Comparative Study of Big Data Classification Algorithm Based on SVM

Huasheng Zou 1st Affiliation (*Author*)
College of Information Technology and Engineering
Ningbo Dahongying University
Ningbo City, Zhejiang Province, China
zoufan99@163.com

Zhiyuan Jin 2nd Affiliation (*Author*)
College of Information Technology and Engineering
Ningbo Dahongying University
Ningbo City, Zhejiang Province, China

***Abstract*—**Linear support vector machine (SVM) is one of the most effective machine learning methods to process large-scale data set. Based on the simple expatiation of the principle of support vector machine (SVM), linear support vector machine (SVM) and traditional support vector machine's training time and classification precision are compared and analyzed through the experiment. The results show that linear support vector machine has obvious advantages in terms of training speed and classification precision, which is a powerful tool to realize big data classification and regression. The experimental results show that for linearly separable data set, the LIBLINEAR toolkit is characterized by short training time and high classification precision, which is very suitable for large-scale data set classification.

Keywords—Linear Support Vector Machines; Classifcation;Big data

I. INTRODUCTION

The Support Vector Machine (SVM) developed in the early 1990s is a statistic-based learning method, which avoids the problems of difficult to determine, over-learning and under-learning and local minimum of network structure in the artificial neural networks[1,2]. It shows many unique advantages in solving the classification and regression problems in terms of small sample, non-linear and high-dimension, which is considered as the best theory for the small sample classification, regression and other issues. The theory of support vector machine based on this theory provides a new idea and effective path to solve the nonlinear problem and classification problem.

As the size of the data processed increases, people begin to try to solve the problem of big data by using support vector machine. When using SVM to process large-scale data set, there are problems such as training time is too long and memory space is too large. Thus, how to use SVM to process big data effectively becomes a key problem to be solved urgently. To this end, domestic and foreign scholars conducted a lot of exploration and research, and achieved a series of results. For large-scale data sets, scholars have proposed a method of combining multiple SVMs and parallel algorithms. Ronan Collobert et al. proposed to solve the problem of large-scale data classification by combining multiple SVM classifiers[3]. Zanghirati et al. proposed a parallel algorithm for solving convex quadratic programming problem in the support vector machine[4]. From the perspective of data set division, Graf et al. proposed a parallel algorithm of cascaded support vector machine to decompose data sets[5,6]. Lin et al. used TRON method to solve L2-SVM[7]. For linear support vector machines, Shalew-Shwartz et al. improved the stochastic gradient descent method to solve the dual problem of support vector machines and proposed the Pegason linear support vector machine method[8]. Jochims proposed SVMperf linear support vector machine method[9]. Linear support vector machines are attracting attention for their simplicity, ease-of-use and efficiency when processing large-scale data and have become an effective method of processing large-scale and high-dimensional sparse data[10]. Based on three different datasets, this paper compares the linear support vector machine with traditional support vector machines from three aspects: training time, precision and kernel function.

II. THE CONCEPT AND PRINCIPLE OF SUPPORT VECTOR MACHINE

A. The Concept and Characteristics of Support Vector Machine

Support Vector Machine (SVM) is the first statistical learning theory put forward by Vapnik et al. in 1995[11,12]. It is a major achievement of machine learning research in recent years. This theory not only has a solid theoretical basis, but also suits to solve problem of highly nonlinear classification and regression compared with the conventional intelligent methods such as statistical methods and artificial neural networks. Meanwhile, this theory has been widely used in handwritten digital recognition, face recognition, automatic text classification and image recognition and more[13-15]. Since this method has shown excellent performance over the existing methods, the theory and technology have become new research hotspots after neural network research and will promote the research and application of machine learning theory and technology.

Support vector machine method has the following specific features: (1) support vector machine is a kind of small sample learning method with solid theoretical foundation. It basically does not involve probability measure and law of large numbers and avoids the inference process from induction to deduction, and realizes the transduction reasoning from training samples to forecasting samples, which greatly simplifies the classification and regression. (2) Support vector machine is only decided by a few support vectors. The computational complexity depends on the number of support vectors, and has nothing to do with the dimension of sample space. This not only makes minority support vector become key sample, but

also removes a large number of redundant samples and simplifies the calculation. Meanwhile, computational complexity is not directly related to the dimensionality of the input sample, thus avoiding the computational complexity caused by the dimensionality of the sample space. (3) Support vector machine model has better generalization ability. Support vector machines have a rigorous theoretical foundation that determines the upper bound of the generalization ability of the established model, which is not available in other learning methods so far. (4) Support vector machine method is widely used. It has been applied in many fields, such as text recognition, face recognition, image compression and fault diagnosis and other fields, and has achieved good results.

B. Basic Principle of Support Vector Machines

The basic idea of SVM is to classify the samples in the problem space to be solved into different categories by defining the optimal hyperplane, and to solve the convex programming problem by finding the optimal hyperplane problem, then based on the Mercer kernel theorem, nonlinear mapping, the sample space is mapped to a high dimensional or infinite dimensional feature space (Hilbert space), so that it can solve classification and regression of highly non-linear in the sample space through linear learning machine[16,17], as shown in figure 1.

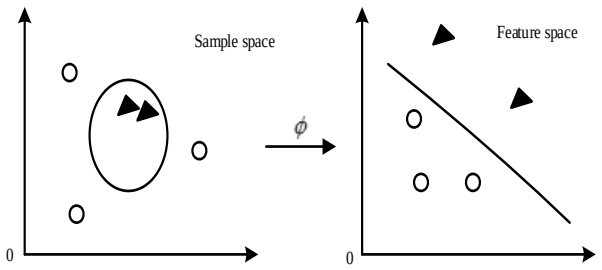

Fig. 1. Mapping from sample space to feature spac

Suppose that the sample set is $D=\{(x_i,y_i),i=1,2,...,l\}$ $D=\{(x_i,y_i),i=1,2,...,l\}$, $x_i\in R^n$ is an n-dimensional vector, $y_i\in(+1,-1),i=1,2,...,l$ is the class label value of the corresponding x_i . When the set of sample sets D is nonlinear separable, the sample space D is mapped to the feature space F by the nonlinear mapping ϕ ., and the optimization problem of its objective function is as follows:

$$\begin{cases} \min\dfrac{1}{2}\|w\|^2+C\sum\limits_{i=1}^{l}\xi_i \\ s.t.\quad y_i((w\bullet\phi(x_i)+b)\geq 1-\xi_i,\quad \xi_i\geq 0,\quad i=1,2,...,l \end{cases} \tag{1}$$

Among them, $\xi_i\geq 0\quad(i=1,2,...,l)$ is a relaxation variable, and C is a adjustable parameter, which indicates the punishment for the wrong sample. The greater the C, the heavier the punishment for the wrong sample.

The optimal classification decision function for the formula (1) is as follows:

$$f(x)=\mathrm{sgn}(\sum_{i=1}^{l}\alpha_i y_i(\phi(x_i)^T\bullet\phi(x))+b) \tag{2}$$

In the above formula, $(\varphi(x_i)\cdot\varphi(x))$ is the dot product of the samples in the feature space, but its essence is the Mercer kernel function, and the commonly used Mercer kernel function is known. Therefore, as long as the appropriate Mercer kernel function is selected according to the specific problem, the optimal decision function can be obtained, thus avoiding the trouble of seeking non-linear mapping ϕ .

III. Support Vector Machine Classification Algorithm for Big Data

A. Classification Algorithm

The classification algorithm uses the LIBSVM and LIBLINEAR software kits developed by Chih-Jen Lin. LIBSVM is a simple, easy-to-use and fast and effective SVM classification and regression software package. User can use kernel functions in the LIBSVM to train nonlinear classifiers and also can use more basic linear SVM. IBSVM has stronger classification ability, which can solve more complex issues. LIBLINEAR is a toolkit designed for linear classification, aiming to improve the efficiency of linear classification. In addition to linear SVM, LIBLINEAR can support linear Logistic Regression model, but cannot achieve nonlinear classification by defining kernel function[18].

LIBSVM has strong classification capabilities than LIBLINEAR and can solve more complex problems. Although both LIBSVM and LIBLINEAR can achieve similar results when performing linear classification, LIBLINEAR is more efficient than LIBSVM in training and prediction, especially in large-scale sample training and has better performance. In addition, LIBSVM and LIBLINEAR provide a wealth of optimization and parameter options, and auto-tuning parameters through grid traversal functions, making the output model become the most optimal model for the current configuration.

B. Algorithm design and Implementation

The experiment uses internationally accepted data sets a9a, real-sim and duke breast-cancer, and number of training samples, test samples and feature dimensions are shown in Table 1.

Table1 Training samples, testing samples and feature dimension

Data Set	Training Samples	Testing Samples	Feature Dimensions
a9a	40,700	20,350	123
real-sim	88,772	88,772	20,958
duke breast-cancer	76	19	7,129

The above three data sets use LIBSVM and LIBLINEAR for classification experiments. LIBSVM has performed training and prediction by using kernel function and Gaussian kernel function respectively. LIBLINEAR has performed simulated training and prediction using L2-regularized L1-Loss SVM, L2-regularized L2-Loss SVM and L1-regularized L2-Loss SVM respectively as shown in table 2.

Table2 Training time and classification accuracy

Algorithm		a9a		real-sim		duke breast-cancer	
		time	precision	time	precision	time	precision
LIBSVM	Linear kernel	187.5s	85.8%	1546s	98.95%	0.712s	77%
	Gauss kernel	112.5s	84.7%	3135s	98.63%	0.726s	77%
LIBLINEAR	L1-L2-Loss	1.089s	85.3%	9.36s	99.16%	0.720s	100%
	L2-L2-Loss	0.596s	85.2%	8.15s	99.62%	0.734s	100%
	L2-L1-Loss	0.927s	86.5%	8.87s	98.91%	0.816s	100%

C. Algorithm Analysis

Analysis of Table 2 shows that training time and classification precision of LIBLINEAR are not different when using different loss functions, while training time and classification precision of LIBSVM when using linear kernel function for big sample data sets are obviously higher than those of Gaussian kernel Function, and its classification precision slightly decreases; for the data set with smaller sample data, the training time and classification precision that use two kernel functions are basically the same[19]. Therefore, for big sample datasets a9a and real-sim, the training time advantage of linear SVM is obvious and the precision of classification is slightly improved. For the small sample dataset duke breast-cancer dataset, the difference between two training time is not big, but the classification precision of linear support vector machine is higher.

IV. CONCLUSIONS

With the development of big data and artificial intelligence technology, the data involved in computing is getting larger and larger, which not only requires computer with large storage space, but also puts forward higher requirements on the arithmetic speed and precision of the algorithm. The emergence of linear support vector machines algorithm package LIBLINEAR has solved the problem. In fact, LIBLINEAR is originally designed to solve the problem of large amount of data. Using completely different optimization algorithms from LIBSVM, LIBLINEAR greatly reduces training computational complexity and time consumption while maintaining similar effects in linear SVM classification. Meanwhile, under the background of big data, there is not much difference between the linear classification and the non-linear classification. Especially in the case of high characteristic dimension and limited sample size, the kernel function method may wrongly classify the category space, resulting in worsening the effect. In addition, LIBLINEAR occupies a large memory, 10GB of data needs close to 50G of memory. Therefore, for large-scale data sets, support vector machines should be further optimized from the time, space and precision[20].

ACKNOWLEDGMENT

The first author thanks Ningbo Dahongying University for providing excellent office and research environment when writing the manuscript. The authors would like to thank the anonymous reviewers for their valuable suggestions and constructive comments on the manuscript.

REFERENCES

[1] V. N. Vapnik. The Nature of Statistical Learning Theory. NY:Spring-Verlag,1995

[2] V. N. Vapnik. Statistical Learning Theory. New York:Wiley,1998

[3] R. Collobert, S. Bengio and Y. Bengio. A parallel mixture of SVMS foe very large scale problems. Neural Computation, 2002,14(5):1105-1.

[4] G. Zanghirati, L. Nanni. A parallel solver for large quadratic programs in traning support Vector machine. Parallel Computing, Vol. 29,2003

[5] Graf H P, Cosatto E, Bottou L, et al. Parallel support vector machines: the cascade SVM. In: Advanced in Neural Information Processing Systems. MIT Press. 2004, 521-528

[6] Alham, N.K.,et al. A distributed SVM for scalable image annotation, Fuzzy Systems and Knowledge Discovery(2011 Eighth International Conference), 2011, 2655-2658

[7] R. Collobert, S. Bengio and Y. Bengio. A parallel mixture of SVMS foe very large scale problems. Neural Computation, 2002,14(5):1105-1114

[8] Shai Shalvev-Shwartz, Yoram Singer, and Snathan Serbro. Pegasos:Primal estimated sub-gradient solver for SVM. In Proceedings of the 24th International Conference on Machine Learning(ICML), 2007

[9] T. Joachims. Training linear SVMs in linear time. In ACM KDD, 2006

[10] C.-C. Chang and C.-J. Lin. A Library for Support Vector Machines. ACM TIST, 2011,2(3):1-27

[11] Ding SF, Hua XP, Yu JZ, An overview on nonparallel hyperplane support vector machine algorithms. Neural Comput Appl, 2014, 25(5):975–982

[12] Davide Anguita, Sandro Ridella, Fabio Rivieccio, ea tl. Hyperparameter design criteria for support vector classifiers[J]. Neurocompting, 2003, 55:109-134