Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 595 – 604

doi:10.3906/elk-1801-157

A multiseed-based SVM classification technique for training sample reduction

Imran SHARIF[1,*], **Debasis CHAUDHURI**[2]
[1]Image Analysis Center, DEAL, DRDO, Dehradun, Uttarakhand, India
[2]DRDO Integration Center Ministry of Defence, Panagarh, West Bengal, India

Received: 17.01.2018 • **Accepted/Published Online:** 22.10.2018 • **Final Version:** 22.01.2019

Abstract: A support vector machine (SVM) is not a popular method for a very large dataset classification because the training and testing time for such data are computationally expensive. Many researchers try to reduce the training time of SVMs by applying sample reduction methods. Many methods reduced the training samples by using a clustering technique. To reduce its high computational complexity, several data reduction methods were proposed in previous studies. However, such methods are not effective to extract informative patterns. This paper demonstrates a new supervised classification method, multiseed-based SVM (MSB-SVM), which is particularly intended to deal with very large datasets for multiclass classification. The main contributions of the paper are (i) an efficient multiseed technique for selection of seed points from circular/elongated class training samples, (ii) adjacent class pair selection from the set of multiseeds by using the minimum spanning tree, and (iii) extraction of support vectors from class pair seed equivalent regions to manage multiclass classification problems without being computationally expensive. Experimental results on a variety of datasets showed better performance compared to other sample-reducing methods in terms of training and testing time. Traditional support vector machine (SVM) solution suffers from $O(n^2)$ time complexity, which makes it impractical for very large datasets. Here, multiseed point technique depends on the estimated density of each data, and the order of computation is $O(n \log n)$. Using the estimated density, the computational cost of the seed selection algorithm is $O(n)$. So, this is the only burden for reducing the sample. However, reducing the sample takes less time with the proposed algorithm compared to the clustering methods. At the same time, the number of support vectors has been abruptly reduced, which takes less time to find the decision surface. Apart from this, the classification accuracy of the proposed technique is significantly better than other existing sample reduction methods especially for large datasets.

Key words: Remote sensing, support vector machine, supervised learning, image processing, sample reduction techniques/methods, multiple classifications

1. Introduction

Image classification is extremely helpful for classifying satellite images. Different classification procedures are described in the literature; for example, nearest neighbor classifiers, artificial neural networks, and support vector machines (SVMs). Among the available techniques, SVMs give better classification accuracy due to their optimization solution and ease of use [1–3]. Recently, various SVM-based classification techniques have been reported in the literature [4–8]. Despite the popularity of SVMs, they are not preferred for large-scale datasets. The main reason is that the training complexities of SVMs vary according to the size of the training dataset. Traditional SVMs take too much running time when trained with a large dataset rather than with a

*Correspondence: imran_ietk@rediffmail.com

set of fine quality samples. Researchers have proposed several solutions for SVMs to be able to handle immense data while avoiding memory capacity and computational cost problems by using sample reduction techniques. However, these solutions are expensive and not generally suitable because they require multiple calls of the SVM or multiple scans of the data [9,10].

There are various sample reduction techniques which reduce the computational burden of SVMs, such as selective sampling, random sampling, and clustering-based SVMs but they are themselves very complex for large datasets [11]. Selective sampling attempts to choose the training data shrewdly in more than one scan of the dataset [12]. Smaller quadratic programming problems can be solved by reduced support vector machines (RSVM) [13] through the selection of important, high-quality training samples as SVs. Genetic programming [14] can manage very large datasets which cannot be accommodated in computer memory. Neural network techniques [15] can likewise be implemented for SVMs to modify the training process. Tresp [16] proposed the Bayesian committee machine technique for SVM training on very large datasets. Cervantesa [17] introduced SVMs for grouping voluminous data utilizing minimum ball clustering. Li [18] proposed a random selection method and a two-phase SVM classification approach for large datasets. This decreases the training dataset, yet it requires twice the number of classifications. Selection of important data from a big data bank, especially when nonstationary, combined of both old and new data samples, is a very critical problem due to computational complexity. In this context, Lin et al. [19] proposed a representative data detection methodology based on pattern recognition techniques. Liu et al. [20] proposed a new methodology using the centroid and its distance from samples to get the geometrical center of the class from labeled datasets without drastically demeaning the accuracy of SVM classification. Gonzalez et al. [21] presented a computational efficient algorithm by using families of locality-sensitive functions (LSH) for selecting the effective data from the big dataset. Feng et al. [22] suggested a novel ensemble classifier to handle imbalanced datasets and compared it with other existing ensemble margin-based methods. Wang et al. [23] suggested a training data reduction method in two major steps; the first step was training for data cleaning and the second step was extracting the important training data using a novel entropy-based algorithm.

The current paper presents a new supervised classification method, i.e. multiseed-based SVM (MSB-SVM), that scans the whole dataset only once and returns the samples that have a high probability of support vectors for the SVM. The MSB-SVM classification technique reduces the computational cost in two ways: firstly, an efficient training sample selection method is implemented based on the multiseed technique without clustering the data, and secondly, multiclass problems are handled using the minimum spanning tree (MST) to improve the cost efficiency of the proposed classification technique. The multiseed technique is explained in Section 2. In Section 3, the proposed MSB-SVM method is described. Experimental results and analysis are presented in Section 4. Finally, a summary of the paper is given in Section 5.

2. The multiseed technique

Clustering techniques, for example K-means and Forgy, as well as their enhanced variant ISODATA, are based on single seed points. For noncircular classes, these methods do not work properly since in these cases more than one seed is required. Here, we needed a multiseed selection algorithm for handling noncircular classes which would later be used for support vector extraction. Chaudhuri and Chaudhuri [24] proposed a parameter-free seed point detection approach and we used this method in our study.

3. The proposed MSB-SVM technique

In this section, we show another technique for authentic SVM classification. According to this, the sample set should be small and represent the original training sample set sufficiently in order to reduce the cost of the training time. There are four steps in our approach. We begin with the training samples of all classes and the seed points of each class are detected by the seed point detection (SPD) algorithm which was proposed by Chaudhuri and Chaudhuri [24]. For simplification, we expect that there are just two classes. Figure 1a shows, training sample of two classes, and due to the elongated nature of both classes, more than one seed is required. First, the seed points are found for both classes by using the SPD algorithm. Suppose that there are two and three seed points (★) which are detected for the classes $S1$ and $S2$, respectively. These are denoted as $\{C_{11}, C_{12}\}$and $\{C_{21}, C_{22}, C_{23}\}$for $S1$ and $S2$, respectively. The next step is to find the MST between all these seed points. The seed points of these classes and the MST between these seed points are shown in Figure 1b. We then find the equivalent point set between the two seed points $\{C_{12}, C_{21}\}$from these two different classes, which form an edge in the MST. Figure 1c shows the equivalent points set between the two classes and those points are marked by a red plus (✦) and a green plus (✦). The equivalent point set is the condensed training data of fine quality for support vector extraction. Finally, we find only four support vectors (■), which are necessary to define the optimal decision hyper-plane as shown in Figure 1d. We now describe our approach step by step.

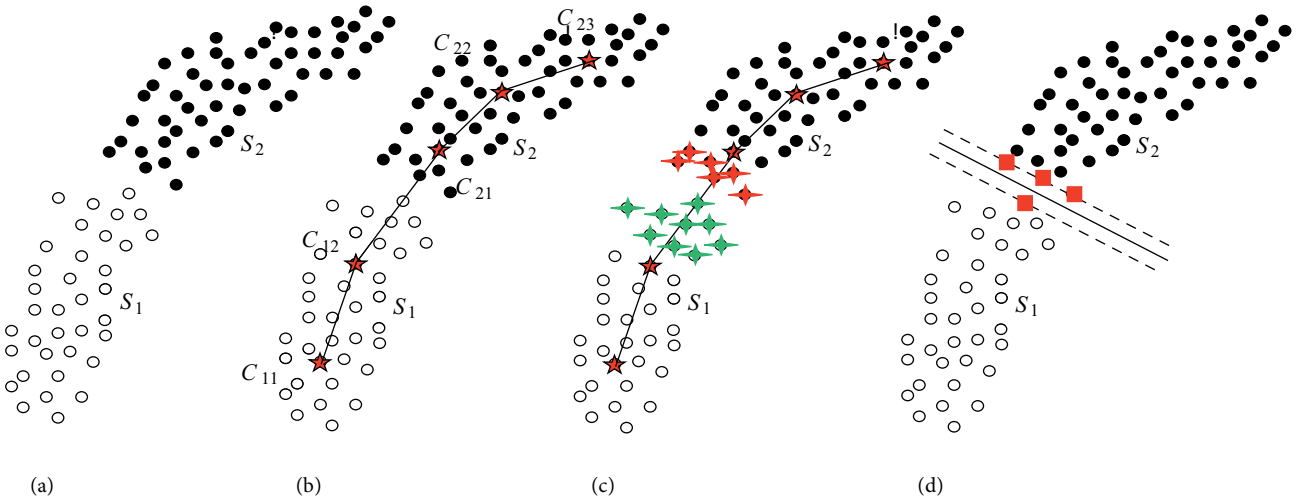

Figure 1. Two-class problem. (a) Two training classes, (b) Multiseed points and MST between seed points, (c) Equivalent points of the two seed points of both the classes which are connected by MST edge, and (d) Support vector and the discriminant function.

3.1. Seed point detection (SPD) algorithm

Seed points are represented as the highest density point of a cluster. If the pattern is circular (homogeneous), then a single seed is sufficient to cluster all the data. If the pattern is elongated (nonhomogeneous), then multiseeds of that cluster are required. Earlier, we proposed a seed point detection based on border points of the pattern [24]. Here, however, training samples of a particular class are collected from different images and do not form an object. In such cases, seed point detection using a border point is not possible. Hence, here, we

detected seed points from training classes and we did not consider border points.

3.2. MST-based seed point connectivity

For n_0 numbers of seed points, ${}^{n_0}C_2$ seed pairs are chosen for finding transfer $n_0(n_0-1)/2$ decision planes, which represents a huge computation. For two nearby classes, we form an MST using n_0 seed points. Two nearby classes may be considered for the SVM training decision plane if these classes are connected by an edge in the MST.

3.3. Equivalent point detection

In this section, we discuss the choice of possible pair seeds from the MST-connected seed in order to find the equivalent point set. The pair (C_{12}, C_{21}) in Figure 1b is considered for the equivalent point set for three reasons: (1) $C_{12} \in S_1$ and $C_{21} \in S_2$ and hence, they belong to different classes, (2) the seed points C_{12} and C_{21} form an edge in the MST, and (3) they are the border region seed points between the two classes. Now, in Figure 1b, the equivalent point set is denoted as EQ_{PS} and defined by the union of two border point sets as follows: The border point set of S_1 is denoted as BP_{S_1} and defined by:

$$BP_{S_1} = \{(x_i y_i) : d\{(x_i, y_i), C_{21}\} \leq d(C_{12}, C_{21}) \forall (x_i, y_i) \in S_1\}.$$

Similarly, the border point set of S_2 is denoted as BP_{S_2} and defined by:

$$BP_{S_2} = \{(x_i y_i) : d\{(x_i, y_i), C_{12}\} \leq d(C_{12}, C_{21}) \forall (x_i, y_i) \in S_2\},$$

where $d(A, B)$ represents the Euclidian distance between the two points, A and B. Therefore, the equivalent point set is defined as $EQ_{PS} = BP_{S_1} \cup BP_{S_2}$.

3.4. Support vector extraction

The equivalent point set is detected from the MST connected interclass seed points. It is true that $EQ_{PS} = (S_1 \cup S_2)$ in Figure 1b, and $\#EQ_{PS} \ll \#(S_1 \cup S_2)$ where "$\#N$" refers to the number of points in the set N. The set EQ_{PS} has a much smaller number of training points than the whole dataset $(S_1 \cup S_2)$. EQ_{PS} is represented as the condensed training points for finding probable support vectors and decision planes using the SVM training method.

3.5. SVM for multiclass classification

Originally, SVMs were developed to perform binary classification while in remote sensing applications, multiclass classification handles more than one class. There are a number of methods for multiclass classification proposed by researchers. Two of the most widely adopted multiclass classification strategies [25–27] are one-against-all and one-against-one. In the one-against-all procedure, N binary SVM classifiers are required for N class classifications, where each classifier trains one v/s rest class while the one-against-one strategy requires $N(N$ - $1)/2$ binary SVM classifiers. A drawback of the one-against-all approach is that the ratio of the training sample of one class to the rest of the classes is unbalanced, and in the one-against-one approach, the number of classifiers and the number of classes increase proportionally. The idea behind the SVM-based classifier is to find the optimal hyper-plane between two classes which are very near to each other. Previously, in multiclass problems there has been no mechanism for finding the most possible pairs of closer classes. If there are N

classes with total $n_0 (n_0 >> N)$ seed points (by the SPD algorithm), then also there are (N-1) edges formed in the MST, which are interclass edges. Hence, in the proposed multiclass architecture only (N-1) hyper-planes $H : y = \langle \mathbf{W}, \mathbf{X} \rangle + \ b = 0$ using the SVM paradigm are computed. The classification criteria for labelling an unknown pattern $\boldsymbol{X}$ will be as follows:

i. Compute (N-1) distances from the pattern $\boldsymbol{X}$ to the (N-1) hyper-planes $f_i(\mathbf{X}) = \langle \mathbf{W}.\mathbf{X} \rangle + \ b = 0, \ i = 1, 2, ..., N-1$ i.e. $D_i(\mathbf{X}) = \frac{\langle \mathbf{W}.\mathbf{X} \rangle + b}{\|\mathbf{W}\|}, i \ = \ 1, 2, ..., N-1$.

ii. If $D_i(\mathbf{X})$ is unique positive then find the hyper-plane and corresponding class for which it is positive. Then assign X to that class.

iii. If $D_i(\mathbf{X})$ is not unique, i.e. there is more than one hyper-plane which is positive, then find the minimum distance hyper-plane from the pattern X and assign X to the corresponding class.

Figure 2 shows the flowchart of the proposed algorithm. We reduce the computational cost of the MSB-SVM classification technique through two means. First, an efficient training sample (equivalent point set) selection method based on the multiseed technique is used. The equivalent set is the condensed training sample and the balanced set between the two MST connected interclasses. Second, handling multiclass problems by using MST is another contribution to the cost efficiency of data classification. A maximum of (N-1) hyper-planes are required among N classes and the ratio of the training samples of one class to another class is also balanced.

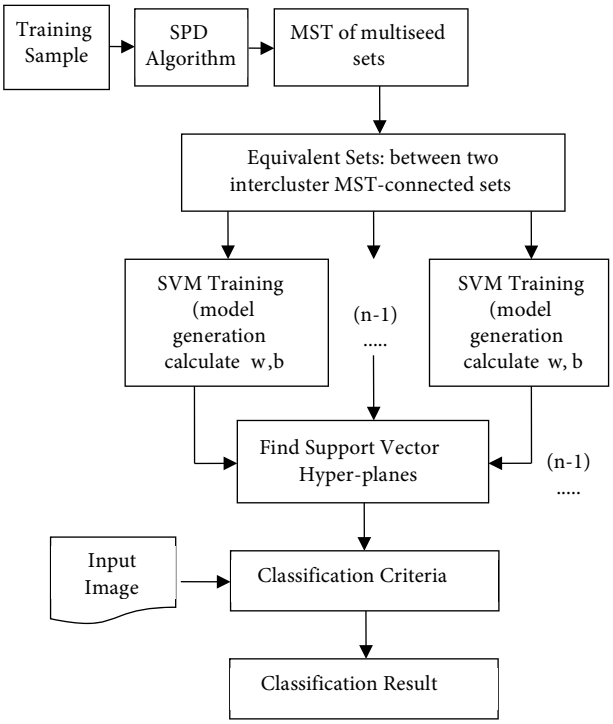

Figure 2. The flowchart of the proposed classifier.

4. Experimental results and discussion

In this paper, we assessed the execution of our approach with a scene obtained from IRS-1D, IKONOS imageries. We investigated the outcomes of the proposed algorithm compared with the existing algorithms in the literature using real-life benchmark data. Numerical experiments were conducted in the C++ language with 2-GB RAM memory in the Intel Xeon 2.0 GHz hardware environment using LIBSVM. Our purpose was to examine the potential of the proposed method in handling large datasets in terms of training and testing time.

Table 1 shows the execution of the different techniques using an IRS-1D image of size 512 × 512 pixels. The scene was classified into six different vegetation, concrete, and natural classes using the multiseed, SVM, and MSB-SVM techniques. The classes obtained were concrete, water, sand, forest, soil, and rock. We took a total of 3003 training samples of the above classes using a guidance map. We applied a linear kernel multiclass (one-against-one architecture) SVM classifier on these training samples and the parameters were generated using ten-fold cross-validation. We obtained a total of 219 support vectors and the time taken by the SVM classifier was 80.3 s. A total of only 360 points (equivalent point set) were obtained between the MST connected interclass edges of the training samples by the proposed algorithm. A total of 32 support vectors were obtained by the proposed algorithm; this number is close to one seventh of the number of total support vectors obtained using the SVM classifier. In addition, the time taken by the proposed technique was 22.57 s, which is close to one quarter of the time taken using the SVM classifier. Although the time taken by the multiseed supervised classifier was less (15.06 s), the accuracy was poorer. The accuracy of the performance using the proposed algorithm is better than those of the other two methods because it has a high probability of retaining the support vector in the reduced training dataset.

Table 1. Comparison of different classifiers on IRS-1D data.

Algorithm	Training sample	Training time (s)	Support vector	Overall accuracy (%)
SVM	3003	80.30	219	88.6070
Multiseed	3003	15.06	–	87.0438
Proposed	360 (Equivalent point set)	22.57	32	88.5743

Table 2 shows combined confusion matrices for multiseed, SVM, and MSB-SVM techniques using an IKONOS multispectral image with 4m ground resolution. The values are represented in the order of multiseed/SVM/MSB-SVM. The scene is classified into seven different land-cover classes (water, building, road, concrete, bare land, dark vegetation, and vegetation) using the multiseed, SVM, and proposed supervised classification techniques. We have seen that many concrete structures are classified as road pixels by multiseed and SVM classifiers. However, fewer road structures are classified as concrete when using the proposed classifier, which is reflected in the combined confusion matrices table. We have considered a total of 886 training samples of seven classes, and among them, the training samples of water, building, road, concrete, bare land, dark vegetation, and vegetation classes are 120, 100, 68, 274, 42, 69, and 213, respectively. We have classified the training samples by using multiseed, SVM, and MSB-SVM classifier techniques. It is clear from the confusions matrices table (Table 2) that the proposed technique yields significantly improved results. For example, all 120 water pixels of the training samples are classified as water class by using multiseed, SVM, and MSB-SVM

classifier techniques and the classification result for methods multiseed, SVM, and MSB-SVM classifier is represented as 120/120/120, i.e. all the above methods are classified 120 water pixels as water pixels. Similarly, the classification results of other classes are represented in similar fashion in the confusions matrices table (Table 2).

Table 2. Comparison of different classifiers on IKONOS data.

Class name	Water	Building	Road	Concrete	Bare land	Dark vegetation	Vegetation
Water	120/120/120	0/0/0	0/0/0	0/0/0	0/0/0	0/0/0	0/0/0
Building	0/0/0	88/94/95	2/0/0	10/6/5	0/0/0	0/0/0	0/0/0
Road	0/0/0	1/1/0	62/63/67	5/4/1	0/0/0	0/0/0	0/0/0
Concrete	0/0/0	8/4/4	35/21/9	225/245/258	6/4/3	0/0/0	0/0/0
Bare land	0/0/0	4/3/2	0/0/0	4/1/3	34/38/37	0/0/0	0/0/0
Dark vegetation	0/0/0	0/0/0	0/0/0	0/0/0	0/0/0	65/62/61	4/7/8
Vegetation	0/0/0	0/0/0	0/0/0	0/0/0	0/0/0	7/12/4	206/201/209

We compared our proposed algorithm with other sample reduction techniques such as Kernel Bisecting K-means and sample removal (KBK-SR) [28], sample reduction by data structure analysis (SR-DSA) [29], and no sampling using the GERMAN and LIVER-DISORDERS benchmark datasets. We divided the datasets into training and testing datasets and compared the time complexity and testing accuracies with the above-mentioned techniques as shown in Table 3. We found that our algorithm performed well in terms of the sampling time and test accuracy.

Table 3. Performance comparison of sampling methods.

Dataset	Sampling method	No. of samples	Sampling time (s)	Training time (s)	Test accuracy (%)
GERMAN	MSB-SVM	700	4.8	26.01	78.93
	KBK-SR	700	7.1	62.31	72.81
	SR-DSA	700	83.2	38.61	72.12
	No sampling	700	0	180.71	75.01
LIVER-DISORDERS	MSB-SVM	150	1.25	7.71	76.21
	KBK-SR	150	1.73	9.12	74.04
	SR-DSA	150	2.37	8.72	72.19
	No sampling	150	0	13.23	75.83

In order to test the effectiveness of the proposed method, a series of experiments for large datasets were implemented. We tested methods on the four benchmark datasets obtained from UCI, Statlog, and other collections. Table 4 shows the performance comparison results of different methods for large datasets. The sample reduction percentage is much greater in the "USPS" and "Letter" datasets compared to the other two. Due to the low number of samples and the data attributes, it takes less time for the support vector calculation. In contrast, the forest cover type dataset had the more number of samples; hence, the time taken for the SVM training was significant for the whole training set compared to the reduced training set. We have noticed that the computational time was reduced and the accuracy of the classification was much closer to the traditional

Table 4. Performance comparison of methods on large datasets.

Dataset	Training DATA	Test DATA	Feature	Class	Training time (s)		Number of support vectors		Testing time (s)		Test accuracy %	
					LIB SVM	MSB SVM	LIB SVM	MSB SVM	LIB SVM	MSB SVM	LIB SVM	MSB SVM
IJCNN1	49,990	21,701	22	2	465	90	9710	338	204	45	98.2	98.8
COV- TYPE	100,000	40,000	54	2	6850	440	26720	1243	370	84	92.1	94.2
USPS	7291	2007	256	10	110	35	2760	52	120	11	95.7	95.9
LETTER	15,000	5000	16	26	145	65	7120	215	45	19	97.1	97.2

SVM for small and large datasets (Table 4). Therefore, the proposed technique is more effective for large datasets.

5. Conclusion

We have presented a new approach for handling large datasets and authentic SVM classification. When the size of the dataset is large, the classification performance is poor while using the SVM classifier. Here, we have presented a new supervised classification method, MSB-SVM, which scans the entire dataset only once and provides a high-quality training sample which has a greater probability of becoming a support vector for SVM classification. The MSB-SVM classification technique reduces the computational cost in two ways. First, an efficient training sample selection method based on the multiseed technique without clustering the dataset, and second, handling of the multiclass problem by using the minimum spanning tree for decreasing the computational complexity in terms of the testing time of the proposed classification technique. We have also compared the results with other existing methods for small and large datasets and we have seen that the accuracy of performance by the proposed algorithm is better than those of the other methods. The remarkable feature is that the method can be used in both circular and elongated training datasets. Our proposed algorithm has a faster sampling time compared to the other methods while maintaining classification accuracy.

References

[1] Vapnik VN. The Nature of Statistical Learning Theory. New York, NY, USA: Springer, 1995.

[2] Foody GM, Mathur A. A relative evaluation of multiclass image classification by support vector machines. IEEE T Geosci Remote 2004; 42: 1335-1343.

[3] Foody GM, Mathur A. toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification. Remote Sens Environ 2004; 93: 107-117.

[4] Du S, Chen S. Weighted support vector machine for classification. IEEE Sys Man Cybern 2005; 4: 3866-3871.

[5] Tsai C. Training support vector machines based on stacked generalization for image classification. Neurocomputing 2005; 64: 497-503.

[6] Strack R, Kecman V, Strack B, Li Q. Sphere support vector machines for large classification tasks. Neurocomputing 2013; 101: 59-67.

[7] Gautam RS, Singh D, Mittal A, Sajin P. Application of SVM on satellite images to detect hotspots in Jharia coal field region of India. Adv Space Res 2014; 41: 1784-1792.

[8] Hwang YS, Kwon JB, Moon JC, Cho SJ. Classifying malicious web pages by using an adaptive support vector machine. Journal of Information Processing Systems 2013; 9: 395-404.

[9] Ghoggali N, Melgani F, Bazi Y. A multiobjective genetic SVM approach for classification problems with limited training samples. IEEE T Geosci Remote Sensing 2009; 47: 1707-1718.

[10] Li X, Cervantes J, Yu W. Two-stage SVM classification for large data sets via randomly reducing and recovering training data. IEEE Sys Man Cybern; 7–10 Oct 2007; Montreal Que. Canada: pp. 3633-3638.

[11] Yu H, Yang J, Han J. Classifying large datasets using SVMs with hierarchical clusters. Lect Notes Artif Int; 24–27 Aug 2003; Washington DC, USA: pp. 306-315.

[12] Tong S, Koller D. Support vector machine active learning with applications to text classification. The Journal of Machine Learning Research 2002; 2: 45-66.

[13] Lee YJ, Huang SY. RSVM: Reduced support vector machines. IEEE T Neural Networ 2007; 18: 1-13.

[14] Folinno G, Pizzuti C, Spezzano G. GP Ensembles for large-scale data classification. IEEE T Evolut Comput 2006; 10: 604-616.

[15] Lin CT, Yeh CM, Liang SF, Chung JF, Kumar N. Support-vector-based fuzzy neural network for pattern classification. IEEE T Fuzzy Syst 2006; 14: 31-41.

[16] Tresp V. A bayesian committee machine. Neural Computation 2000; 12: 2719-2741.

[17] Cervantesa J, Li X, Yu W, Li K. Support vector machine classification for large data sets via minimum enclosing ball clustering. Neurocomputing 2008; 71: 611-619.

[18] Li X, Cervantes J, Yu W. Two-stage SVM classification for large data sets via randomly reducing and recovering training data. IEEE Sys Man Cybern; 7–10 Oct 2007; Montreal, Que., Canada: pp. 3633-3638.

[19] Lin WC, Tsai CF, Ke SW, Hung CW, Eberle W. Learning to detect representative data for large scale instance selection. J Syst Software 2015; 106: 1-8.

[20] Liu C, Wang W, Wang M, Lv F, Konan M. An efficient instance selection algorithm to reconstruct training set for support vector machine. Knowl-Based Syst 2017; 116: 58-73.

[21] Gonzalez AA, Pastor JFD Rodriguez JJ, Osorio CG. Instance selection of linear complexity for big data. Knowl-Based Syst 2016; 107: 83-95.

[22] Feng W, Huang W, Ren J. Class imbalance ensemble learning based on the margin theory. Applied Sciences 2018; 8: 815-843.

[23] Wang S, Li Z, Liu C, Zhang X, Zhang H. Training data reduction to speed up SVM training. Appl Intell 2014; 41: 405-420.

[24] Chaudhuri D, Chaudhuri BB. A novel multiseed nonhierarchical data clustering technique. IEEE T Syst Man Cy B 1997; 27: 871-877.

[25] Melgani F, Bruzzone L. Classification of hyper-spectral remote sensing images with support vector machines. IEEE T Geosci Remote 2004; 42: 1778-1790.

[26] Hsu CW, Lin CJ. A comparison of methods for multi-class support vector machines. IEEE T Neural Networ 2002; 13: 415-425.

[27] Cheng L, Zhang J, Yang J, Ma J. An improved hierarchical multi-class support vector machine with binary tree architecture. International Conference on Internet Computing in Science and Engineering; 28–29 Jan 2008; Harbin, China: pp. 412-414.

[28] Liu XZ, Feng GC. Kernel bisecting k-means clustering for SVM training sample reduction. Int C Patt Recog; 08-11 Dec 2008; Tampa, FL, USA: pp. 4562-4568.

[29] Wang D, Shi L. Selecting valuable training samples for SVMs via data structure analysis. Neurocomputing 2008; 71: 2772-2781.