

Chapter 38

Ensemble Method Combination: Bagging and Boosting



Jyoti Deshmukh, Mukul Jangid, Shreeshail Gupte, Siddhartha Ghosh and Shubham Ingle

1 Introduction

Machine learning using a single model is more prone to error and is less efficient. Error is disintegrated into 3 terms: bias, variance and irreducible error term. Bias is contributing error due to model simplifying the learning function and variance is contributing error due to learner making the function more complex. Ensemble learning is a better method of utilizing the notion that combining the output of different models produces more accurate results. The success in ensemble learners lies when the sub-ensemble classifiers are able to overcome the failure of each other and generate diverse predictions. Two popular ensemble learning methods: boosting and bagging are applied successfully to classification problems.

Boosting and bagging are homogenous ensemble methods. They use the same base learning algorithm but the dataset distribution is changed. Data can be distributed by sampling the instances, another approach is by dividing the data into feature subsets. Features are selected by evaluating which feature are more relevant for classifying the output. Features can be evaluated individually or as a subset. Creating subsets requires the grouping of features in different sizes. Evaluating a subset requires more computation and time.

J. Deshmukh · M. Jangid (✉) · S. Gupte · S. Ghosh · S. Ingle
Department of Computer Engineering, Rajiv Gandhi Institute of Technology, Andheri West,
Mumbai, Maharashtra 400053, India
e-mail: mkljngd@gmail.com

J. Deshmukh
e-mail: [jyotideshmukh11@gmail.com](mailto: jyotideshmukh11@gmail.com)

S. Gupte
e-mail: shreeshail04@gmail.com

S. Ingle
e-mail: shubhamingle@yahoo.com

© Springer Nature Singapore Pte Ltd. 2020
H. Vasudevan et al. (eds.), *Advanced Computing Technologies and Applications*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-15-3242-9_38

In the proposed method boosting is used in bagging algorithm as a base learner. Boosting itself will use its own base learner C4.5 [1]. Boosting is affected by variance error. Using multiple boosting on different data subsets makes the combined model have less variance. In this, the number of instances given to each boosting algorithm remains the same. In the results section, the results of the proposed method are compared with the combined method and individual bagging and boosting on standard datasets.

Ensembling is a process of merging at least two procedures of homogeneous or heterogeneous nature called base learners. This increases the robustness of the system which accepts the predictions from all the base learners. It is like a committee among various subcommittees to arrive at a conclusion on the problem. All of them have a different outlook on the problem thus a dissimilar mapping function from the problem statement to the favorable outcome. As a result, they have to make different predictions on the problem according to their own outlooks.

The final decision is made by accounting on all the predictions. Hence the output decision will be precise, less bias and robust. If one of these sub-committees decided this alone, the final decision might be contradictory. A single learner might not be able to predict with great accuracy. But the numerous feedbacks of multiple learners usually increase the accuracy. In the regression problem or the classification problem, the mean of probability predictions from the models are calculated by Averaging.

Majority votes consider the prediction with the highest vote count from various models predictions while predicting the results of a classification problem [2].

In weighted average, different weights are taken which are to be applied for predictions from various models, then considering the mean which results in providing high or low importance to specific model output.

2 Related Work

2.1 *Bagging as an Ensemble*

The paper “A Bagging Method using Decision Trees in the Role of Base Classifiers” proposed by the authors Kristna Machov, Frantiek Bark, Peter Bednr described a list of experiments with bagging—a process that can be used to enhance the results of the classification algorithm [3]. Their use of this process aims at classification algorithms implying decision trees. Bagging procedure improves the performance of classification using the minimum number of the decision trees.

2.2 *Boosting as an Ensemble*

The authors Thakkar et al. Proposed a paper on “Boost a Weak Learner to a Strong Learner Using Ensemble System Approach” which demonstrated the weak learner’s potential to reduce the rate of an error on the testing data and the ability of boosting algorithm to reduce the error rate of the weak learner [4]. In the experiment, they utilized decision stump as a weak learner (classifier) and by utilizing the boosting approach, the output denotes the enhancement in the classifier’s accuracy. The boosting meta-algorithm is a simple and well-organized model building strategy. Boosting motivates new model to become specialists, for instance, handle wrongly by earlier ones. A model’s contribution is weighted by its performance than giving equal weight to all the methods. Boosting algorithms creates various models from a dataset, consisting of some model builder techniques such as a decision tree builder which may not be a good model builder. The basic idea of boosting is that each entity in the dataset is mapped with a weight. If a model incorrectly classifies the entity, then a series of models are built and the weights are increased (boosted). The final model is then a sum of different models built from the sequence of models in which each model’s output weighted by little score.

2.3 *Techniques for Combining Bagging and Boosting*

In [5], the authors proposed a methodology in which the expected error of a learning procedure on a specific target function having training set size of 3 components: [1] A bias term computing how close the mean classifier makes by the learning procedure will be to the target function. [2] A variance term computes the amount of the every learning algorithms estimates how often they disagree. [3] A variable that measures the least classification error that is related with the Bayes optimal classifier for the target function. For enhancing the prediction of a classifier, authors suggest arranging bagging and boosting methodology with sum rule voting (Vote BB). All sub-ensemble gives a confidence value for each candidate class. In the proposed method, the voters express the degree of their preference as confidence weighs the chances of sub ensemble prediction. Later, confidence values are summed for each candidate and the candidate with the maximum sum proves to be superior.

MultiBoosting [6] is an another method proposed by Webb G. I. which is acknowledged as wagging committees formed by AdaBoost. Wagging is a aberrant version of bagging, as wagging makes use of re-weighting for every training example and bagging uses resampling to receive the datasets.

BagBoo A Scalable Hybrid Boosting and Bagging Model [7] proposed by Gorodilov et al. is another method for combining bagging and boosting into a hybrid approach. The only important change from bagging and boosting is that they bagged boosted models which allowed the resulting model to be quite a bit more powerful.

3 Proposed Methodology

A comparative study on bagging [8–10] and the boosting [11, 12] algorithms is performed and combined them to eliminate both of their flaws. In the proposed method, \hat{f} features are selected from the originally present f features. Then s number of subsets is randomly sampled without replacement from the total subsets generated of \hat{f} features. The same number of samples are present in all the subsets. The boosting algorithm is fitted on each training subset giving s learned hypothesis function. Prediction is made on test tuple by all the models. For each class a confidence value which is the probability between 0 and 1 is returned, this is summed and the maximum confidence class is predicted output. Algorithm 1 presents Random attribute subset selection

Algorithm 1: Random attribute subset selection

a. **Input.**

Set all training sets $\{X_i, Y_i\}$ where $i = 1$ to n
 X consists of different features i.e. (X_1, X_2, \dots, X_k) , Y is the target variable

b. **Procedure.**

Apply Base Learning Algorithm as AdaBoost
 Generate all combinations of attributes of length m
 Subset: Select s number of random subsets

c. **For i to s do**

$M[i] := \text{AdaBoost}(\text{subset}[i])$
 end for

d. **Output.:** $\text{argmax} \Sigma_i M[i]$

Figure 1 describes the workflow of the proposed methodology which starts with the training of the dataset including cleaning by filtering the dataset. N bags of the dataset are created, each portraying random attributes and applying boosting on each bag and later combining the results to give the predicted value.

4 Experiment Results

Experiments are performed in Python. The experiments are carried out on 64-bit Core i5–7th generation 3 GHz processor, 8 GB RAM. In the experiment, the aim is to compare the proposed method of integrating boosting consisting random feature subset selection with bagging, boosting and sum vote combining [5] classifiers. The 22 datasets of various domains are selected from UCI repository [13] containing a different number of instances, types of features: numerical as well as categorical and

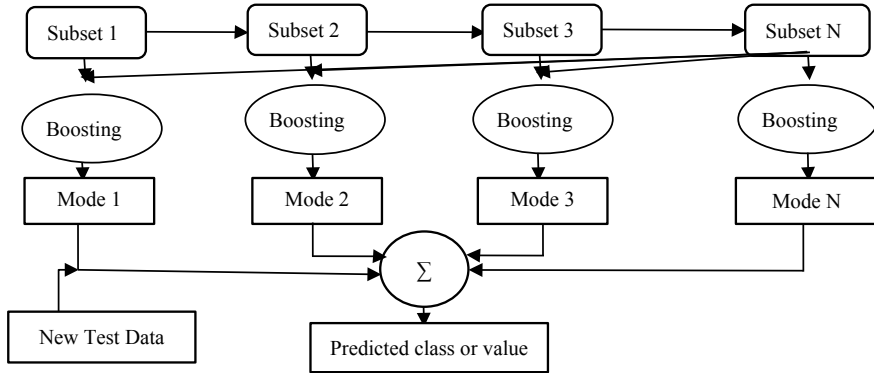


Fig. 1 Flow diagram of proposed method

class distributions of two till ten classes. The base sub-classifier CART algorithm induced decision tree is used for all the methods. In the literature [5] the authors compared several base classifiers like Decision tree, Decision stump, Bayesian algorithm and Rule Learner on various datasets and concluded that Decision tree gave the best results compared to the others. As referring to the literature, Decision Tree is used as our base estimators in the proposed method. Equal count of base sub-classifier is combined in all the models for making the results comparable equally. The accuracy is estimated by taking an average of tenfold stratified cross validation. The database is split into ten folds by sampling and rearranging without replacement. The classifier algorithm is tested on each fold and trained on the remaining nine folds. Other evaluation metrics—recall, precision, and specificity are computed for each database.

Experiment is first tested by changing the number of features selected and the number of subsets selected to decide the proportion of randomly picked feature subsets from feature vector giving the optimal result. The time taken for running the proposed algorithm takes more time than bagging and boosting due to algorithm running on additional data subspaces. Increasing the number of bags or subsets, consumes more time for training. It can be parallelized easily to decrease the running time.

Table 1 describes databases which are used to express the count of instances, categorical features, numerical features and classes. Here some well-known datasets are used for the comparison of the results.

Various experimentation is performed on all the datasets with a various count of base estimators. It is found that when the count of base classifiers is 5, the optimum result is achieved with least time and space complexity as shown in Table 2.

Bagging, boosting and combining using vote B and B are executed on the different datasets, for each method accuracy, recall, precision and specificity quality evaluation metrics are estimated and listed in Tables 3, 4 and 5.

Figure 2 visually represents the results of vote bagging and boosting ensemble on datasets.

Table 1 Description of datasets

Dataset	Instance	Categorical features	Numerical features	Classes
Anneal	898	32	6	6
Australian	690	8	6	2
Breast-cancer	699	0	9	2
Car	1748	6	0	4
Ecoli	336	1	7	8
Glass	214	0	10	7
Haberman	306	0	3	2
Heart-statlog	270	0	13	2
Heart-h	297	7	6	5
Hepatitis	155	13	6	2
Liver	583	1	9	2
Mobile	2000	5	15	3
Nursery	12,960	8	0	5
Somhappy	143	6	0	2
Teaching assistant	151	4	1	3
Tic-Tac-Toe	958	9	0	2
Travel-insurance	63,326	6	4	2
Vertebra	310	0	6	4
Wifi-localization	2000	0	7	4
Wine	178	0	13	3
Wine-red	1599	0	11	10
Wine-white	4899	0	11	10

Table 2 Comparative analysis of the number of base classifiers with Breast-cancer, Car and Haberman datasets

No. of base classifiers	Breast-cancer	Car	Haberman
5	0.9557	0.8825	0.7192
55	0.9528	0.8663	0.6960
105	0.9500	0.8674	0.6794
155	0.9514	0.8692	0.6796
205	0.9500	0.8709	0.6926
255	0.9500	0.8657	0.6991
305	0.9486	0.8663	0.6990
355	0.9486	0.8657	0.6926
405	0.9486	0.8657	0.6926
455	0.9486	0.8640	0.6958
505	0.9486	0.8634	0.7024

Table 3 Result of bagging classifier using decision tree

Dataset	Bagging			
	Accuracy	Recall	Precision	Specificity
Anneal	0.8872	0.7490	0.7608	0.9395
Australian	0.8347	0.8343	0.8352	0.8343
Breast-cancer	0.9472	0.9441	0.9428	0.9441
Car	0.8592	0.7959	0.8497	0.9312
Ecoli	0.8406	0.6792	0.7199	0.9630
Glass	0.9263	0.9163	0.9182	0.9840
Haberman	0.6446	0.5310	0.5263	0.5310
Heart-statlog	0.7704	0.7667	0.7722	0.7667
Heart-h	0.7747	0.7721	0.7829	0.7721
Hepatitis	0.8250	0.7138	0.7157	0.3138
Liver	0.6775	0.5966	0.6181	0.5966
Mobile	0.8740	0.8740	0.8765	0.9580
Nursery	0.7598	0.7541	0.7858	0.9151
Somhappy	0.4638	0.4733	0.4742	0.4733
Teaching assistant	0.6627	0.6617	0.6625	0.8307
Tic-Tac-Toe	0.8698	0.8777	0.8845	0.8777
Travel-insurance	0.9774	0.5108	0.5230	0.5108
Vertebra	0.8387	0.7911	0.8137	0.9227
Wifi-localization	0.9730	0.9730	0.9745	0.9910
Wine	0.9332	0.9398	0.9424	0.9666
Wine-red	0.5372	0.2671	0.2756	0.8762
Wine-white	0.4548	0.2424	0.2830	0.8689

Comparison of the 3 ensemble methods is performed with the proposed method whose result is shown in Table 6. It contains the evaluation metrics, number of attributes taken and number of bags which is the subsets selected randomly from different combinations of attributes.

Figure 3 visually represents the results of proposed method on different datasets.

The proposed method has higher accuracy in 10 out of 22 datasets for bagging, 15 for boosting and 8 for vote B and B method. Comparing the Tables 4 and 6 it is seen that presented classifier gives better precision in 15 out of 22 datasets than boosting alone. Precision is better in 4 datasets for both bagging and combine vote B and B method. It states that positive predictions made by the proposed classifier are more relevant. The accuracy values also depend upon the random features selected. A feature can contribute more to the classifier prediction than others.

Table 4 Result of Adaboost classifier using decision tree

Dataset	Boosting			
	Accuracy	Recall	Precision	Specificity
Anneal	0.7947	0.3970	0.3663	0.8223
Australian	0.8579	0.8582	0.8593	0.8582
Breast-cancer	0.9400	0.9287	0.9409	0.9287
Car	0.7991	0.4544	0.3857	0.9345
Ecoli	0.6407	0.3831	0.2758	0.9089
Glass	0.7773	0.4820	0.4395	0.9411
Haberman	0.7160	0.5876	0.5643	0.5876
Heart-statlog	0.8148	0.8108	0.8232	0.8108
Heart-h	0.8142	0.8127	0.8208	0.8127
Hepatitis	0.8250	0.7204	0.7215	0.3204
Liver	0.6705	0.5459	0.5850	0.5459
Mobile	0.6225	0.6225	0.7195	0.8742
Nursery	0.7334	0.5395	0.4575	0.9063
Somhappy	0.5433	0.5407	0.5430	0.5407
Teaching assistant	0.4749	0.4739	0.4998	0.7376
Tic-Tac-Toe	0.7111	0.6420	0.7010	0.6420
Travel-insurance	0.9854	0.5000	0.4927	0.5000
Vertebra	0.6742	0.6489	0.6876	0.8438
Wifi-localization	0.8750	0.8750	0.8873	0.9583
Wine	0.8955	0.9015	0.9269	0.9466
Wine-red	0.5304	0.2132	0.1792	0.8684
Wine-white	0.4276	0.1769	0.1431	0.8540

5 Conclusion and Future Scope

Authors compared the ensemble methodologies of bagging and boosting with the proposed methodology. Further, authors also compared the proposed methodology to that of combining bagging and boosting using VOTE B and B method. After the comparison, it reveals that in almost all the cases it got better results than boosting alone. In 8 datasets (Australian, Breast-cancer, Haberman, Heart-statlog, Heart-h, Liver-IND, Somhappy, Travel Insurance) the proposed methodology of combination gave better results, whereas the combination using VOTE B and B gave better results in the remaining 14 datasets. Further, the authors stated that, boosting works better compared to bagging on noiseless dataset, whereas bagging is more efficient compared to boosting on the data containing noise. The methodology the authors proposed has been proved to give better accuracy and achieve a lower error rate in general than bagging and boosting considering Decision Tree as the base classifier.

Table 5 Result of vote bagging and boosting ensemble

DATaset	Vote B and B			
	Accuracy	Recall	Precision	Specificity
Anneal	0.8872	0.7315	0.7844	0.9302
Australian	0.8405	0.8402	0.8415	0.8402
Breast-cancer	0.9429	0.9358	0.9406	0.9358
Car	0.8626	0.7880	0.8455	0.9350
Ecoli	0.8343	0.6657	0.7211	0.9616
Glass	0.9263	0.9163	0.9182	0.9840
Haberman	0.6478	0.5332	0.5272	0.5332
Heart-statlog	0.7889	0.7850	0.7913	0.7850
Heart-h	0.8012	0.7983	0.8104	0.7983
Hepatitis	0.8375	0.7700	0.7657	0.2700
Liver	0.6879	0.6039	0.6299	0.6039
Mobile	0.8685	0.8685	0.8740	0.9562
Nursery	0.7648	0.7585	0.8043	0.9170
Somhappy	0.4705	0.4760	0.4721	0.4760
Teaching assistant	0.6961	0.6950	0.6985	0.8474
Tic-Tac-Toe	0.8698	0.8777	0.8845	0.8777
Travel-insurance	0.9808	0.5062	0.5259	0.5062
Vertebra	0.8355	0.7911	0.8305	0.9206
Wifi-localization	0.9750	0.9750	0.9766	0.9917
Wine	0.9617	0.9663	0.9671	0.9811
Wine-red	0.5516	0.2739	0.2774	0.8807
Wine-white	0.4652	0.2344	0.2922	0.8691

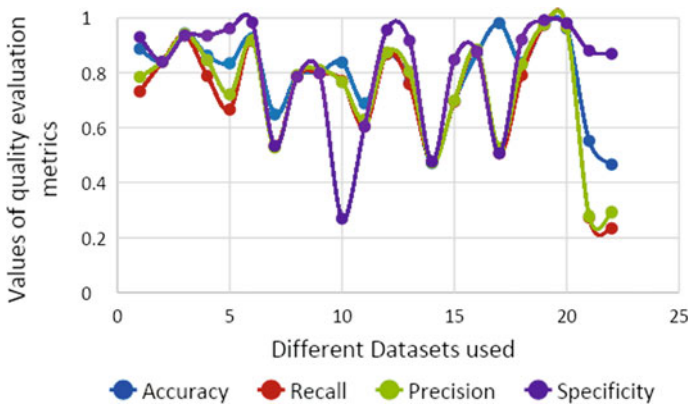


Fig. 2 Visualization of results of vote bagging and boosting ensemble

Table 6 Result of proposed method on datasets

Dataset	Accuracy	No. of bags	No. of attributes	Recall	Precision	Specificity
Anneal	0.7933	50	5	0.3915	0.4070	0.8117
Australian	0.8623	1201	8	0.8581	0.8638	0.8581
Breast-cancer	0.9557	50	5	0.9524	0.9529	0.9524
Car	0.7041	8	3	0.4713	0.4479	0.6768
Ecoli	0.6263	28	4	0.4506	0.4750	0.5275
Glass	0.7206	84	6	0.6063	0.6179	0.3926
Haberman	0.7355	1	1	0.5000	0.3577	0.5000
Heart-statlog	0.8084	686	7	0.4453	0.5333	0.4453
Heart-h	0.8481	686	7	0.8444	0.8529	0.8444
Hepatitis	0.8369	775	15	0.6667	0.5942	0.6667
Liver	0.7133	50	5	0.5499	0.6051	0.5499
Mobile	0.6370	387	16	0.6332	0.7165	0.8786
Nursery	0.7559	28	4	0.6360	0.6268	0.9000
Somhappy	0.6022	8	3	0.5940	0.6226	0.5940
Teaching assistant	0.3896	4	3	0.2867	0.3857	0.7236
Tic-Tac-Toe	0.4355	50	5	0.2651	0.4900	0.2651
Travel-insurance	0.9854	50	5	0.5000	0.4927	0.5000
Vertebra	0.6903	8	3	0.3672	0.4771	0.5689
Wifi-localization	0.9325	14	4	0.4758	0.5081	0.5773
Wine	0.9287	57	10	0.9302	0.9507	0.9629
Wine-red	0.5485	184	6	0.2391	0.1992	0.8512
Wine-white	0.4545	184	6	0.1831	0.1805	0.8381

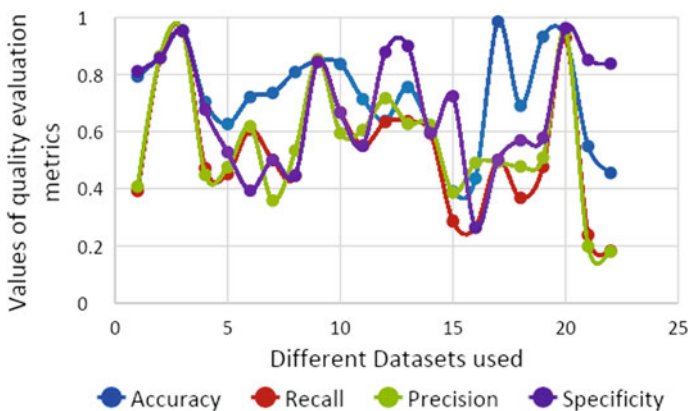


Fig. 3 Visualization of results of proposed method on datasets

Acknowledgements The authors intend to convey gratefulness to David Aha and co-founders, for providing various UCI datasets for our work. Authors would also like to thank Anthony Goldbloom and Ben Hamner for providing Kaggle datasets.

References

1. Quinlan JR (1993) C4. 5: program for machine learning. Morgan Kaufmann, San Francisco
2. Bauer E, Kohavi R (1999) An empirical comparison of voting classification algorithms: bagging, boosting, and variants *Mach Learn* 36(1–2):105–139
3. Machová K, Barcak F, Bednár P (2006) A bagging method using decision trees in the role of base classifiers. *Acta Polytech Hung* 3(2):121–132
4. Vaghela VB, Ganatra A, Amit Thakkar A (2009) Boost a weak learner to a strong learner using ensemble system approach. In: 2009 IEEE international advance computing conference, IEEE. pp 1432–1436
5. Kotsiantis SB, Pintelas PE (2007) Combining bagging and boosting. *Int J Mathe Comput Phys Electr Comput Eng* 1(8):372–381 (2007)
6. Webb GI (2000) Multiboosting: a technique for combining boosting and wagging. *Mach Learn* 40(2):159–196
7. Pavlov DY, Gorodilov A, Brunk CA (2010) BagBoo: a scalable hybrid bagging-the-boosting model. In: Proceedings of the 19th ACM international conference on Information and knowledge management. ACM
8. Breiman L (1996) Bias, variance, and arcing classifiers. Tech. Rep. 460, Statistics. Department, University of California, Berkeley, CA, USA
9. Breiman L (1996) Bagging predictors. *Mach Learn* 24(2):123–140
10. Breiman L (1984) Classification and regression trees, 1st edn. Routledge, New York
11. Freund Y, Schapire RE (1996, July) Experiments with a new boosting algorithm. In: International conference on machine learning. pp 148–156
12. Schapire RE, Freund Y, Bartlett P, Lee WS (1998) Boosting the margin: a new explanation for the effectiveness of voting methods. *Ann Stat* 26:1651–1686
13. Blake C, Merz CJ (1998) UCI repository of machine learning databases [Machine-readable data repository]. University of California, Department of Information and Computer Science, Irvine, CA