Incremental Learning with Support Vector Machines and Fuzzy Set Theory

Yu-Ming Chuang\textsuperscript{1} and Cha-Hwa Lin\textsuperscript{2*}
\textsuperscript{1}Department of Computer Science and Engineering
\textsuperscript{2}Department of Computer Science and Engineering and College of General Education
National Sun Yat-Sen University, Kaohsiung 80424, Taiwan
chlin@cse.nsysu.edu.tw

Abstract

Over the past few years, a considerable number of studies have been made on Support Vector Machines (SVMs) in many domains to improve classification or prediction. However, SVMs request high computational time and memory when the datasets are large. Although incremental learning techniques are viewed as one possible solution developed to reduce the computation complexity of the scalability problem, few studies have considered that some examples close to the decision hyperplane other than support vectors (SVs) might contribute to the training process. Consequently, we propose a novel algorithm by improving Syed's incremental learning method based on fuzzy set theory. At each learning step, SVs and potential informative examples, called candidate examples (CEs), are added to the next incremental learning step. We expect to achieve better accuracy and less execution time than other methods. In this ongoing study, the proposed algorithm would be investigated on five standard machine learning benchmark datasets to demonstrate the effectiveness of the method.

1 Introduction

In the last few years, many articles have been devoted to the study on support vector machines (SVMs) which have good performance as the classifiers in pattern recognition and text classification. They also have been applied to categorize Spam [3][6], and significantly outperform the conventional methods (e.g., Boosting, Rocchio, Bayes, and k-Nearest Neighbors). At the same time, there are complicated classification tasks to be solved in many other fields, such as bioinformatics, image processing, social studies, and stock market.

Vapnik [11] introduced SVMs in 1995. SVMs have strong theoretical groundwork and statistical foundation, which are based on the structural risk minimization principle [11] from computational learning theory. By most informatory data examples, called support vectors (SVs), the decision hyperplane is found with the lowest true error. The true error of the hyperplane is the probability that the hyperplane will make an error on an unknown and randomly picked test data.

Some experimental results show that SVMs do not require any parameter tuning, since they can determine good parameter setting automatically [6]. Compared to the customary learning algorithms, SVMs handle large feature spaces without feature selection. As real-word datasets expand in size, there is a need to design some novel learning algorithms to train increasing instances. Incremental (or active) learning procedures are one possible paradigm for reducing the sample complexity of large-scale learning task.

There are many different approaches of incremental learning to shorten the training time [1][7][10]. Processing data in several steps instead of learning from randomly selected data, these methods would use less memory space and saving more computational time. The aim is to choose a part of samples for training and keep the performance as using total training instances.

In the next section, we introduce the related work about the incremental learning presented by the researchers and the fuzzy set theory. Section 3 presents our proposed mechanism for incremental learning. Section 4 describes the design of experiment. Finally, we conclude in Section 5 with some discussion of our method, and remark on future research directions.

2 Related Research

2.1 Support Vector Machine

Support vector machines are usually used to solve two-class classification problems by training the datasets and obtaining the optimal separating hyperplane. SVMs are also applied to multi-class classification and regression problems.

Suppose we are given $k$ patterns $(x_1, y_1), \ldots, (x_k, y_k)$. The label $y_i \in \{-1, 1\}$ and the associated

---

\textsuperscript{*}The corresponding author.
vector \( \mathbf{x}_j \in \mathbb{R}^d \) where \( d \) is the dimension of the vector. Let \( X \) be the space of patterns, \( Y \) be the space of labels. The decision function is in form of \( \text{sign}(\mathbf{w} \cdot \mathbf{x} + b) \), where \( \mathbf{w} \cdot \mathbf{x} \) means the inner product of \( \mathbf{w} \) and \( \mathbf{x} \). If all \( k \) patterns are linearly separable, we can find a decision function like

\[
y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad i = 1, 2, ..., k
\]

Training samples that satisfy the equality are called support vectors. These two-class support vectors belong to different hyperplanes. SVMs maximize the margin when the norm of the weight vector \( \|\mathbf{w}\| \) is minimum. A margin is defined as the distance between the two hyperplanes. Since the optimization problem is difficult to handle numerically, Lagrange multiplies are used to translate the problem into an equivalent quadratic optimization problem [11]. Thus, we should maximize (2) subject to (3)

\[
Q(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j)
\]

\[
\sum_{i=1}^{k} \alpha_i y_i = 0
\]

When the training set is not linearly separable, slack variable \( \xi_i \) can be added to allow misclassification of noisy or difficult samples

\[
y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad \text{for all} \quad i
\]

To learn the datasets that are nonlinearly separable, SVMs make use of kernel function \( K(\mathbf{x}_j, \mathbf{x}_j) \). The kernel function satisfies Mercer’s Theorem [11]. This means that we can calculate the inner product of the vectors \( \mathbf{x}_j \) and \( \mathbf{x}_j \) after they have been mapped into another feature space by a non-linear mapping \( \Phi \):

\[
K(\mathbf{x}_j, \mathbf{x}_j) = \Phi(\mathbf{x}_j) \cdot \Phi(\mathbf{x}_j)
\]

2.2 Incremental Learning

Incremental learning algorithms for support vector machines have received considerable attention over past few years. The discussion of some existing incremental learning is presented as follows.

Syed et al. [10] presented an incremental learning procedure by partitioning the whole database into subsets that fits into the main memory. Then training SVM classifier incrementally with the partitions. The training would preserve only the support vectors at each incremental step, and add them to the training set for the next step.

Moreover, most current strategies of active learning perform the measurement of proximity to the separating hyperplanes. Mitra et al. [7] developed an active support vector learning algorithm, which is a probabilistic generalization of purely margin-based method. The likelihood of an sample being an support vector is estimated using a combination of two factors: The margin of the particular sample with respect to the current hyperplane and the degree of confidence that the current set of support vectors provides the actual support vector.

Cheng [1] presented an improved incremental training method for SVMs using active query. In active query, they assign a weight to each example according to its distance to the separating hyperplane and its confidence factor. The confidence factor is calculated from the error upper bound of the SVMs to indicate the closeness of the current hyperplane to the optimal hyperplane. The initial set of training examples is collected by dividing the whole training examples into groups and applying the k-means clustering algorithm.

2.3 Fuzzy Set Theory

There are many ambiguous words in our conversation, such as “Tom is tall” and “It is very cold”. We often describe circumstances in such imprecise statements. Fuzzy set [12], which is a mathematical tool used to model an uncertain, incomplete concept. The development of fuzzy set theory provides a lot of applications in the daily life, such as air-conditions, washing machines, and automobiles.

Traditionally, the crisp set is a set with a boundary. For example, a set \( A \) of real numbers less than or equal to 12 can be presented as \( A = \{x \mid x \leq 12\} \). If \( x \) is less than or equal to 12, then \( x \) is belong to the set \( A \) which has a certain boundary 12. The ordinary subset \( A \) of set \( X \) can be thought of a function from \( X \) to the 2-element set \{0, 1\} [9], where the set \( X \) is called universe of discourse.

\[
A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}
\]

where \( A(x) \) is called characteristic function of the set \( A \). Crisp sets are also referred to as ordinary sets, classical sets, non-fuzzy sets, or just sets.
The definition of fuzzy sets is usually by the concept of membership function. If \( X \) is a collection of objects denoted generically by \( x \), then a fuzzy set \( A \) in \( X \) is defined as a set of ordered pairs [5]:

\[
A = \{(x, \mu_A(x)) \mid x \in X\} \tag{7}
\]

Where \( \mu_A(x) \) is called the membership function of the fuzzy set \( A \). The value of \( \mu_A(x) \) also means the degree of membership of \( x \) in \( A \). In contrast, elements of fuzzy sets are in varying degrees of membership function. The membership function maps each element of \( X \) to a value between 0 and 1. Obviously, a fuzzy set is a simple extension of a crisp set in which the membership function is permitted to have any values between 0 and 1. If the value of the membership function \( \mu_A(x) \) is restricted to either 0 or 1, then \( A \) is reduced to a crisp set and \( \mu_A(x) \) is the characteristic function of \( A \).

### 3 The Proposed Method

Some studies showed that the instances which near the decision hyperplane are informative. Thus, it might be possible to find some data instead of the closest ones. Fuzzy set theory is helpful for us to take these instances in consideration, and we can determine the optimal separating hyperplane more effectively. Hence, we modify Syed’s incremental learning algorithm based on fuzzy set theory.

Syed’s method partitioned a huge database that each partition would fit into the main memory. The training would preserve only the support vectors at each incremental step, and add them to the training set for the next step. The model obtained by this method should be the same or similar to what would have been obtained using all the data to train. Their technique looked at the examples only once to determine if they will become support vectors. Once discarded, the vectors are not considered again.

#### 3.1 Candidate Examples

The training samples which satisfy (1) are called support vectors. By fuzzy set theory, some examples outside the two parallel hyperplanes forming the maximum margin of the SVM interest us. We think these non-support vectors, called candidate examples (CEs), are useful in training steps.

According to some factor, CEs can be fairly selected. The factor is a type of membership function (Figure 2), which is exploited by fuzzy set theory. Using membership function not only helps us find the relative samples but also be aware of the data distribution.

Let \( A \) be the set of CEs. The membership function on set \( A \) is defined as

\[
\mu_A(x) = \begin{cases} 
1 & \text{if } x > d \\
1 + e^{-a(x-c)} & \text{if } x \leq d 
\end{cases} \tag{8}
\]

where \( a \) determines the slope at the crossover point, \( x = c \), \( c \) shifts the function right or left, and \( d \) is half of the maximum margin of the SVM.

Because the examples which are far away from the separating hyperplane might hardly be chosen as CEs, we set the parameter \( a < 0 \) to obtain the decreasing function in the region \( x > d \). The value of \( d \) is usually scaled to 1 according to the distance between the separating hyperplane and each hyperplane which forms the maximum margin of the SVM. Moreover, the

![Figure 1. The incremental learning procedure with candidate examples](image-url)
Figure 2. The membership function of CEs with $m = 8$

parameter $c$ must be greater than $d$ to present the distribution of the CEs. If $c \leq d$, CEs might be non-informative for training steps or regarded as SVs. Both the value of $a$ and $c$ are set by the experimental processes.

3.2 The Improved Learning Algorithm

Hence, we follow the above idea and consider more data than support vectors (Figure 1). The basic concept of the proposed learning algorithm is described as follows.

Step 1: Partition the initial unlabeled training samples $x_1, x_2, \ldots, x_k$ into $m$ subsets of equal size.

Step 2: Randomly select a subset, and train SVM classifier with the subset of data. Select $l$ examples from the remaining data which are called CEs by fuzzy set theory.

Step 3: Add the SVs and the CEs to the training set for the next step until all partitions are used.

At step 2, we randomly divide the initial datasets into $m$ groups. The reason is when there is a large number of training examples, it is time-consuming to find the CEs.

4 Experimental Design

In the experiments, our method would use three kernels: linear SVM, Gaussian RBF SVM and polynomial SVM in comparison with five existing methods.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No. of attributes</th>
<th>Total instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancer</td>
<td>9</td>
<td>699</td>
</tr>
<tr>
<td>Diabetes</td>
<td>8</td>
<td>768</td>
</tr>
<tr>
<td>German</td>
<td>24</td>
<td>1000</td>
</tr>
<tr>
<td>Heart</td>
<td>13</td>
<td>270</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>34</td>
<td>351</td>
</tr>
</tbody>
</table>

4.1 Datasets

There are available public domain datasets provided by some organization, such as Ohsumed corpus (ftp://medir.ohsu.edu/pub/OHSUMED/) and StatLib (http://lib.stat.cmu.edu/datasets/). We would carry out the empirical study by the datasets obtained from the UCI machine learning repository [8]. The characteristics of the five datasets selected are described in Table 1.

4.2 Evaluation Measure

We use the same set of parameters for all methods for comparison purposes. We set regularization constant $C = 1$ and the variance of the Gaussian RBF $\sigma^2 = 1$. Our method would be compared with five existing algorithms: BatchSVM, IncrSVM, SubsetSVM, ActiveQuery, and LibSVM. BatchSVM [2] uses all the training examples to train the SVM, it needs numerous data. At each incremental step, IncrSVM randomly chooses the training examples. SubsetSVM [10] is an incremental learning method by partitioning the entire training dataset into several subsets and only adding the SVs to the next step. ActiveQuery [1] selects the initial training examples by k-means clustering instead of random selection, and actively queries the informative examples based on confidence factor. LibSVM [4] is a library for support vector machines with high accuracy.

5 Conclusion

Either use only SVs in the training process might lose some valuable data or use all training examples might waste too much time. In this study, the proposed incremental learning algorithm incorporates fuzzy set theory into SVM classifier to solve the above problems. We consider not only the SVs are important, also some of the remaining examples are useful to train the SVMs.
To find candidate examples, a membership function would be defined to express the degree of importance of the potential examples and could be used to filter the training examples in the incremental learning steps. It is possible to adjust the parameters of the membership function to increase the candidate instances. The proposed method could handle classification problems and present a novel view to recognize the useful examples in addition to SVs of SVMs by fuzzy set theory.

At present, using fuzzy set theory at each incremental step might spend more time, it is possible to find an alternative procedure to reduce the time complexity in the future.

References


