A Novel Clustered Support Vector Machine with Reduced Support Vectors for Big Data Classification

Gokkul Nath T.S and Ramanathan. R*
Department of Electronics and Communication Engineering,
Amrita School of Engineering, Coimbatore,
Amrita Vishwa Vidhyapeetham, Amrita University, India.
Email: r_ramanathan@cb.amrita.edu*.

***Abstract*— Big data classification demands support vector models with huge number of support vectors, which are prone to overfitting and complex in nature. In this paper, we propose a method to solve the overfitting problem and improve the generalization of the model by reducing the number of support vectors. Using the proposed approach, the number of support vector has been reduced on average of 90% of the original count when trained using conventional SVM approach. The Discussed method proves to improve the performance of the conventional method by reducing the number of support vectors at cost of accuracy. This method can be used in applications involving long term prediction like weather/climate prediction and time critical applications which require rapid performance with a compensated accuracy.**

Keywords—Support vector machine; Sequential Minimization Optimization (SMO); Big Data; Clustering;

I. INTRODUCTION

Support vector machines (SVM) are powerful binary classifiers based on optimal hyper planes with hard or soft margins. SVM based solutions have successfully been applied to various problems ranging from text categorization[1,2], face recognition, detection, and verification [3,4], speech recognition, to bankruptcy prediction[5], bioinformatics[6], remote sensed data analysis[7], information and image retrieval, time series forecasting, information security and etc. SVMs classifiers are fast, have capability to use kernels, solutions obtained are sparse in nature and have no local maxima. Further, optimizing margin can be controlled easily. SVM overcomes various traditional issues like overfitting , curse of dimensionality and etc. SVMs have well established theoretical foundation and implementation. They are gaining popularity and have rapid development because they possess various captivating features which include: Excellent mathematical illustrations, good generalization capabilities and reassuring good performance. Reduction in number of support vectors is crucial because large number of support vectors makes the model more complex and the model becomes prone to over-fitting. Computational complexity of the model is huge since scalar product is computed the in the input space for a binary classification . Fewer support vectors leads to smaller testing error and SVM model possesses better generalization capability. The state of art SVM techniques range from least squares SVM (LSSVM), proximal SVM (PSVM) , twin SVM (TWSVM), multi-kernel SVM, AUC maximizing SVM, localized SVM, cost sensitive SVM, fuzzy SVM, K-support vector classification regression (K-SVCR) and etc., have been developed[8-11].This paper used LibSVM [12] library for implementing the proposed method to solve the SVM formulation using Sequential Minimization optimization(SMO) method. The paper is organised as follows: Section II reviews the concept of Support Vector Machines and various solving strategies along with discussion of available open source libraries. Section III describes about the proposed method for reduction in number of support vectors using clustering techniques. The proposed approach is validated by applying the proposed method on historical climate data of various cities and the results are discussed in Section IV. Finally Section V will provide concluding remarks and future scope of improvements.

II. CONVENTIONAL SVM METHOD

Vladimir Vapnik along with his co-workers invented Support Vector Machines in 1979. SVM separates positive instances from negative instances of the data with a maximum margin using a hyperplane (see Figure 1). The distance between the nearest of the positive and negative instance from the hyperplane defines the margin. The data instances that lie on the bounding margin that separates the data set are called support vectors.

A. Linear SVM Model: The Hyperplane separating the positive and negative examples is a line given by the equation:

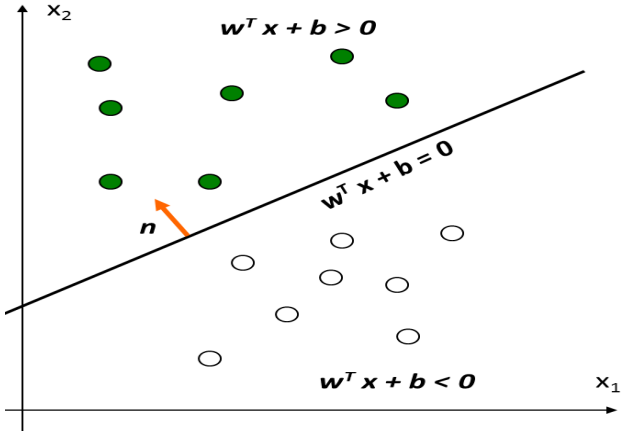

Figure 1: Linear SVM Model [16]

$$v = \vec{w}.\vec{x} + b$$

$\vec{w}$ - Vector normal to the Hyperplane.

$\vec{x}$ – Input vector. b- Bias term.
The Hyperplane separating the positive and negative instances are given by the plane $\vec{w}.\vec{x} + b = 0$. The support vectors lie on the planes $\vec{w}.\vec{x} + b = \pm 1$
The margin $M = \frac{1}{\|w\|_2}$
Maximization of margin can be expressed by the following optimization problem:

$$\underbrace{min}_{\vec{w},b} \frac{1}{2}\|w\|^2 \quad \text{Such that } y_i(\vec{w}.\vec{x} - b) \geq 1.$$

B. Nonlinear SVM: In this case, Hyperplane separating the positive and negative examples set is obtained by applying the kernel trick ie. Mapping the given data to higher dimensions such that positive examples and negative examples set are linearly separated by a hyperplane.Figure 2 shows that nonlinear data mapped to higher dimension using kenel function $\varphi(x)$ and construction of hyperplane in higher dimensional space.

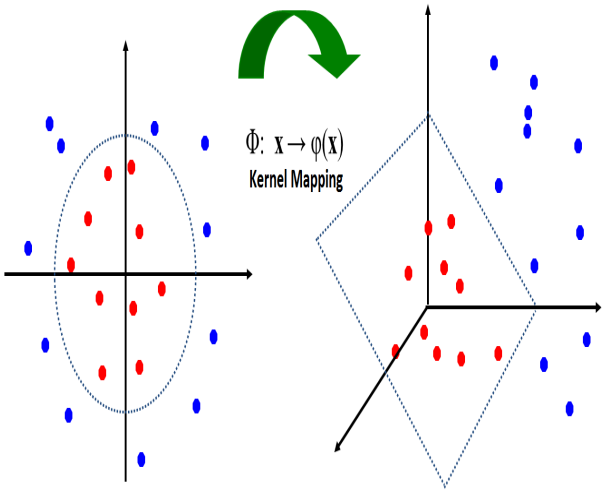

Figure 2: Example for Nonlinear SVM Model [13]

C. Types Of SVM Models:

1) Hard Margin SVM:

If the data set is linearly separable, two hyperplanes are selected such that distance between them is large and no misclassification is allowed. When data becomes linearly inseparable the width of margin becomes very small consequently the performance of the classifier decreases. Thus, hard margins usually preferred only when the data is linearly separable.

2) Soft Margin SVM (L1-SVM :Hinge Loss)*:*

When the data set is nonlinear and inseparable, a loss function is introduced such that a tradeoff between width of the margin and` misclassification error is established. Misclassification errors can be separation of data on wrong side of Hyperplane. Soft margins can be applied to all problems.

D. Stratergy to Solve SVM Problem:

SVMs are constructed by reduction of machine learning problems into optimization problem and these optimization tasks are solved using various techniques such as linear programming, quadratic programming, semi-definite programming and etc. SVM optimization problems can be solved either in primal form or dual form. Generally Dual formulation is solved since it exploits the kernel trick is applied. Dual formulation can be solved by coordinate descent, Gradient projection and decomposition.
A huge quadratic programming (QP) optimization problem must be solved in order to train a SVM. Hence, to reduce the computation complexity this huge QP problem is broken down successively into smaller QP problem. These smaller QP problems can be solved analytically henceforth reducing the computation complexity and training time.

E. OpenSource SVM Libraries:

SVMlight, Scikit-learn, LibSVM are the most widely used open source libraries that solve SVM related problems. LibSVM was developed by Chang and Lin and contains C-classification, ν-classification, ε-regression, and ν-regression. It implements SMO solving strategy to solve the quadratic problem. It has been developed in C++, Python and Java. It also supports MATLAB interface. Further, multi-class classification and weighted SVM for unbalanced data can be implemented. It has inbuilt cross-validation and automatic model selection features.

III. Proposed Clustered SVM

Reduction in number of Support Vector is essential because it reduces the computation complexity of the model, which in turn gives the user the ability to implement real time applications on low power computing devices and reduces hardware requirement. Further, it also reduces the latency time as reduction in number of support vectors makes the classification process faster and easier for upgrading /reconfiguration of SVM model if required. When Conventional SVM Method is used to models typically have nominal number of support vectors which contain components that are not important. Thus the model becomes complex and inefficient. Hence, the proposed method can be used to supress the number of support vector by extracting only the essential support vectors which leads to less complex model. Hyperplane parameter (C) and bias term (Gamma) are to be computed which define the model. Figure 3 depicts the steps involved in reduction of support vectors by applying the proposed method.

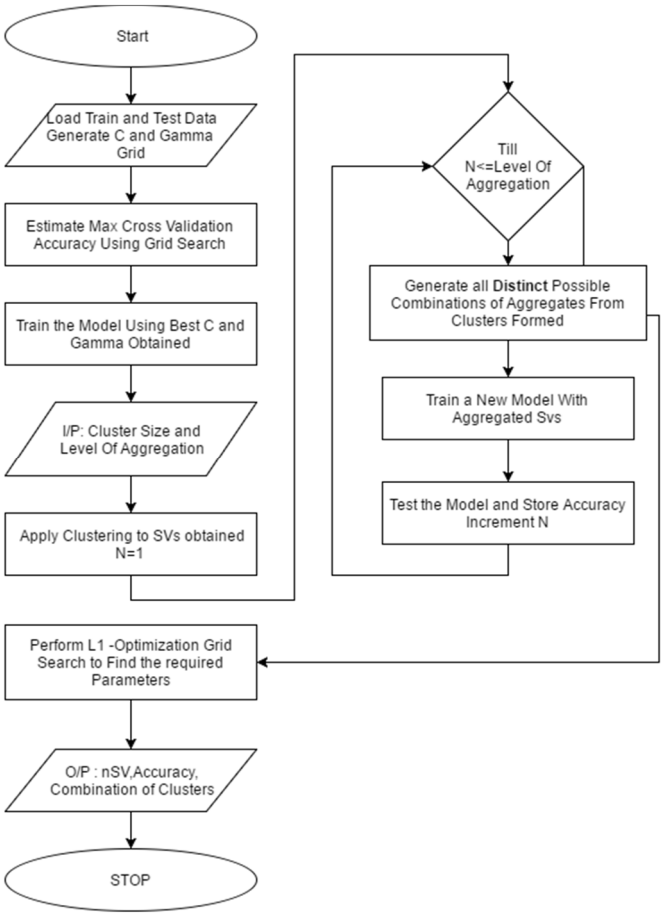

Figure 3: Flowchart of Proposed Method

Process is commenced by loading the training instances along with its corresponding class labels in the required syntax. Grid search is performed to obtain optimum C and Gamma parameters. The C and gamma parameter are found from the respective cross validation Accuracies obtained and the values are taken. The C and gamma parameters are again fine-tuned by making a fine grid in a suitable range for better performance. They are taken as Best C and Best gamma. Using this Best C and Best gamma Parameter, an optimum SVM model is trained. Now, clustering algorithm is applied to the support vectors obtained for this trained model. These Clusters are then used for aggregation. Aggregation of these support vectors into all possible distinct groups is done based on the level of aggregation and cluster size given as input by the user. These aggregated clusters are to be swapped with existing support vectors of the model and then tested to obtain the accuracy. This process of aggregation is done till all possible distinct groups are tested. The respective combination of clusters obtained, level of aggregation, cluster size and corresponding accuracy are stored for further processing. The L1–optimization grid search is performed to find the best parameter as per Requirement.

IV. Experimentation and Validation

To verify the effectiveness and efficiency of the proposed method, historical climate data of Trichy, Delhi, Coimbatore, Calcutta, Bombay and Chennai were taken. These data were used to train SVM (Conventional SVM-SMO Method) models using LibSVM library. Then, the clustered (K-means clustering) approach was applied and the following results were observed.

Nomenclature

TRY-Trichy; DLI- Delhi; CBE-Coimbatore;

CAL-Calcutta; BOM-Bombay; CHN-Chennai;

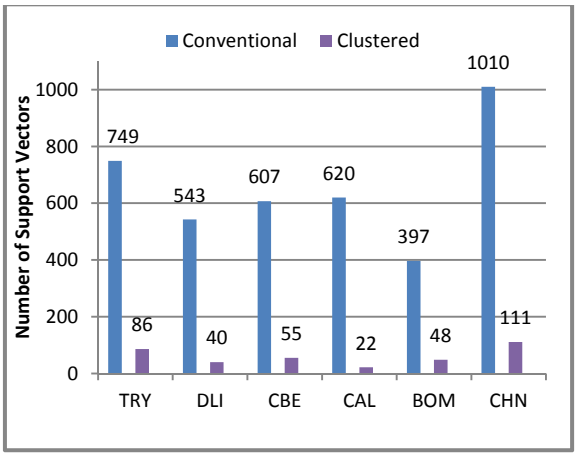

Figure 4: Comparison of Number of Support vectors

Figure 4 illustrates the correlation between the number of support vectors for the model obtained when conventional SVM approach and clustered approach are used. It is inferred that Chennai has highest number of support vectors when trained using conventional method (1010) and these get reduced to 111(89% Reduction) when clustered approach is followed. Delhi has the highest (92.6%) reduction of number support vectors from 543 to 40 support vectors. On an average the proposed method has 90% reduction in number of support vectors.

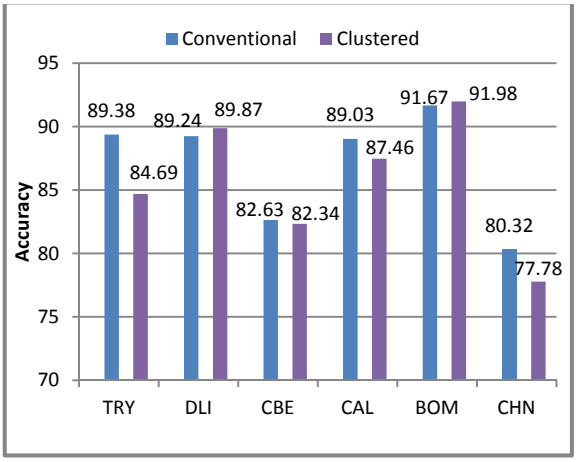

Figure 5: Comparison of Accuracy

Figure 5 illustrates the comparison between accuracies for the model obtained when conventional SVM approach and clustered approach are followed. The performance of the

proposed approach is almost similar to that of conventional method. Models of Delhi and Bombay obtained using clustered approach (89.87% & 91.97%) performed better than the conventional SVM method (89.2% & 91.6%) while Trichy and Chennai had reduction in accuracy of 4.6% & 2.53% respectively along with the reduction in support vectors. Hence, we can infer that the proposed approach has performance similar to that of conventional SVM method with a maximum deviation of 4% in accuracy.

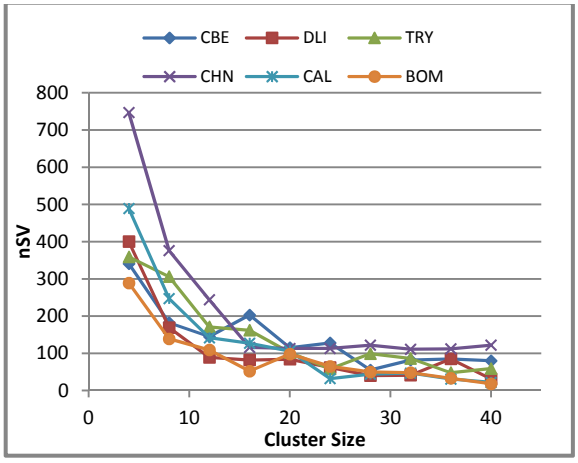

Figure 6: Number of Support Vectors vs. Cluster Size

Using the proposed clustering approach, trivial reduction in number of support vectors can be observed. Figure 6 illustrates the reduction of number of support vector as the cluster size increases. Due to increase in number of clusters the redundant and non-essential support vectors which have negligible effect on the accuracy gets removed and only essential support vectors are extracted from the total support vector set. Hence, as the cluster size increases the number of essential support vectors gets supressed exponentially.

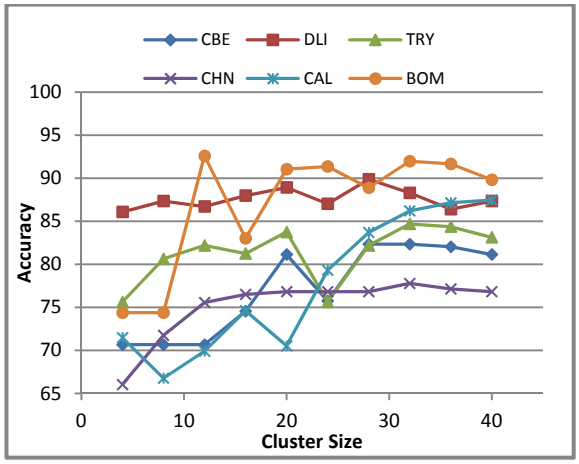

Figure 7: Accuracy vs. Cluster Size

Figure 7 explains the relationship between cluster size and accuracy. We infer that the accuracy increases as the cluster size increases, reaches a maximum value where extracted support vectors are only essential and then decreases. When cluster size is increased after saturation in extraction of essential support vectors there is no trivial improvement in accuracy of the model. In certain cases like Bombay, the accuracy might decrease because of extraction after saturation will miss out certain essential support vectors during the extraction process. Hence, the cluster size must be chosen in an appropriate way such that the requirements are met.

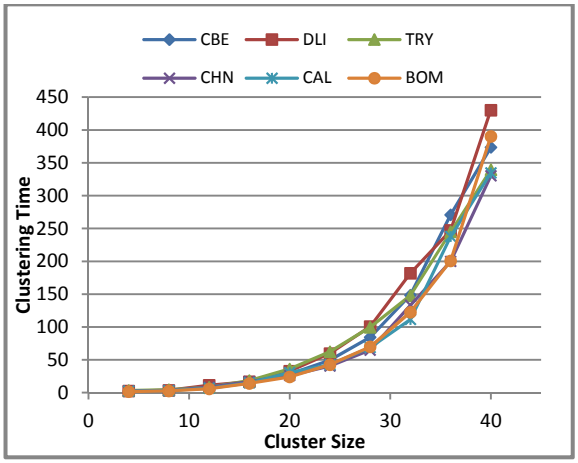

Figure 8: Clustering Time vs. Cluster Size

Figure 8 shows the relationship between clustering time overhead and cluster size for training the model. It can be observed that as the cluster size increases the clustering time increases exponentially as the number of combinations of clusters increases.

V. CONCLUSIONS

Thus the proposed method reduces the computational complexity of the model by reducing the number of support vectors at the cost of extra clustering time overhead and computational resources. The results were verified by applying the proposed method to Weather Prediction Problem of different cities using historical climate data. It was observed there was an average of 90% reduction in number of support vectors with a deviation of 5% in accuracy. Thus, the proposed method proves to be better than the conventional SVM method. The future scope of the proposed method involves extension to multi class problems and application to various time critical applications which require rapid performance with a compensated accuracy.

REFERENCES

[1] Joachims T, "Text categorization with support vector machines," Learning with many relevant features. Springer Berlin Heidelberg; Apr 21 1998.

[2] R. Ramanathan, Ponmathavan, S., Valliappan, N., Thaneshwaran, L., Nair, A. S., and Soman, K. P., "Optical character recognition for English and Tamil using support vector machines", in ACT 2009 - International Conference on Advances in Computing, Control and Telecommunication Technologies, Trivandrum, Kerala, 2009.

[3] Ganapathiraju A, Hamaker JE, Picone J, "Applications of support vector machines to speech recognition," IEEE Transactions on Signal Processing, vol.52, no.8, pp.2348-55, Aug 2004.

[4] R. Ramanathan, Nair, A. S., Sagar, V. V., Sriram, N., and Soman, K. P., "A support vector machines approach for efficient facial expression recognition", in ARTCom 2009 - International Conference on Advances in Recent Technologies in Communication and Computing, Kottayam, Kerala, 2009.

[5] Shin KS, Lee TS, Kim HJ, "An application of support vector machines in bankruptcy prediction model," Expert Systems with Applications, vol.28, no.1, pp.127-35, Jan 31 2005.

[6] Zhou X, Tuck DP, "MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data," in Bioinformatics, vol.23, no.9, pp.1106-14, May 1 2007.

[7] Xia XL, Lyu MR, Lok TM, Huang GB, "Methods of decreasing the number of support vectors via k-mean clustering," In Advances in Intelligent Computing 2005, Springer Berlin Heidelberg, (pp. 717-726) Aug 23.

[8] Adankon MM, Cheriet M, "Model selection for the LS-SVM. Application to handwriting recognition," in Pattern Recognition, vol.42, no.12, pp.3264-70. Dec 31 2009.

[9] Khemchandani R, Chandra S, "Twin support vector machines for pattern classification," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.29, no.5, pp.905-910, May 2007.

[10] Segata N, Blanzieri E, "Fast local support vector machines for large datasets," In Machine Learning and Data Mining in Pattern Recognition, Springer Berlin Heidelberg (pp. 295-310), Jul 23 2009.

[11] Tian Y, Shi Y, Liu X, "Recent advances on support vector machines research," in Technological and Economic Development of Economy. vol.18, no.1, pp.5-33, Mar 1 2012.

[12] Chih-Chung Chang and Chih-Jen Lin, LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

[13] Andrew Moore (2003).Support Vector Machines Slides [Online], Available FTP: http://www.autonlab.org/tutorials/svm15.pdf.