Pattern Recognition Letters 60–61 (2015) 48–56

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Modified criterion to select useful unlabeled data for improving semi-supervised support vector machines ☆,☆☆

Thanh-Binh Le [a], Sang-Woon Kim [a,*]

Department of Computer Engineering, Myongji University, Yongin 449-728, South Korea

ARTICLE INFO

Article history:
Received 20 November 2014
Available online 4 May 2015

Keywords:
Semi-supervised learning
Semi-supervised boosting
Support vector machines
Semi-supervised support vector machines

ABSTRACT

Recent studies have demonstrated that semi-supervised learning (SSL) approaches that use both labeled and unlabeled data are more effective and robust than those that use only labeled data. In SemiBoost, a boosting framework for SSL, a similarity based criterion is developed to select (and utilize) a small amount of useful unlabeled data. However, sometimes it does not work appropriately, particularly when the unlabeled data are near the boundary. In order to address this concern, in this paper the selection criterion is modified using the class-conditional probability in addition to the similarity: first, the criterion is decomposed into three terms of positive class term, negative class term, and unlabeled term; second, when computing the confidences of unlabeled data, using the conditional probability estimated, impacts of the three terms on the confidences are adjusted; third, some unlabeled data that have higher confidences are selected and, together with labeled data, used for re-training a supervised classifier. This select-and-train process is repeated until a termination condition is met. The experimental results, obtained using semi-supervised support vector machines (S3VMs) with benchmark data, demonstrate that the proposed algorithm can compensate for the shortcomings of the traditional S3VMs and, when compared with previous approaches, can achieve further improved results in terms of the classification accuracy.

1. Introduction

In semi-supervised learning (SSL) approaches [7,33], a large amount of unlabeled data (U), together with labeled data (L), is used to build better classifiers. That is, SSL exploits the examples of U in addition to the labeled counterparts in order to improve the performance of a classification task, which leads to a performance improvement in the supervised learning algorithms with a multitude of unlabeled data. However, it is also well known that using U is not always helpful in SSL algorithms. In particular, it is not guaranteed that adding U to the training data (T), i.e. $T = L \cup U$, leads to a situation in which the classification performance can be improved [2,19].

Therefore, in order to select (and utilize) a small amount of useful unlabeled data, various approaches have been proposed in the literature, including the procedures that are used in self-training [21,30] and co-training [5,32], simply recycled strategy (SRS) [16,20], incrementally reinforced strategy (IRS) [15], semi-supervised support vector machines with unlabeled instance selection (S3VM-us) [17], and other criteria used in active learning (AL) algorithms [9,11,13,24–26]. However, in AL, selected instances are useful when they are labeled, thus it is required to query their true class label from a human annotator [27]. Thus, the selected instances are generally the most uncertain instances, and, in turn, trying to predict their labels (to use them for training) will more likely result in a prediction error. From this, the selection approaches for SSL and AL are different.

Recently, a hybrid method (HYB) of SRS and IRS has been proposed [14]. In SRS, a subset of useful unlabeled data is selected and the process is repeated after returning the examples of the subset to unlabeled data pool. Meanwhile, in IRS, selection is performed in an incremental fashion; all the examples of the subset previously selected are included in the next selection. Consequently, certain kinds of the selected examples, for which confidence levels are evaluated appropriately but pseudo-labels are predicted incorrectly, or vice versa, continue to be used in the following iterations. This means that the learning leads to poor classification performance. In order to remedy this problem, in HYB, SRS is embedded into IRS while repeating the process (see Figs. 1 and 2 in [14]).

Also, the approaches can be categorized into two groups, graph-based approach and confidence-based approach, by considering whether or not they can utilize the pertinent features of the graph. Graph-based SSL treats both L and U examples as vertices (nodes)

☆ This paper has been recommended for acceptance by S. Sarkar.

☆☆ A preliminary version [16] of this paper was presented at ICPRAM2014, the 3rd International Conference on Pattern Recognition Applications and Methods, Angers, France, in 6–8 March 2014.

* Corresponding author. Tel.: +82 31 330 6437; fax: +82 31 335 9998.
E-mail address: kimsw@mju.ac.kr (S.-W. Kim).

http://dx.doi.org/10.1016/j.patrec.2015.04.011
0167-8655/

in a graph and builds pairwise edges between these vertices which are weighed by the affinities (i.e., similarities) between the corresponding example pairs. Thus, in the graph-based approach, the U points are selected based on their connection on graph or hypergraph without considering the usefulness of those points on training data. Various applications, including Laplacian regularized D-optimal design (LapRDD) [11], multi-view metric learning [31], medical image segmentation [8], etc., have been reported in the literature.

Meanwhile, in the confidence-based approach, the confidence levels are measured for each example on U data. Choosing top confident examples will make sure that the helpful examples are included in the training data. Thus, in order to select a small amount of useful U data, various selection criteria have been proposed in the literature. One criterion, for example, is based on the prediction by a base classifier and the similarity between pairwise training examples. Since the criterion is only concerned with the distance information among the examples, however, sometimes it does not work appropriately, particularly when the U examples are near the boundary. In order to address this concern, a method of training semi-supervised support vector machines (S3VMs) using a selection criterion is investigated; this method is a modified version of that used in SemiBoost [20].

In particular, in SemiBoost, the confidence value of $x_i \in U$ is computed using two quantities, named p_i and q_i, which are calculated using the pairwise similarity between x_i and other U and L examples. The p_i and q_i can be used to guide the selection at each iteration using differences in their values ($|p_i - q_i|$) as well as to predict the pseudo class label using a signum function ($sign(p_i - q_i)$). Therefore, the difference in values between p_i and q_i, $p_i - q_i$, should be measured first as follows:

$$p_i - q_i = X_i^+ - X_i^- + X_i^u, \tag{1}$$

where X_i^+ and X_i^- denote respectively the measurements obtained with L^+ and L^- (the examples of the positive and negative classes of L), while X_i^u denotes the measurement obtained with U. However, when x_i is near the boundary between the two classes, the first two terms have similar values and, in turn, the result of (1) depends on U only. From this observation, it can be noted that L data do not have an influence in selecting x_i as well as predicting its label; consequently, the confidence value obtained might be inappropriate for selecting useful data.

In order to address this issue, a modified criterion that minimizes the errors in estimating the confidence value is investigated in this paper; the criterion in (1) is modified by taking a balance among the three terms. This balance can be achieved with reducing the impact of X_i^u using an *uncertainty* level of x_i. That is, the class-conditional probability estimated (p_E) represents the likelihood of predicting the label of the example, i.e. a certainty level in prediction. Thus, $1 - p_E$ represents the uncertainty level in the prediction, which is reflecting the incompleteness in measuring the confidence value. The main contribution of this paper is the demonstration that the classification accuracy of S3VMs can be improved using a modified criterion when selecting unlabeled data and predicting their pseudo-labels. Furthermore, a comparison of the accuracies of S3VMs between the proposed method and traditional algorithms is performed empirically. In particular, some critical questions concerning the strategies employed in the present work are investigated, including *why the modified criterion is better than the original.*

The remainder of the paper is organized as follows. In Section 2, after providing a brief introduction to S3VMs, an explanation for the use of selection criterion in SemiBoost is provided. In Section 3, a modified criterion of selecting a small amount of useful unlabeled data for improving S3VMs through utilizing the class-conditional probability is presented. In Section 4, some illustrative examples for comparing the two criteria and a discussion on the results are presented. In Section 5, the experimental setup and results obtained using certain real-world data are presented. Finally, in Section 6, the concluding remarks and limitations that deserve further study are presented.

2. Related work

2.1. Semi-supervised support vector machines (S3VM)

A set of n_l labeled object pairs ($L = \{(x_1, y_1), \ldots, (x_{n_l}, y_{n_l})\}$, $x_i \in \mathbb{R}^d$, and $y_i \in \mathbb{R}$) and a set of n_u unlabeled objects ($U = \{x_1, \ldots, x_{n_u}\}$ and $x_j \in \mathbb{R}^d$) are considered. Referring to [28], SVMs have a decision function $f_\theta(\cdot)$, which is defined as $f_\theta(x) = w \cdot \Phi(x) + b$, where $\theta = (w, b)$ denotes the parameters of the classifier model, $w \in \mathbb{R}^d$ is a vector that determines the orientation of the discriminating hyperplane, and $b \in \mathbb{R}$ is a bias constant such that $b/\|w\|$ represents the distance between the hyperplane and origin. Also, $\Phi : \mathbb{R}^d \to F$ is a nonlinear feature mapping function, which is often implemented implicitly using the kernel trick.

S3VMs are an expansion of SVMs using an SSL strategy [3,12]. When denoting η_i (and η_j) as the loss for x_i (and x_j), the quadratic programming formulation for S3VM is defined as follows:

$$\begin{aligned} min \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n_l}\eta_i + C^*\sum_{j=1}^{n_u}\eta_j \\ s.t. \quad & y_i f_\theta(x_i) + \eta_i \geq 1, \quad i = 1, \ldots, n_l, \\ & |f_\theta(x_j)| \geq 1 - \eta_j, \quad j = 1, \ldots, n_u, \end{aligned} \tag{2}$$

where C (and C^*) is the penalty regularization parameter.

2.2. SemiBoost

The goal of SemiBoost, which is a boosting framework for SSL, is to iteratively improve the performance of a supervised learning algorithm ($\mathcal{A}$) by regarding it as a black box, using U and pairwise similarity. In order to follow the boosting idea, SemiBoost optimizes performance through minimizing the objective loss function defined as (see Proposition 2 in [20]):

$$\overline{F_1} \leq \sum_{i=1}^{n_u}(p_i + q_i)(e^{2\alpha} + e^{-2\alpha} - 1) - \sum_{i=1}^{n_u} 2\alpha h_i(p_i - q_i), \tag{3}$$

where $h_i(\equiv h(x_i))$ is the classifier learned by $\mathcal{A}$ at the iteration, α is the weight for combining h_i's, and

$$\begin{aligned} p_i &= \sum_{j=1}^{n_l} S_{i,j}^{ul} e^{-2H_i}\delta(y_j, 1) + \frac{C}{2}\sum_{j=1}^{n_u} S_{i,j}^{uu} e^{H_j - H_i}, \\ q_i &= \sum_{j=1}^{n_l} S_{i,j}^{ul} e^{2H_i}\delta(y_j, -1) + \frac{C}{2}\sum_{j=1}^{n_u} S_{i,j}^{uu} e^{H_i - H_j}. \end{aligned} \tag{4}$$

Here, $H_i(\equiv H(x_i))$ denotes the final combined classifier and S denotes the pairwise similarity. For all x_i and x_j of the training set, for example, S can be computed as follows: $S(i, j) = exp(-\|x_i - x_j\|_2^2/\sigma^2)$, where σ is the scale parameter controlling the spread of the function. In addition, S^{lu} (and S^{uu}) denotes the $n_l \times n_u$ (and $n_u \times n_u$) submatrix of S. Also, S^{ul} and S^{ll} can be defined correspondingly; the constant C, which is different from that in (2) and computed using $C = |L|/|U| = n_l/n_u$, is introduced to weight the importance between L and U; and $\delta(a, b) = 1$ when $a = b$ and 0 otherwise.

From (4), by substituting $L^+ \equiv \{(x_i, y_i)|y_i = +1, i = 1, \ldots, n_l^+\}$ and $L^- \equiv \{(x_i, y_i)|y_i = -1, i = 1, \ldots, n_l^-\}$ as the L examples in class $\{+1\}$ and class $\{-1\}$, respectively, the difference in values between p_i and q_i, $p_i - q_i$, can be measured as follows: