Voting based Extreme Learning Machine with search based Ensemble Pruning

Sanyam Shukla[†], Rajesh Wadhvani[††], Jyoti Bharti[†††] and Manasi Gyanchandani[††††]
Maulana Azad National Institute of Technology, Bhopal, India-462003

Summary
Voting based Extreme Ensemble is a majority voting based ensemble of Extreme Learning Machines. An ensemble may contain some highly correlated classifiers. Ensemble pruning is used to remove these redundant classifiers, reduce the size of ensemble. It may also increase the accuracy of ensembles by selecting subsets of classifiers that, when combined, can perform better than the whole ensemble. This paper proposes a gain based ensemble pruning technique that adds classifiers based on their diversity and contribution towards ensemble. This algorithm reduces time complexity by reducing the size of pruning set which is done by eliminating training instances that are present far away from the decision boundary. These are the instances which are classified correctly by majority of classifiers with high confidence. Results show that the proposed algorithm works equally well or even better in some cases than Voting Based Extreme Learning Machine in terms of accuracy.
Key words:
Extreme Learning Machine, Voting Based Extreme Learning Machine, Ensemble Pruning, Gain.

1. Introduction

Extreme learning machine [1] is a feed forward neural network classifier with single hidden layer. Input of ELM are features of the training dataset and their corresponding values. Output layer contains nodes for each class. In ELM, weights between input layer and hidden layer are randomly generated whereas, the weights between the hidden layer and the output layer are computed analytically. ELM gives good generalization performance and works faster than other neural networks. Ensemble [2] is a collection of classifiers generated by combining several weak/instable classifiers. Recently proposed Voting based Extreme learning machine, VELM [4] combines the output of multiple ELM classifiers which have different weights between input and hidden layer by using majority voting. Some classifiers may be redundant and highly corelated and some may give incorrect results. These faulty classifiers lead to degrade the performance of ensemble. Having redundant classifiers only increase the size and complexity of ensemble and may also decrease the accuracy of ensemble. Ensemble pruning can be used to remove these redundant classifiers. It tries to select a subset of individual classifiers to comprise the ensemble.

Various ensemble pruning algorithms have been proposed so far such as Order Based, Search Based etc. The aim of each algorithm is to select the accurate and diverse classifiers to get better or at par performance with decrease in ensemble size. [3]. Performance of VELM can be enhanced by making use of ensemble pruning. Removing individual classifiers with low performance along with high diversity among the remaining members of the ensemble is typically considered as a proper subset for designing a pruned ensemble. In the next section this paper discusses related work, i.e. various ensemble pruning techniques. After this section, this paper describes the proposed work i.e. Voting based Extreme Learning Machine with Gain based ensemble Pruning, VELMGP. After that, this paper describes the experimental setup and results obtained. The last section consists of conclusion and future work

2. Related work

Ensemble Pruning is done to reduce the ensemble size leading to reduce computational cost and space requirement. Pruning is needed for efficiency and performance of the ensemble. Ensemble pruning can also be used to increase accuracy as proved by [8]. So, it is possible to have a small and yet strong ensemble. The two most popular methods for Ensemble pruning are order based and search based:

1) Order Based: Each classifier has different behavior towards the data. The idea behind this method is to order classifiers on the basis of evaluation function chosen for pruning. Classifiers having rank above a given threshold are selected to make pruned ensemble. For example, [10] and [11] order classifiers by calculating margin distance for each instance and collective agreement for each classifier respectively. Kappa statistic [12] is an agreement based approach in which kappa statistics is calculated for each classifier to design the pruned ensemble.
2) Search Based: In this method, instead of ranking, heuristic search is applied to find the best classifiers in order to improve efficiency of the bagging ensemble (Forward Selection). Classifiers are selected till there is increment in performance of subset. One method

Manuscript received March 5, 2017
Manuscript revised March 20, 2017

could be to find the worst classifier and remove it from the ensemble (Backward Elimination). There are several metrics to search for a classifier, such as diversity [13], in which classifiers having more diversity is assumed to be more capable of enhancing ensemble performance than others. Similarly, uncertainty weighted accuracy(UWA) [14] proposes a measure, which takes into account the uncertainty of the decision of the current ensemble for designing pruned ensemble. [15] formulates the ensemble pruning problem as semi definite programming problem. Also, class specific soft voting [16] uses condition number of matrix, which reveals the stability of linear equation, to identify the stable ELM classifiers in the ensemble. This paper has also taken into account the diversity of classifiers in terms of gain to the current ensemble. Searching based methods provide a better classification performance than ordering based methods. This paper introduces a search based greedy approach which significant reduces the complexity of ensemble.

III. Proposed Work

Efficiency of VELM may be improved by selecting optimal accurate and diverse classifiers. This paper focuses on both complexity of solution and efficiency of the ensemble. Complexity can be reduced by reducing the size of pruning set. This is done by eliminating instances that lie far away from the decision boundary and are strongly related to a specific class. Here, it is assumed that an instance classified as belonging to a specific class by predefined percentage of classifiers will surely be correctly classified. For example, if an instance is classified as of class ’A’ by all the classifiers, then that instance may be eliminated for pruning purpose. We have done ensemble pruning using reduced training set. This work removes all the training instances which are classified as a specific class by more than a threshold number of classifiers. This work sets this value to 80 % as this threshold gives better results which was observed during experimentation. Proposed Pruning algorithm is based on greedy strategy. Classifiers are added to the ensemble subset using forward selection. A classifier is only added if it increases the performance of sub-ensemble. The increase in correctly classified sample is termed as gain. All the classifiers which are not part of pruned ensemble are explored to find the classifier having maximum gain. Gain must be positive and maximum for the classifier to be added to pruned ensemble. Classifiers having negative gain will never be the part of pruned ensemble and will be discarded. When the maximum gain becomes negative the algorithm terminates. A classifier in an ensemble must be able to affect the ensemble in an efficient and effective way. This ability of the classifier can be calculated as gain. A relation between current pruned ensemble and classifier is calculated as explained in the following section. There are four categories of prediction of any classifier i.e. True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Then, gain is calculated by the following formula

$$\text{gain} = (TP + FP) - (TN + FN)$$

True Positive and True Negative are correct classifications Whereas, False Negative and False Positives are incorrect classifications. Gain will be maximum when incorrect classification is minimal. By using this formula for calculating gain, the classifier having the maximum positive effect on the pruned ensemble can be found and then that classifier is added to the pruned ensemble. In each iteration, the classifier having maximum gain relative to the current pruned ensemble is added to the pruned ensemble. Gain of classifier increases when it supports the correct class of instance or correctly classifies an incorrect class, otherwise decreases. Gain can be both positive and negative but classifiers having negative gain are not added in pruned ensemble as they are assumed to decrease the performance of ensemble. The pseudocode of the proposed algorithm is given below:

Algorithm 1 Gain Based Ensemble Pruning, VELM GP

Begin.
Generate NCE classifiers using ELM algorithm.
Select classifier with maximum G-Mean as base classifier and add to pruned ensemble.
Find the pruning dataset by removing instances from training data which are classsified with confidence score greater than threshold.
for number of instances **do**
    Calculate fractions of classifiers predicting each class.
    if any fraction > threshold **then**
    Remove instance from dataset.
    end if
end for
Initialize gain = 1.
for gain > 0 **do**
    for all remaining classifiers **do**
    Calculate gain with respect to current pruned
    ensemble.
    end for
    Find classifier with maximum gain and update gain.
    if gain > 0 **then**
    Add classifier to pruned ensemble.
    end if
end for
End.

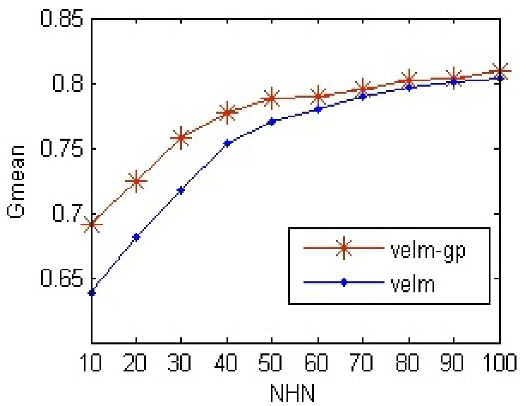

Fig. 1 Gmean of VELM vs VELM GP for phoneme dataset

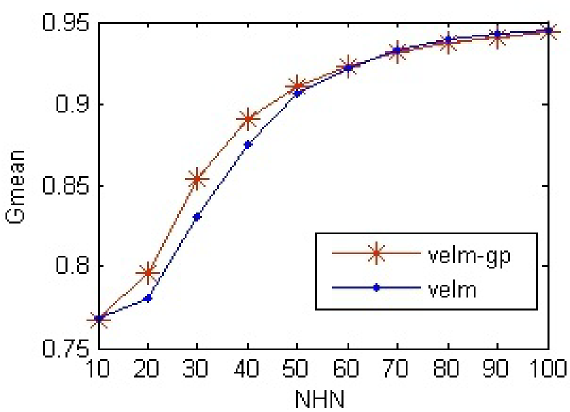

Fig. 2 Gmean of VELM vs VELM GP for ring dataset

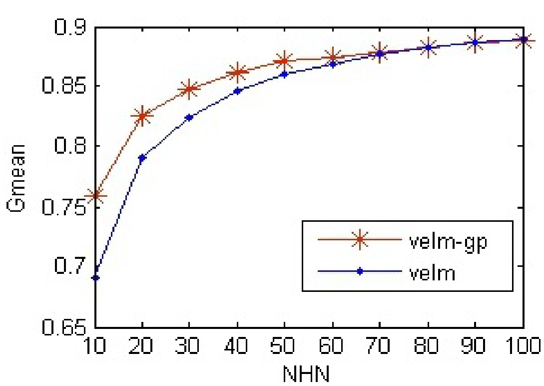

Fig. 3 Gmean of VELM vs VELM GP for spambase dataset

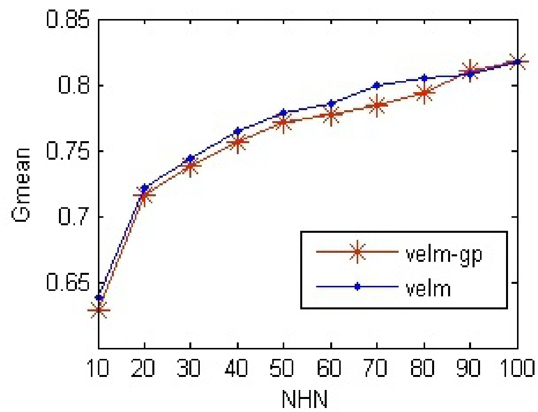

Fig. 4

Gmean of VELM vs VELM GP for vehicle dataset

TABLE 1: Comparison between VELM and VELM_GP on the basis of G-mean and Number of Hidden Neurons

Data set	Size of PE	VELM_GP		VELM	
		NHN	G-mean	NHN	G-mean
appendicitis	18	10	0.67871	10	0.7001
banana	8	100	0.89835	100	0.8981
Bupa	6	20	0.69752	20	0.69724
Ecoli	1	10	0.98644	10	0.98644
Glass	29.4	60	0.77877	60	0.77856
haberman	14	20	0.46189	20	0.49074
hayes-roth	17.4	30	0.70864	20	0.70864
Heart	21.4	40	0.82683	20	0.83258
ionosphere	29	80	0.89195	100	0.89388
Iris	1	10	1	10	1
Monk	28	70	0.9717	70	0.97253
newthyroid	14	30	0.86373	20	0.86149
phoneme	1	100	0.80835	100	0.80388
Pima	1	40	0.70154	40	0.70474
Ring	38	100	0.85392	100	0.83095
saheart	1	30	0.64806	20	0.64201
Sonar	45	70	0.84239	70	0.84351
spambase	25	100	0.88872	100	0.88921
spectheart	40	100	0.40228	100	0.39242
titanic	5	10	0.67083	10	0.67083
vehicle	8	100	0.8174	100	0.8168

IV. Experimental Setup

The proposed work is evaluated using 21 binary class datasets, downloaded from the Keel-data set Repository [17]. The data sets in Keel Repository are available in 5 fold cross validation format i.e. for each dataset we have 5 training and testing sets. Gmean is the best measure for comparison in case of class imbalanced as well as balanced classification problems. Gmean can be calculated as follows:

$$\text{Gmean} = \prod(\text{Recall}_i)^{1/c}$$

where i varies from 1,2.....,c and c is equals to the number of classes.

VELM is treated as the special case of VELM GP, when all the classifiers participate in majority voting. Results presented in this paper are averaged over 50 trials. In each trial 50 ELM classifiers are generated and the final outcome of VELM is the majority voting of all these 50 classifiers. Optimal number of hidden neurons(NHN) for VELM has been found by varying NHN from [10, 20 100]. Overall accuracy of the pruned ensemble, PE is calculated by conducting voting of selected P_{NCE} classifiers. In any pruning technique, we will get different overall accuracy corresponding to the choice of P_{NCE} and NHN. Overall Gmean of proposed classifier for various datasets is shown in Table I. It can be observed from Table I that VELM GP is better than VELM in terms of size of ensemble. VELM GP performs equally well or even better than VELM for more than half of datasets. Accuracy of VELM GP for ring, spambase, phoneme and vehicle dataset is more than VELM. Their corresponding graphs have been shown in Fig 1-4. Results obtained shows that specified threshold is appropriate for reducing size of dataset. Removing instances does not affect the results but significantly reduces time complexity of algorithm. TABLE-I shows that size of pruned ensemble is significantly reduced with more ar at par performance.

V. CONCLUSION

This work proposes a new classifier, VELM GP which is an extension of VELM. VELM gives better performance than ELM with increased computational and memory requirement. VELM GP applies search based ensemble pruning based on forward selection scheme to reduce this overhead. This proposed algorithm uses subset of training data for ensemble pruning. In VELM GP first classifier with highest Gmean is chosen as pruned ensemble. In VELM GP difference between number correctly classified instances and number incorrectly classified instances is used as a search metric to find the next classifier to be added in the current pruned ensemble. The results show that the proposed algorithm gives better or at par performance compared to VELM with significant reduction in ensemble size.

References

[1] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. Neurocomputing, 70(1):489– 501, 2006.
[2] Thomas G Dietterich. Ensemble methods in machine learning. In Multiple classifier systems, pages 1–15. Springer, 2000.
[3] Muhammad AO Ahmed, Luca Didaci, Giorgio Fumera, and Fabio Roli. An empirical investigation on the use of diversity for creation of classifier ensembles. In Multiple Classifier Systems, pages 206–219. Springer, 2015.
[4] Jiuwen Cao, Zhiping Lin, Guang-Bin Huang, and Nan Liu. Voting based extreme learning machine. Information Sciences, 185(1):66–77, 2012.
[5] Guang-Bin Huang. What are extreme learning machines? filling the gap between frank rosenblatts dream and john von neumanns puzzle. Cognitive Computation, 7(3):263–278, 2015.
[6] Sanyam Shukla and R. N. Yadav. Unweighted class specific soft voting based ensemble of extreme learning machine and its variant. International Journal of Computer Science and Information Security, 13(3):59, 2015.
[7] Robi Polikar. Ensemble based systems in decision making. Circuits and Systems Magazine, IEEE, 6(3):21–45, 2006.
[8] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: many could be better than all. Artificial intelligence, 137(1):239–263, 2002.
[9] Rich Caruana, Alexandru Niculescu-Mizil, Geoff Crew, and Alex Ksikes. Ensemble selection from libraries of models. In Proceedings of the twenty-first international conference on Machine learning, page 18. ACM, 2004.
[10] Li Guo and Samia Boukir. Margin-based ordered aggregation for ensemble pruning. Pattern Recognition Letters, 34(6):603–609, 2013.
[11] Lior Rokach. Collective-agreement-based pruning of ensembles. Computational Statistics & Data Analysis, 53(4):1015–1026, 2009.
[12] Dragos D Margineantu and Thomas G Dietterich. Pruning adaptive boosting. In ICML, volume 97, pages 211–218. Citeseer, 1997.
[13] Ludmila I Kuncheva and Christopher J Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine learning, 51(2):181–207, 2003.
[14] Ioannis Partalas, Grigorios Tsoumakas, and Ioannis Vlahavas. An ensemble uncertainty aware measure for

directed hill climbing ensemble pruning. Machine Learning, 81(3):257–282, 2010.

[15] Yi Zhang, Samuel Burer, and W Nick Street. Ensemble pruning via semi-definite programming. The Journal of Machine Learning Research, 7:1315–1338, 2006.

[16] Jingjing Cao, Sam Kwong, Ran Wang, Xiaodong Li, Ke Li, and Xiangfei Kong. Class-specific soft voting based multiple extreme learning machines ensemble. Neurocomputing, 149:275–284, 2015.

[17] J Alcal´a, A Fern´andez, J Luengo, J Derrac, S Garc´ıa, L S´anchez, and F Herrera. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. Journal of Multiple-Valued Logic and Soft Computing, 17(2-3):255–287, 2010.