x_i) - \rho,\ i = l+1, ..., n$ *regarding the current hyperplane.*
The num_+ *unlabeled data samples with the highest output values are assigned to the positive class.*
The remaining unlabeled data samples are assigned to the negative class.
$C'_- = 10^{-5}$; $C'_+ = 10^{-5} \times \frac{num_+}{u - num_+}$
while $\left(\left(C'_+ < C'\right) \parallel \left(C'_- < C'\right)\right)$
{
$(w, \rho, \xi, \xi') = solve\_lmocsvm(X, Y_u, C, C'_+, C'_-)$;
while $\left(\begin{array}{c} \exists r, s > l : (y_r \times y_s = -1)\ \&\ \left(\xi_r' > 0\right)\ \&\ \left(\xi_s' > 0\right) \\ \&\ \left(\xi_r' + \xi_s' > 1\right) \end{array} \right)$
{
$y_r = -y_r$; $y_s = -y_s$;
$(w, \rho, \xi, \xi') = solve\_lmocsvm(X, Y_u, C, C'_+, C'_-)$;
}
$C'_+ = min(2 \times C'_+, C')$;
$C'_- = min(2 \times C'_-, C')$;
}

C. The Rationale of S2LM-OCSVM

To persuasively show the rationale of *S2LM-OCSVM*, in *Theorem* 1 we prove that gradually the hyperplane and the labeling assignment become better and finally converges to

a stable configuration. We also prove in this section that *Algorithm* 1 must terminate after a finite number of iterations.

Theorem 1. *Let* $(w, \rho, Y_u, \xi, \xi^{'})$ *and* $(\overline{w}, \overline{\rho}, \overline{Y_u}, \overline{\xi}, \overline{\xi'})$ *be two consecutive optimal solutions in the loop 2 of Algorithm 1, we have:*

$$\begin{aligned} &\tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}\xi_i + C^{'}_{+}\sum_{y_i=1}\xi^{'}_i + C^{'}_{-}\sum_{y_i=-1}\xi^{'}_i \\ &> \tfrac{1}{2}\left\|\overline{w}\right\|^2 + C\sum_{i=1}^{l}\overline{\xi_i} + C^{'}_{+}\sum_{y_i=1}\overline{\xi_i} + C^{'}_{-}\sum_{y_i=-1}\overline{\xi^{'}_i} \end{aligned} \tag{17}$$

Proof: We first prove that after finding two unlabeled samples, e.g., x_r, x_s, for swapping their labels, the new configuration with the new labels for x_r, x_s and $\xi^{*^{'}}_r = max\left\{0, 1-\xi^{'}_r\right\}$, $\xi^{*^{'}}_s = max\left\{0, 1-\xi^{'}_s\right\}$ while the rest is kept the same is still a feasible solution of the optimization problem in Eq. (16).

Without loss of generality, we can assume that $y_r = 1$ and $y_s = -1$. We need to verify the constrains regarding x_r, x_s with the new labeling assignment. We have:

$$\begin{aligned} &y_r(w^T\phi(x_r) - \rho) = -\xi^{'}_r \\ &\rightarrow -y_r(w^T\phi(x_r) - \rho) = \xi^{'}_r \geq 1 - \xi^{*^{'}}_r \\ &y_s(w^T\phi(x_s) - \rho) = 1 - \xi_s \\ &\rightarrow -y_s(w^T\phi(x_s) - \rho) = \xi^{'}_s - 1 \geq -\xi^{*^{'}}_s \end{aligned} \tag{18}$$

Finally, we gain the conclusion as follows:

$$\begin{aligned} &\tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}\xi_i + ... + C^{'}_{+}\xi^{'}_r + C^{'}_{-}\xi^{'}_s + ... \\ &> \tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}\xi_i + ... + C^{'}_{+}max\left\{0, 1-\xi^{'}_s\right\} \\ &+C^{'}_{-}max\left\{0, 1-\xi^{'}_r\right\} + ... \\ &= \tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}\xi_i + ... + C^{'}_{+}\xi^{*^{'}}_s + C^{'}_{-}\xi^{*^{'}}_r + ... \\ &> \tfrac{1}{2}\left\|\overline{w}\right\|^2 + C\sum_{i=1}^{l}\overline{\xi_i} + C^{'}_{+}\sum_{y_i=1}\overline{\xi^{'}_i} + C^{'}_{-}\sum_{y_i=-1}\overline{\xi^{'}_i} \end{aligned} \tag{19}$$

■

Theorem 2. *Algorithm 1 terminates after a finite numbers of iterations.*

Proof: The number of labeling assignments Y_u is finite. According to *Theorem* 1, the objective function is decreased across the iterations. It concludes this proof. ■

V. Fuzzy Entropy Semi-supervised Large Margin One-class Support Vector Machine (FES2LM-OCSVM)

A. Optimization Problem

We need to deal with the following optimization problem:

$$\min_{w,\rho,Yu}\left(\frac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}V(o_i, y_i) + C^{'}\sum_{i=l+1}^{n}V(o_i, y_i)\right) \tag{20}$$

where $Y_U = \{y_{l+1}, ..., y_n\}$ is a labeling assignment, $o_i = \mathrm{w}^T\phi(x_i) - \rho$, the loss function is defined as $V(o_i, y_i) = max\{0, \theta_i - y_i o_i\}$, and C, $C^{'}$ are two parameters which stand for the trade-offs between the empirical losses of the labeled and unlabeled data and the general loss.

The above optimization problem means that we need to find out the optimal labeling assignment Y_U such that the margin for the whole data set, i.e. $X = X_l \cup X_u$, is maximized.

TABLE I
The expressions of the average loss and the fuzzy entropy according to the fuzzy membership u_i

	$x_i \in normal\ class$ or $y_i = 1$	$y_i = -1$
Fuzzy membership	u_i	$1 - u_i$
Average loss	$AV(o_i, u_i) = u_i V(o_i, 1) + (1-u_i)V(o_i, -1)$	
Fuzzy entropy	$S(u_i) = -u_i lnu_i - (1-u_i)ln(1-u_i)$	

B. Solution

For each unlabeled sample $x_i\,(l+1 \leq i \leq n)$, we introduce fuzzy membership u_i which stands for the possibility of that x_i belongs to the normal class or $y_i = 1$.

Given temperature $T > 0$, regarding sample x_i, we need to minimize the following extended loss function:

$$EV(o_i, u_i) = AV(o_i, u_i) - T \times S(u_i) \tag{21}$$

In the above extended loss function, we employ the entropy to encourage the purity of fuzzy partition. The reason is that to minimize $EV(o_i, u_i)$ when T becomes smaller, i.e. $T \rightarrow 0$, $S(u_i)$ is encouraged to be smaller which also means purer fuzzy partition. The temperature variable T is regarded as trade-off parameter which controls the trade-off between the average loss value and the purity of fuzzy partition. In experiment, the temperature T is led to approach 0.

The extended optimization problem is of the following form (referred to Table I):

$$\min_{w,\rho,U}\left(\begin{array}{l} \tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}V(o_i, y_i) \\ +C^{'}\sum_{i=l+1}^{n}(u_i V(o_i, 1) + (1-u_i)V(o_i, -1)) + \\ +C^{'}\sum_{i=l+1}^{n}(Tu_i lnu_i + T(1-u_i)ln(1-u_i)) \end{array}\right) \tag{22}$$

The constraint regarding the ratio of normal data in X_u is that in X_l can be interpreted as:

$$\frac{1}{u}\sum_{i=l+1}^{n}u_i = \frac{1}{l}\sum_{i=1}^{l}max\{0, y_i\} = r \tag{23}$$

We apply the alternative method to solve out the above optimization problem. The fuzzy membership array U and the optimal hypersphere are alternatively kept fixed. The temperature variable T is driven to approach 0.

1) Keep U fixed: : We come up with the following optimization problem:

$$\min_{w,\rho}\left(\begin{array}{l} \tfrac{1}{2}\left\|w\right\|^2 + C\sum_{i=1}^{l}V(o_i, y_i) \\ +C^{'}\sum_{i=l+1}^{n}(u_i V(o_i, 1) + (1-u_i)V(o_i, -1)) \end{array}\right) \tag{24}$$

The above optimization problem is that of the standard *LM-OCSVM*. Actually, we can transform it to another equivalent optimization problem as follows:

$$\min_{w,\rho} \left(\begin{array}{l} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i \\ +C^{'}\sum_{i=l+1}^{n} u_i\xi_i + C^{'}\sum_{i=l+1}^{n}(1-u_i)\xi_i^{'} \end{array} \right) \tag{25}$$

subject to:

$$\begin{array}{l} y_i(w^T\phi(x_i)-\rho)-\theta_i+\xi_i \geq 0,\ \xi_i \geq 0,\ i=1,...,l \\ w^T\phi(x_i)-\rho \geq -\xi_i,\ \xi_i \geq 0,\ i=l+1,...,n \\ w^T\phi(x_i)-\rho \leq -1+\xi_i^{'},\ \xi_i^{'} \geq 0,\ i=l+1,...,n \end{array} \tag{26}$$

The Lagrange function is of:

$$\begin{array}{l} L(w,\rho,\xi_i,\xi_i^{'},\alpha_i,\alpha_i^{'},\beta_i,\beta_i^{'}) \\ = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i + C^{'}\sum_{i=l+1}^{n} u_i\xi_i + C^{'}\sum_{i=l+1}^{n}(1-u_i)\xi_i^{'} \\ -\sum_{i=1}^{l}\alpha_i\left[y_i(w^T\phi(x_i)-\rho)-\theta_i+\xi_i\right] \\ -\sum_{i=l+1}^{n}\alpha_i\left[w^T\phi(x_i)-\rho+\xi_i\right] - \sum_{i=1}^{n}\beta_i\xi_i \\ +\sum_{i=l+1}^{n}\alpha_i^{'}\left[w^T\phi(x_i)-\rho+1-\xi_i^{'}\right] - \sum_{i=1}^{n}\beta_i^{'}\xi_i^{'} \end{array} \tag{27}$$

Setting the partial derivatives to 0, we gain:

$$\begin{array}{l} \frac{\partial L}{\partial w}=0 \rightarrow w=\sum_{i=1}^{l}\alpha_i y_i\phi(x_i)+\sum_{i=l+1}^{n}\alpha_i\phi(x_i)-\sum_{i=l+1}^{l}\alpha_i^{'}\phi(x_i) \\ \frac{\partial L}{\partial \rho}=0 \rightarrow \sum_{i=1}^{l}\alpha_i y_i+\sum_{i=l+1}^{n}\alpha_i-\sum_{i=l+1}^{l}\alpha_i^{'}=0 \\ \frac{\partial L}{\partial \xi_i}=0 \rightarrow \alpha_i+\beta_i=C,\ i=1,...,l \\ \frac{\partial L}{\partial \xi_i}=0 \rightarrow \alpha_i+\beta_i=C^{'}u_i,\ i=l+1,...,n \\ \frac{\partial L}{\partial \xi_i^{'}}=0 \rightarrow \alpha_i^{'}+\beta_i^{'}=C^{'}(1-u_i),\ i=l+1,...,n \end{array} \tag{28}$$

Substituting the above equations to the Lagrange function, we have:

$$\begin{array}{l} L(w,\rho,\xi_i,\xi_i^{'},\alpha_i,\alpha_i^{'},\beta_i,\beta_i^{'}) \\ = -\frac{1}{2}\|w\|^2 + \sum_{i=1}^{l}\alpha_i\theta_i + \sum_{i=l+1}^{n}\alpha_i^{'} \\ = -\frac{1}{2}\sum_{i=1}^{l+2ul+2u}\sum_{j=1}^{l+2ul+2u} y_i y_j K(x_i,x_j)\alpha_i\alpha_j + \sum_{i=1}^{l+2u}\alpha_i\theta_i \end{array} \tag{29}$$

where $\theta_i = 0, \forall i : y_i = 1$, $\theta_i = 1, \forall i : y_i = -1$. and $y_i = 1,\ i = l+1, ..., l+u$, $y_i = -1,\ i = l+u+1, ..., l+2u$, and $x_i = x_{i-u},\ i = l+u+1, ..., l+2u$.

We come up with the following optimization problem:

$$\min_{\alpha}\left(\frac{1}{2}\sum_{i=1}^{l+2ul+2u}\sum_{j=1}^{l+2ul+2u} y_i y_j K(x_i,x_j)\alpha_i\alpha_j - \sum_{i=1}^{l+2u}\alpha_i\theta_i\right) \tag{30}$$

subject to

$$\begin{array}{l} \sum_{i=1}^{l+2u}\alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C,\ i=1,...,l \\ 0 \leq \alpha_i \leq C^{'}u_i,\ i=l+1,...,l+u \\ 0 \leq \alpha_i \leq C^{'}(1-u_i),\ i=l+u+1,...,l+2u \end{array} \tag{31}$$

2) Keep w, ρ fixed: : By removing the constants, we achieve the following optimization problem:

$$\min_{U}\left(\sum_{i=l+1}^{n}\left(\begin{array}{l} u_iV_i+(1-u_i)V_i^{'} \\ +Tu_ilnu_i+T(1-u_i)ln(1-u_i) \end{array}\right)\right) \tag{32}$$

where $\sum_{i=l+1}^{n} u_i = ur$ and $V_i = V(o_i, 1)$, $V_i^{'} = V(o_i, -1)$.

The Lagrange function is of the following form:

$$\begin{array}{l} L(u,\lambda) \\ = \sum_{i=l+1}^{n}\left(u_iV_i+(1-u_i)V_i^{'}+Tu_ilnu_i+T(1-u_i)ln(1-u_i)\right) \\ -\lambda\left(\sum_{i=l+1}^{n}u_i - ur\right) \end{array} \tag{33}$$

Setting the partial derivatives to 0, we obtain:

$$\begin{array}{l} \frac{\partial L}{\partial u_i}=0 \\ \Rightarrow V_i - V_i^{'} + T(1+lnu_i) + T(-1-ln(1-u_i)) - \lambda = 0 \\ \Rightarrow ln\frac{1-u_i}{u_i} = \frac{V_i - V_i^{'} - \lambda}{T} \\ \Rightarrow u_i = \frac{1}{e^{\frac{V_i - V_i^{'} - \lambda}{T}}+1} \end{array} \tag{34}$$

Furthermore, we can evaluate $V_i - V_i^{'}$ as follows:

$$V_i - V_i^{'} = V(o_i,1) - V(o_i,-1) = max\{0,-o_i\} - max\{0,1+o_i\} \tag{35}$$

Let us define $\tau_i = max\{0,-o_i\} - max\{0,1+o_i\}$. We have:

$$\begin{array}{l} \tau_i = -o_i,\ o_i < -1 \\ \tau_i = -1-2o_i,\ 0 \leq o_i \leq -1 \\ \tau_i = -1-o_i,\ o_i > 0 \end{array} \tag{36}$$

Substituting the above equation to Equation (34), we can evaluate u_i as follows:

$$u_i = \frac{1}{e^{\frac{V_i - V_i^{'} - \lambda}{T}}+1} = \frac{1}{e^{\frac{\tau_i - \lambda}{T}}+1} \tag{37}$$

To determine λ, we use the constraint:

$$\sum_{i=l+1}^{n} u_i = \sum_{i=l+1}^{n}\frac{1}{e^{\frac{\tau_i-\lambda}{T}}+1} = ur \tag{38}$$

We define and investigate the following function:

$$f(\lambda) = \sum_{i=l+1}^{n}\frac{1}{e^{\frac{\tau_i-\lambda}{T}}+1} \tag{39}$$

The derivative of the above function is as follows:

$$f^{'}(\lambda) = \frac{1}{T}\sum_{i=l+1}^{n}\frac{e^{\frac{\tau_i-\lambda}{T}}}{\left(e^{\frac{\tau_i-\lambda}{T}}+1\right)^2} > 0 \tag{40}$$

It follows that the function $f(\lambda)$ is strictly increased. Moreover, we have:

$$\begin{array}{l} \lim_{\lambda\rightarrow\infty} f(\lambda) = \lim_{\lambda\rightarrow\infty}\sum_{i=l+1}^{n}\frac{1}{e^{\frac{\tau_i-\lambda}{T}}+1} = u \\ \lim_{\lambda\rightarrow-\infty} f(\lambda) = \lim_{\lambda\rightarrow-\infty}\sum_{i=l+1}^{n}\frac{1}{e^{\frac{\tau_i-\lambda}{T}}+1} = 0 \end{array} \tag{41}$$

It means that the equation in Eq. (38) has the unique solution λ_0 . To find λ_0, we employ the Newton-Raphson method. The rule for updating the fuzzy membership u_i becomes:

$$u_i = \frac{1}{e^{\frac{\tau_i - \lambda_0}{T}} + 1}, i = l+1, ..., n \quad (42)$$

C. The Overall Algorithm

We start with $T = 10$. For each T, we attempt to solve out the optimization problem in Eq. (22) by alternately keeping w, ρ and U fixed. *The KL-divergence* is used as stopping criterion for each iteration. To direct the local minimizer attained for each T to the global minimizer, T is led to approach 0. The detail of this algorithm is displayed as follows:

Algorithm 2 Algorithms for FES2LM-OCSVM

Initialize

$T = 10$, $\varepsilon = 0.0001$, $U = (r, r, ..., r)$

Execute

while $(T > \varepsilon)\{$

do$\{$

Keep U **fixed**

Calculate w, ρ, *and* $o_i = w^T \phi(x_i) - \rho$

Keep w, ρ **fixed**

$V = U$

Update the fuzzy partition array U

$\}$**while** $(D_{KL}(U, V) > \varepsilon)$

$T = \frac{T}{1.5}$

$\}$

where $D_{KL}(U, V) = \sum_{i=l+1}^{n} u_i ln\left(\frac{u_i}{v_i}\right)$.

VI. Experiment

A. The Experimental Data Sets

We conducted the experiment on 14 *benchmark data sets of UCI repository*. For novelty detection task, we made unbalanced the data sets by first appointing a class as the normal class and then recommending the rest as the abnormal class. In addition, the ratio of data in the normal class and abnormal class was kept by $10 : 1$. To enable the semi-supervised learning, with the current data set, we randomly hid the labels of 30% of data. The details of the data sets are given in Table II.

B. The Parameter Settings

To show the superiority of the proposed methods in the context of semi-supervised learning, we compared our proposed methods *LM-OCSVM, S2LM-OCSVM,* and *FES2LM-OCSVM* with other methods including *SVM, SVDD, OCSVM, TSVM* proposed in [7], and *two self-training methods* for *1-NN* and *SVM*.

For all kernel-based methods, *RBF kernel* given by $K(x, x') = e^{-\gamma \left\| x - x' \right\|^2}$ was used. The width of kernel γ was varied in the grid $\left\{2^{-15}, 2^{-13}, \ldots, 2^3, 2^5\right\}$. The trade-off parameter C was searched in the grid $\left\{2^{-15}, 2^{-13}, \ldots, 2^3, 2^5\right\}$.

TABLE II
THE DETAILS OF THE DATA SETS.

Data Set	#Positive	#Negative	#Dimension
Pima Indians Diabetes	500	50	8
Australian	307	30	14
Breast-cancer	239	23	10
Glass	76	7	9
Ionosphere	225	22	34
Liver Disorder	200	20	6
Sonar	97	9	60
Splice	517	51	60
Letter	543	54	16
Heart	120	12	13
SvmGuide 3	296	29	22
SvmGuide 1	2000	200	4
a7a	3918	391	122
Mushrooms	4208	420	112

The accuracy was measured by $acc = \frac{acc^+ + acc^-}{2}$ where $acc^+ = \%TP$ and $acc^- = \%TN$ are the accuracies on the positive and negative classes, respectively. This measure is correspondent to one-class classification problem since it inspires the high accuracies for both positive and negative classes to ensure the high accuracy for the entire data set. The cross-validation with five folds was used. Indeed, for each trial, we trained the methods on the four labeled portions of the four current folds and tested the trained models on the four unlabeled portions and the remaining fold. For each data set, we run the experiment five times and took average of the five accuracies.

C. The Experimental Results

The experimental results are shown in Tables III, IV and Figures 3, 4 . For each data set, to increase the readability of the tables, we emphasized in bold the methods that result in the highest accuracy and emphasized in double underline the runner-up methods.

As observed from the tables, in case only making comparison the supervised-based methods like *SVM*, *SVDD*, *OCSVM*, and *LM-OCSVM*, our proposed *LM-OCSVM* offers the highest accuracies for all experimental data sets except for the data set *SvmGuide 1*. It is reasonable because *LM-OCSVM* can efficiently take advantage from the negative data samples and its decision hyperplane is pushed as close as possible to the positive region in order to maximally reduce the chance of accepting the abnormal as normal data sample.

In comparison all methods, our proposed *FES2LM-OCSVM* is always the best except for the data set *Mushrooms*. Moreover, our proposed *S2LM-OCSVM* is the best in two cases and is the runner-up for 7 data sets. In our opinion, the fact that *FES2LM-OCSVM* produces the higher accuracies as compared to *S2LM-OCSVM* comes from the fact that the solution of *FES2LM-OCSVM* may be driven to the global minima when the temperature variable T is driven to 0 whereas that of *S2LM-OCSVM* suffers the local minima.

VII. Conclusion

In this paper, we first extend the model of *OCSVM* to enable the use of the negative data samples for classifying

TABLE III
THE EXPERIMENTAL RESULTS ON THE DATA SETS.

Data Set	SVM	SVDD	OCSVM	LM-OCSVM
Pima Indians Diabetes	59%	57%	68%	*70%*
Australian	80%	67%	83%	83%
Breast-cancer	*98%*	94%	93%	**99%**
Glass	79%	77%	75%	85%
Ionosphere	87%	86%	85%	*91%*
Liver Disorder	63%	57%	64%	66%
Sonar	72%	66%	65%	75%
Splice	70%	61%	60%	73%
Letter	95%	93%	91%	**98%**
Heart	88%	79%	71%	*90%*
SvmGuide 3	67%	59%	61%	70%
SvmGuide 1	**95%**	74%	77%	*94%*
a7a	79%	63%	71%	85%
Mushrooms	**100%**	**100%**	95%	**100%**

TABLE IV
THE EXPERIMENTAL RESULTS ON THE DATA SETS.

Data Set	TSVM	S2LM	FESS2LM	ST 1-NN	ST SVM
PMD	**71%**	69%	**71%**	61%	68%
Australian	82%	*85%*	**87%**	75%	84%
Breast-cancer	97%	96%	**99%**	*98%*	*98%*
Glass	85%	*87%*	**89%**	72%	75%
Ionosphere	**93%**	**93%**	**93%**	77%	82%
Liver Disorder	64%	*68%*	**71%**	65%	62%
Sonar	73%	*77%*	**83%**	72%	76%
Splice	71%	*76%*	**79%**	64%	71%
Letter	*97%*	**98%**	**98%**	84%	90%
Heart	86%	88%	**91%**	82%	87%
SvmGuide 3	68%	*72%*	**75%**	63%	60%
SvmGuide 1	92%	92%	**95%**	76%	82%
a7a	83%	*86%*	**89%**	69%	72%
Mushrooms	96%	95%	*98%*	91%	87%

the one-class data sets. To handle the data sets with the labeled portion jointly with the unlabeled portion which is very popular nowadays, we propose two semi-supervised learning methods. The experiment conducted on 14 *data sets of UCI repository* shows the superiority of the proposed methods as compared to the other methods.

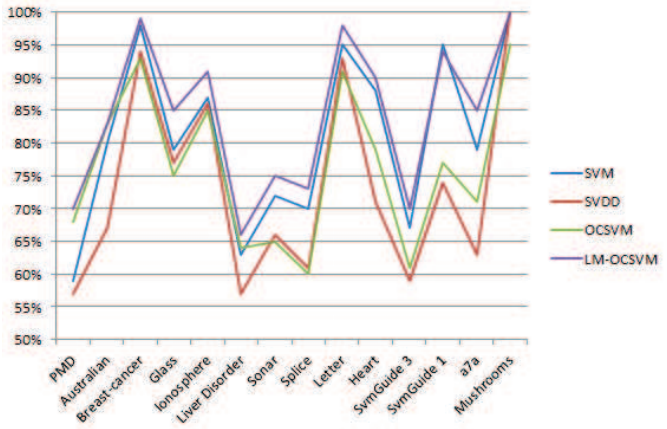

Fig. 3. Experimental results on the data sets for supervised methods.

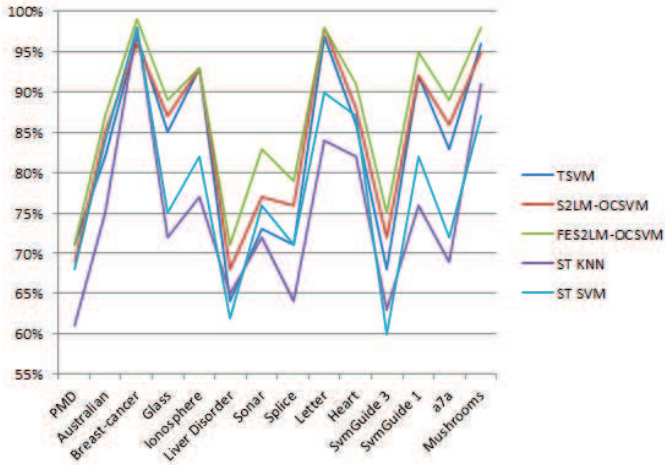

Fig. 4. Experimental results on the data sets for semi-supervised methods.

References

[1] O. Chapelle, V. Sindhwani, and S. Keerthi, "Optimization techniques for semi-supervised support vector machines," *Journal of Machine Learning Research*, vol. 9, pp. 203–233, Jun. 2008.

[2] C. Cortes and V. Vapnik, "Support-vector networks," in *Machine Learning*, 1995, pp. 273–297.

[3] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121–167, 1998.

[4] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[5] ——, *The Nature of Statistical Learning Theory*, 2nd ed. Springer, 1999.

[6] V. Vapnik and A. Sterin, "On Structural Risk Minimization or Overall Risk in a Problem of Pattern Recognition," *Automation and Remote Control*, vol. 10, no. 3, pp. 1495–1503, 1977.

[7] T. Joachims, "Transductive inference for text classification using support vector machines," in *International Conference on Machine Learning (ICML)*, Bled, Slowenien, 1999, pp. 200–209.

[8] T. De Bie and N. Cristianini, "Semi-supervised learning using semi-definite programming," in *Semi-supervised Learning*. Cambridge, MA: MIT Press, 2006.

[9] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, "Maximum margin clustering," in *Advances in Neural Information Processing Systems 17*. MIT Press, 2005, pp. 1537–1544.

[10] V. Sindhwani, S. Keerthi, and O. Chapelle, "Deterministic annealing for semi-supervised kernel machines," in *Proceedings of the 23rd international conference on Machine learning*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 841–848.

[11] K. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Advances in Neural Information Processing Systems*. MIT Press, 1998, pp. 368–374.

[12] O. Chapelle and A. Zien. (2005) Semi-Supervised Classification by Low Density Separation.

[13] O. Chapelle, M. Chi, and A. Zien, "A continuation method for semi-supervised svms," in *Proceedings of the 23rd international conference on Machine learning*, ser. ICML '06. New York, NY, USA: ACM, 2006, pp. 185–192.

[14] G. Fung and O. Mangasarian, "Semi-supervised support vector machines for unlabeled data classification," *Optimization Methods and Software*, vol. 15, pp. 29–44, 2001.

[15] R. Collobert, F. Sinz, J. Weston, L. Bottou, and T. Joachims, "Large scale transductive svms," *Journal of Machine Learning Research*, 2006.

[16] K. Lee, W. Kim, K. Lee, and D. Lee, "Density-induced support vector data description," *IEEE Transactions on Neural Networks*, vol. 18, no. 1, pp. 284–289, 2007.

[17] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[18] D. Tax and R. Duin, "Support vector domain description," *Pattern Recognition Letters*, vol. 20, pp. 1191–1199, 1999.