Proc. Int. Conf. on Computational Intelligence and Information Technology, CIIT

ELSEVIER

Heuristic based Improvements for Effective Random Forest Classifier

Vrushali Y Kulkarni [1] Dr P K Sinha[2] Asshu Singh[3] Farah Shaikh[3] Mehul Mittal[3]

[1] PhD Student, COEP, Pune, India {kulkarnivy@rediffmail.com}
[2] Director, HPC, CDAC,Pune, India
[3] UG Student, MIT, Pune, India

Abstract. Random Forest is an ensemble supervised machine learning technique. Based on bagging and random feature selection, number of decision trees (base classifiers) is generated and majority voting is taken for classification. In this paper, we are presenting some heuristic based improvements towards effective learning of Random Forest classifier. These efforts include disjoint partitions of datasets for learning of base trees, reducing depth of base trees by avoiding repetitive selection of attributes, and selecting smaller subsets of attributes for split at each node. The results of our work are encouraging and there is future research scope in this direction.

Keywords: Data Mining, Classification, Ensemble, Random Forest

1 Introduction

Random Forest is an ensemble supervised machine learning algorithm which is comparable with bagging and boosting. Machine learning techniques have applications in the domain of Data Mining. Classification and Prediction are commonly used tasks in Data Mining where a huge amount of past data is analyzed to predict future trends or values. In this process, a number of input variables named as predictors are used to predict the output variable which is commonly known as target. In case where target variable is nominal, the process is known as Classification and where target variable is numeric, it is known as Regression.

Random Forest is an ensemble in which base classifiers are Decision Trees. It is proved theoretically and empirically [2] that an ensemble always gives better accuracy than the individual base classifier. Bagging [1] and Boosting are fundamental ensemble techniques. Bagging works on the principle of bootstrap samples, while boosting works by assigning votes to input samples on the basis of their accurate prediction. Bagged ensembles can be built in parallel while boosting is a sequential process. Random Forest is based on the concept of Bagging plus random selection of features. As ensemble consists of multiple classifiers, the time required to learn using ensemble is more as compared to a single classifier. In this paper we are presenting some heuristic based methods to improve learning of Random Forest classifier. The experiments made are use of disjoint partitions of datasets to generate base decision

trees, controlling depth of individual decision tree, and reducing the size of attribute set for selection of split at each node. The empirical results are encouraging so that we can continue our work further in this direction.

The paper is organized in following way: Section 2 gives brief introduction to Random Forest and related work. Section 3 presents Methods and Algorithms for the experiments performed. Section 4 presents Results and Discussions and section 5 gives Concluding remarks.

2 Related Work

A Random Forest is a classifier consisting of collection of tree structured classifiers $\{h(x, \Theta_k)\ k=1, 2, \ldots.\}$ where the $\{\Theta_k\}$ are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x [3].

Each tree in Random forest is grown in the following way: If the number of records in the training set is N, then N records are sampled at random but with replacement, from the original data, this is bootstrap sample. This sample will be the training set for growing the tree. If there are M input variables, a number m << M is selected such that at each node, m variables are selected at random out of M and the best split on these m is used to split the node. The value of m is held constant during forest growing. Each tree is grown to the largest extent possible. There is no pruning. The Generalization error of Random Forest is given as,

$$PE^{*} = P_{x,y}(mg(X,Y)) < 0$$

The margin function is given as,

$$mg(X,Y) = av_k I(h_k(X) = Y) - max_{j \neq Y} av_k I(h_k(X) = j)$$

If ρ is mean value of correlation between base trees, an upper bound for generalization error is given by,

$$PE^{*} \leq \rho (1 - s^2) / s^2$$

Hence, to yield better accuracy in Random Forest, the base decision trees are to be diverse and accurate, which is also tested empirically in [10].

In [5], we have presented in detail the survey of work done related to Random Forest classifier along with Taxonomy and Comparative study. The work related to improvement in accuracy uses different split measures in generating Random Forest [4] and performs weighted voting [12]. Also use of Random Forest itself as base classifier is tested in [11]. For improving performance of Random Forest, researchers are trying to reduce number of base decision trees from forest such that the accuracy of this pruned forest is same as that of the original forest. Most of the work in this direction [6] [7] [8], first over produces the forest and then removes trees which are contributing less in the overall accuracy; thus is static in nature. In [9], one by one trees are generated, and added in the forest or discarded based on a predefined accuracy curve. In [13], Random Forest is generated using boosting principle and it is dynamic in nature.

3 Methods and Results

The aim of this research work is to make some improvements in Random Forest so that the time taken to learn the forest can be reduced. All experiments are carried out on datasets from UCI Machine Learning repository. Each dataset is divided into training set (2/3rd) and testing set (1/3rd). The accuracy is noted down with varying number of trees. Results are compared with original Random Forest from Weka.

3.1 Experiment 1

Disjoint partitions of dataset to build base decision tree in Random Forest. The first experiment done is to generate diverse base decision trees. Brieman suggested that to yield less generalization error and hence to get more accuracy, the base trees are to be more diverse, i.e. they should predict differently. For this purpose, we are generating disjoint partitions of original dataset, i.e. for each tree we are selecting fixed number of samples from original dataset without replacement. The size of each partition is same and is decided by the number of trees in Random Forest. Though each tree is getting less number of samples here, the sample set for learning any two trees is entirely different and hence the trees are less correlated. We call this new algorithm as Disjoint Partitioning Random Forest (DPRF), but not immediately publishing it as we are still doing some improvements in it those we have mentioned in future work. This experiment is tested with 20 datasets and results are compared with original Random Forest. Here results on 5 datasets are presented in table 1 due to space limitation.

3.2 Experiment 2

Controlling the depth of base decision trees by avoiding repetitive selection of attributes. In original Random Forest by Brieman, for base decision trees, at each node $\sqrt{m}$ attributes out of total m attributes are selected and the best split among them is decided by using Gini index. This process gets repeated at every node and hence attributes have no control over depth of decision tree. The depth of decision tree is governed by a parameter "nodesize". The node is treated as leaf node if it has "nodesize" instances, and a default value for nodesize is considered as 5. Brieman has stated that this type of tree creation reduces biasing. We have experimented to control this depth of individual base tree through attribute selection process. Once an attribute is selected at a node, then it is discarded from the entire set of attributes. Hence there is no repetitive selection of attributes. This process stops the tree creation when every attribute is considered once. This leads to base trees of reduced depth. The accuracy of original RF and experiment 2 are recorded for different values of number of trees.

3.3 Experiment 3

Heuristic approach to select subset of attributes at each node to generate base decision trees. As per Brieman, Random Forest gives good accuracy if the base decision

trees are less correlated. Also Brieman has proved empirically in his paper [3] that increasing number of attributes (for deciding best split) at each node does not increase strength, the strength remains almost constant after a value of 4; but it increases correlation. Hence we are trying to select less correlated features by taking smaller subsets of attributes. A heuristic for this is to have different subsets of attributes for selection at each node. To achieve a balance between strength and correlation, at each node creation, we have randomly taken subset of total m attributes as (2/3*m) where m is total number of attributes. Then we selected $\sqrt{m}$ attributes from this subset, as it is done in original Random Forest. In this way, we are selecting attributes at each node from different subsets and there is a chance that $\sqrt{m}$ attributes at each node will be different though they are not disjoint. This leads to more diverse tree creation in Random Forest which can improve accuracy. Table 2 shows results of this experiment for a few datasets.

4 Discussions

Experiment 1 was expected to give good results for large datasets as disjoint partitioning of dataset in large datasets can provide sufficient number of samples for each tree. The empirical results are showing that experiment 1 is not giving good results for small datasets (i.e. number of samples less that 1000). It is also not giving good results for dataset of large size but with more number of classes. But it is giving good results for datasets of moderate size and less number of classes. The bar-graph in figure 1 shows that out of all datasets we have tested, 75% of times experiment 1 results are either same as original Random Forest or better. At primary level, our conclusion is that experiment 1 results are not getting affected by the nature of dataset, i.e. whether it is balanced or imbalanced; but we need to do more experimentation on this. With experiment 1, the learning time for Random Forest is reduced. Our future work to continue in this direction is to increase size of each disjoint partition by randomly sampling instances with replacement inside the partitions (similar to bootstrap); this will help in increasing the strength of individual tree. Also we will test experiment 1 with large datasets and less number of classes. The aim of experiment 2 was to limit the depth of each tree by avoiding repetitive selection of attributes in generating base decision trees. The empirical results show that this experiment is not giving good results and hence the results are not presented here. The reason analyzed is that at the deeper levels of tree creation, limited attributes are available. The attributes available at those nodes may not be generating pure partitions and hence reducing the overall classification capability of the decision tree. Experiment 3 is giving good results for datasets where number of attributes is moderate (i.e. in the range 10-50). With datasets having large number of attributes, the results are not good as selecting 2/3*m gives a subset of large size which reduces strength and increases correlation [3]. The future work in this direction is to test subsets of (1/2*m) or (1/3*m) for large datasets. The results of this experiment are not getting affected by number of classes. The bar-graphs in figure 2 shows that for 77% out of total readings, our results are either same or better than that of original Random Forest.

5 Conclusion

The heuristic based approaches presented in this paper are carried out with a goal of achieving effective learning using Random Forest classifier. With all the experiments, we are trying to achieve learning of base decision tree with either less number of instances or less number of attributes. This will help in learning of individual tree and in turn the entire forest in lesser time. The encouraging results of two of our experiments are leading us to do further work in this direction. The future work is to improve the concept of disjoint partitioning by random sampling with replacement within the partition, selecting smaller subsets for datasets with large attributes, and improvements related to datasets with multiple classes.

Table 1. Results (% Accuracy) of experiment 1

ataset	50 trees		100 trees		150 trees		200 trees		250 trees	
	RF	Exp1	RF	Exp1	RF	Exp1	RF	Exp1	RF	Exp1
mushroom	98.04	98.44	93.46	93.83	90.32	91.32	93.79	94.53	88.99	88.99
Musk 2	85.58	86.08	85.17	83.71	86.03	86.53	87.13	87.94	85.49	85.35
Onehr	97.27	97.15	96.8	96.68	96.44	96.44	96.8	96.8	96.8	96.8
Segment	80.64	84.28	62.98	60.9	47.14	47.66	43.11	48.96	38.6	39.48
splice	64.06	63.59	54.93	64.34	51.9	52.21	50.7	51.45	53.5	53.15

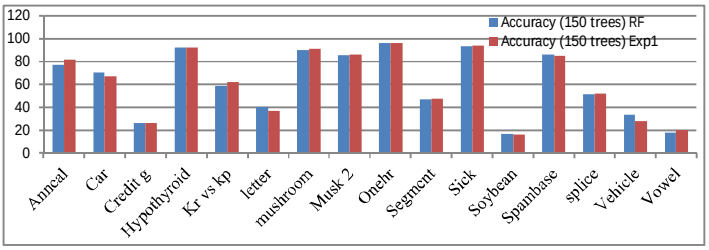

Fig. 1. Bar graph showing comparative results of RF and Experiment 1

Table 2. Results (% Accuracy) of experiment 3

Dataset	50 trees		100 trees		150 trees		200 trees		250 trees	
	RF	Exp1	RF	Exp1	RF	Exp1	RF	Exp1	RF	Exp1
Audiology	73.33	77.33	76.0	76.0	66.66	69.33	66.66	68.0	73.33	74.66
letter	95.82	95.91	95.69	95.93	95.99	96.36	96.20	96.39	95.69	96.12
Musk 2	99.27	97.99	99.49	98.63	99.09	99.54	99.31	98.99	99.13	99.09
Soybean	91.62	92.07	90.30	90.30	92.95	94.71	93.39	94.27	90.30	89.86
BC	69.47	71.57	67.36	70.52	75.78	75.78	66.31	67.36	72.63	71.57

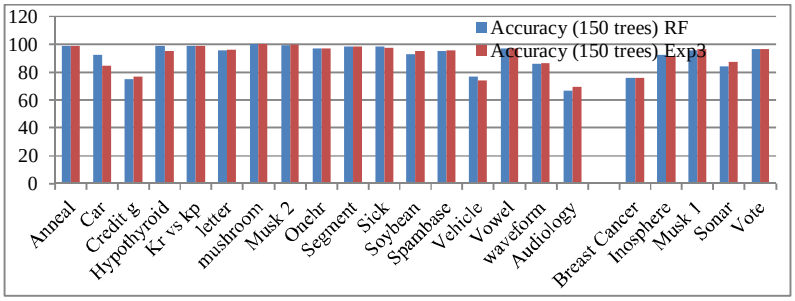

Fig. 2. Bar graph showing comparative results of RF and Experiment 3

References

1. Leo Breiman, " Bagging Predictors" , Technical report No 421, September 1994
2. David Opitz, Richard Maclin, "Popular Ensemble Methods: An Empirical Study", Journal of Artificial Intelligence 11, 169-198, (1999)
3. Leo Brieman, "Random Forests", Machine Learning, 45, 5-32, (2001)
4. Marko Robnik, Sikonja, "Improving Random Forests", J F Boulicaut et al (eds): Machine Learning, ECML 2004 Proceedings, Springer, Berlin, (2004)
5. Vrushali Y Kulkarni, pradeep K Sinha, Random Forest Classifiers: A Survey and Future Research directions, submitted to ACM Computing Surveys in Jan 2012
6. Heping Zhang, Minghui Wang, "Search for the smallest Random Forest", Statistics and Its Interface Volume 2, pp 381-388, (2009)
7. Simon Bernard, Laurent Heutte, and Sebastien Adam, "On the Selection of Decision Trees in Random Forest", Proceedings of International Joint Cobference on Neural Networks, Atlanta, Georgia, USA, June 14-19, pp 302-307, (2009)
8. P. Latinne, O. Debeir, C. Decastecker, "Limiting the number of trees in Random Forest", MCS, UK (2001)
9. E Tripoli, D Fotiadis, G Manis, " Dynamic Construction of Random Forests: Evaluation using Biomedical Engineering Problems", IEEE (2010)
10. S Bernard, L Heutte, and S Adam, Towards a Better Understanding of Random Forests Through the Study of Strength and Correlation, ICIC Proceedings of the Intelligent Computing 5th International Conference on Emerging Intelligent Computing Technology and Applications, (2009)
11. Boinee P, Angelis A and Foresti G, Meta Random Forest, International Journal of Computational Intelligence 2, (2006)
12. Tsymbal A, Pechenizkiy M and Cunningham P, Dynamic Integration with Random Forest, ECML, LNAI, 801-808, Springer-Verlag (2006)
13. Simon Bernard, Sebastein Adam, Laurent Heutte, Dynamic Random Forests, Pattern Recognition Letters, volume 33 issue 12, 1580-86 (2012)