Development of Co-training Support Vector Machine Model for Semi-supervised Classification

Yinghao Chen, Tianhong Pan*, Shan Chen
School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China

Abstract: With the advancement of data processing technology, it is a significance task for machine learning to handle massive amounts of data. The traditional classification method is a supervised learning method, which requires a large number of labeled samples. But it is difficult to achieve. In this paper, a semi-supervised learning algorithm combining co-training with support vector machine (SVM) classification algorithm is introduced. Through an iterative learning procedure, the final new labeled data set can be determined based on unlabeled data set by training two SVM classifiers. Examples are provided for performance evaluation of the proposed method with detailed comparative studies to the traditional SVM and genetic algorithm SVM.

Keywords: Semi-supervised learning; Support vector machine; Co-training strategy

1 Introduction

In machine learning area, a model trained based on labeled data is known as supervised learning, relatively, unsupervised learning only incorporates unlabeled data [1]. As we all known, in many application areas of machine learning and pattern recognition, one has to face the serious problems of massive data with only a small part of which was labeled [2]. In this situation, the performance of classifier cannot be well guaranteed due to only a limited number of labeled samples are used for model training. Therefore, semi-supervised learning (SSL) has received a high degree of attention [3]. Distinguished from both supervised and unsupervised learning methods, SSL can take advantage of the knowledge of labeled and unlabeled samples to train [4-6].

Traditional SSL methods include generative models, self-training, co-training paradigm, etc. Among all those semi-supervised methods, co-training has its unique advantage, different from the self-training, it avoids the disadvantage that wrong results may affect future learning accuracy during the self-training process and has no limit on the structure of the data model [7]. Due to its simple structure, effective performance and easy understanding, co-training has been applied in many fields such as natural language processing [8], content based image retrieval [9, 10], etc.

In this paper, co-training strategy is used and combined with the SVM algorithm for classifier design. In this algorithm, unlabeled samples are categorized by two different

* Corresponding author: Tianhong Pan (thpan@ujs.edu.cn).

classifiers to expand the labeled samples. And then the expanded labeled samples are used to improve the performance of the classifier. Experiment results show that this algorithm can effectively inculcate unlabeled data to improve the performance of SVM, especially the number of labeled data is quite few.

This paper has been organized as follows: Section 2 briefly introduce the basic SVM algorithm. In section 3, the detailed description of the semi-supervised SVM classifier model is given. Section 4 provides case studies on UCI dataset. Finally, conclusions are made.

2 Classification Based on Support Vector Machine

Support vector machine proposed by V. Vapnik [11] is a learning machine based on VC theory and structural risk minimization (SRM) principle of statistical theory [12]. It is based on limited sample information to seek a compromise between the complexity of the model and the learning ability to get the best outreach [13].

SVM classification is developed form the optimal hyperplane in the case of linear separable conditions, it controls the capacity of the machine by maximizing the sorting interval to implement the SRM principle [14]. For the two types of linear separable problems, the optimal classification surface can be constructed directly, so that all the vectors in the sample set meet the following conditions:

1. In order to ensure that the experience risk is minimized, all samples can be correctly divided by a hyperplane;
2. The distance of the nearest heterogeneous vector from the hyperplane is the largest from the hyperplane. That is, the largest classification interval. Thus minimizing the expected risk. It is actually a secondary planning problem. The formula for obtaining the optimal decision function is as follows:

$$F(x) = \operatorname{sgn}\left\{ \sum_{i=1}^{L} y_i a_i K(x_i, x) + b \right\} \tag{1}$$

Among them, $K(.,.)$ is kernel function, $\operatorname{sgn}(.)$ is symbolic function, L is the number of training samples. Coefficient a_i is the solution of quadratic optimization problem:

$$\begin{aligned} &\min_a \frac{1}{2} a^T Q a - e^T a \\ &y^T a = 0 \\ &0 \le a_i < \infty, i = 1, \ldots, L \end{aligned} \tag{2}$$

Generalized optimal classification surface $0 \le a_i < C$. Here, *Q* is a semi-definite matrix of $L \times L$, $Q_{ij} = y_i y_j K\left(x_i, x_j\right)$, *e* is a column vector with all element being 1, *C* is error penalty factor for optimal classification surface.

For linear problems, $K(.,.)$ is not product of two vectors. For non-linear problems, SVM maps the input vector to a high-dimensional feature space H by pre-selected kernel function, then the optimal classification hyperplane is constructed in H. After introducing the non-linear mapping Φ , the linear indivisible problem of the original low-dimensional space is transformed into a linear or almost linear separable problem in high-dimensional space, then the classification problem is transformed into the feature space. Kernel function $K(x, x_i) = \langle \Phi(x), \Phi(x_i) \rangle$ deals with Φ as a whole, so that all operations are still in the original space.

Algorithm: co-training SVM

Input: labeled example set *L* (consists by variables x_l and their correspond label *y*), unlabeled examples set *U* (consists only x_u), maximum number of learning iterations T.

```
Process:
Divide $x_l$ into two parts respectively: $L_1 : \{X^1, Y\}$ ; $L_2 : \{X^2, Y\}$ .
Repeat for T rounds:
For  $j \in \{1,2\}$ do
    $h_j = SVM\left(L_j\right)$
    For each  $x_u \in U$ do
              $u_j = h_j\left(x_u\right)$
      If there exist $u_1 = u_2 = u'$
      Then $U = U - x_u$
    $\pi = \{\left(x_u, u'\right)\}$
      Else    $U = U$
              $\pi = \varnothing$
      End for
$L_{3-j} = L_{3-j} \cup \pi$
If U is not changing
End of repeat
```

Output: new labeled dataset *L*.

One of the main contents of designing SVM is to select kernel functions and kernel parameters. Vapnik and others found, the key factors that affect SVM performance are kernel parameters and penalty factor *C*, rather than the type of kernel function. In this paper, the radial basis function SVM is selected, then chose the excellent kernel parameter σ^2 and penalty factor *C*.

3 Semi-supervised SVM for classification

The co-training method is proposed by Blum and Mitchell in 1988. It trains two learners separately on two sufficient and redundant views and use the predictions of one leaner on unlabeled examples to augment the training set of the other [15]. However, the requirements of sufficient and redundant views are hard to conform, than S. Goldman and Y Zhou proposed an adapted co-training method which does not require sufficient and redundant views [16], but it is time consuming due to the need for ten cross-validations to determine the confidence level of unlabeled samples. For this problem, Tri-training algorithm was proposed by Z H Zhou and M Li, which require neither the sufficient and redundant views nor different leaners as the previous co-training method [17]. After then, Wang and Zhou proved that the performance of learners can be enhanced if there is a big difference between them [18]. Due to the advantages above, the co-training strategy enhance the ability of dealing with classification issues.

3.1 Co-training SVM Model for Classification Developed

In this paper, an algorithm combining co-training method with SVM is proposed. The procedure of co-training SVM algorithm is introduced as follow. Let $L=\{X,Y\}=\left\{(x_1,y_1),(x_2,y_2),...,\left(x_{|L|},y_{|L|}\right)\right\}$ denote the labeled data set, where x_i is the *i*th sample, y_i is the real-label and $|L|$ is the number of labeled data. *U* is unlabeled data.

First, *X* is divided into two parts x_i^1 and x_i^2. Here, two labeled data sets $L_1:\{X^1,Y\}$ and $L_2:\{X^2,Y\}$ represent training samples and test samples respectively. Then two SVM classifiers h_1 and h_2 can be trained by using the labeled set L_1. Then *U* is classified by two classifiers to obtain u_1 and u_2 respectively. Comparing u_1 and u_2, u' will record the same mark results. If the loop termination condition is not reached, u' is added to *L*, otherwise exit the loop. At last, the final training sample set is used to train the classifier. A classifier with better classification result in h_1 and h_2 is selected to test the test sample. The detailed procedures of co-training SVM are listed below.

In this algorithm, an unlabeled sample with the same label result of two classifiers is considered to have higher confidence level, so they can be added into *L* to expand the training sample set. For classifiers h_1 and h_2, although they are all SVMs classifiers, but in actual operation, the parameters of them are different to make a difference between them.

3.2 Case study

In this paper, all the experimental data is selected from the UCI dataset which is a commonly used standard test data set. In order to verify the effectiveness of the algorithm, all the samples are labeled. The structure of the experiment data is shown in Table 1.

Table 1. Experimental Data

Dataset	Number of Samples	Number of Attributes	Number of Categories
Yeast	1481(1394)	8	10(4)
Ionosphere	351	24	2
Iris	150	4	3

For convenience, in dataset Yeast, categories EXC, VAC, POX, and ERL are deleted because the number of them are less. In addition, ME1, ME2 and ME3 are treated as one class. So the values in brackets in Table 1 are really used. Other data sets remain unchanged.

In order to obtain a confident result, a total of 3 simulation times are carried and averaged the results. The traditional SVM algorithm is taken from the toolbox of OSU-SVM3.0 in MATLAB and the RBF kernel function is selected, parameters are default. Among them, penalty factor $C = 1$ and $\sigma^2 = 1$. In order to prevent the degradation of the algorithm into self-training, one of the classifiers is used GA-SVM. In each dataset, 50% samples are used as training set and the rest are used as test set. In training phase, different proportions of the labeled sample in training set are selected, and the rest as unlabeled sample. In the process, the algorithm is used to extend the labeled sample, and then a classifier is trained by the extended labeled ample. Finally, the classifier is used to classify the test set. Table 2 shows the results of 30% labeled sample in training set.

It is worth mentioning that, in dealing with multi-classification SVM problem, the DAG-SVMs is used to organize the classifiers, the five categories of decision-making process are shown in Fig.1.

Table 2. The comparison of classification results of 30% labeled sample

Algorithm / Accuracy	SVM	GA-SVM	CO-SVM
Yeast	0.73	0.79	0.86
Ionosphere	0.76	0.84	0.89
Iris	0.82	0.89	0.94

In the experiment, the co-training SVM algorithm is proposed to implement the number of labeled sample. Compared with other supervised algorithms, the model can learn more information from unlabeled sample, so it obtains high classification accuracy.

Next, more experiments is used to discuss the effectiveness of the proposed algorithm with different proportions of labeled sample (shown in Fig 2, 3, 4).

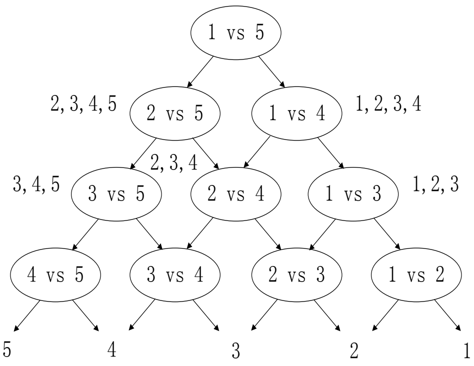

Fig. 1. The classification decision diagram of DAG-SVMs

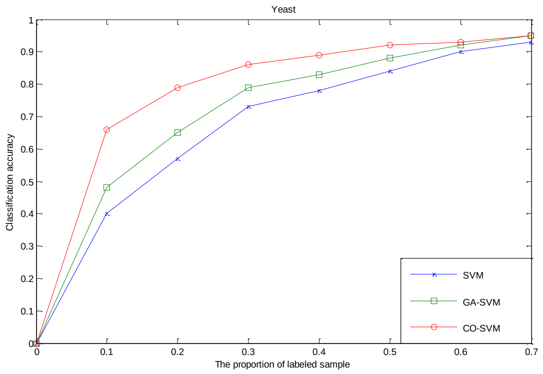

Fig. 2. Three classification results of different classifiers for Yeast dataset with different proportions of labeled dataset

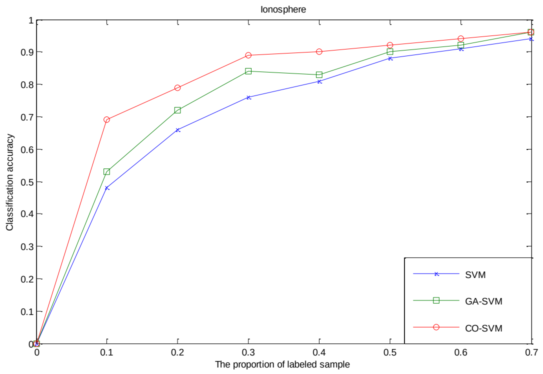

Fig. 3. Three classification results of different classifiers for Ionosphere dataset with different proportions of labeled dataset

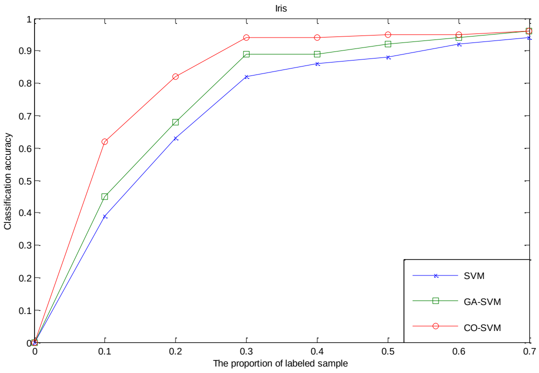

Fig. 4. Three classification results of different classifiers for Iris dataset with different proportions of labeled dataset

4 Conclusions

In this paper, a semi-supervised SVM model for classification has been constructed under the numbers of labeled and unlabeled samples are imbalanced. The effectiveness of semi-supervised strategy based SVM has been validated through UCI dataset. Compared with traditional SVM and GA-SVM, the algorithm proposed has improved the classification accuracy. Although only the basic SVM and GA-SVM model has been combined with the semi-supervised co-training modeling strategy, the idea can be extended to other commonly classification algorithm, such as native Bayesian classifier, neural network, fuzzy classifier, etc. By the integration strategy we can see, the greater the difference between classifiers, the better the final integration effect, it may be a good choice if different structures of the classifiers are used for co-training model development.

Acknowledgments

This work is supported by National Natural Science Foundation (NNSF) of China under Grant 61273142

References

1. B Liang, X F Yuan, Z Q Ge, Co-training partial least squares model for semi-supervised soft sensor development, ***Chemometrics & Intelligent Laboratory Systems*** 147 (2015) 75-85.
2. M Y Zhao, L Jiao, W Ma, H Liu, S Yang, Classification and saliency detection by semi-supervised low-rank representation, ***Pattern Recognition*** 51 (2015) 281-294.
3. X J Zhu, Semi-Supervised Learning Literature Survey, ***Computer Science*** 37 (2005) 63-77.
4. B M Shahshahani, D A Landgrebe, The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon, ***IEEE Transactions on Geosciences & Remote Sensing*** 32 (1994) 1087-1095.

5. Z Xu, I King, Introduction to Semi-Supervised Learning, ***Morgan & Claypool*** 37 (2014) 3036-3036.
6. B Kulis, S Basu, I Dhillon, R Mooney, Semi-supervised graph clustering: a kernel approach, ***Machine Learning*** 74 (2009) 1-22.
7. C K Lee, T L Liu, ***Guided co-training for multi-view spectral clustering***, International Conference on Image Processing IEEE, (2016).
8. D Pierce, C Cardie, ***Limitations of Co-Training for Natural Language Learning from Large Datasets***, Conference on Empirical Methods in Natural Language Processing, (2001).
9. Z H Zhou, K J Chen, H B Dai, Enhancing relevance feedback in image retrieval using unlabeled data, ***Acm Transactions on Information Systems*** 24 (2006) 219-244.
10. Z H Zhou, K J Chen, Y Jiang, Exploiting Unlabeled Data in Content-Based Image Retrieval, ***Lecture Notes in Computer Science*** 3201 (2004) 525-536.
11. V N Vapnik, ***The Nature of Statistical Learning Theory,*** IEEE Transactions on Neural Networks 8 (1995) 1564-1564.
12. R G Brereton,G R Lloyd, Support vector machines for classification and regression, ***Analyst*** 135 (1998) 230-267.
13. Cristianini Nello, S T John, An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods, Cambridge University press, (2000).
14. X G Zhang, Introduction to statistical learning theory and supportive vector machines, ***Acta Automatica Sinica*** 26 (2000) 32-42.
15. A Blum, T Mitchell, ***Combining labeled and unlabeled data with co-training***, Eleventh Conference on Computational Learning Theory ACM, (2000).
16. S A Goldman, Y Zhou, ***Enhancing Supervised Learning with Unlabeled Data***, Proceedings of the Seventeenth International Conference on Machine Learning, Morgan Kaufmann Publishers Inc, (2000).
17. Z H Zhou,M Li, ***Tri-training: exploiting unlabeled data using three classifiers,*** IEEE Transactions on Knowledge & Data Engineering, (2005) .
18. W Wang, Z H Zhou, Analyzing Co-training Style Algorithms, ***Machine Learning: ECML*** 318 (2007) 454- 465.