Two-Tier Ensemble Classifier

Yash Sukhwani , Harpreet Singh , Prashant Singh Rana
Computer Science and Engineering Department,
Thapar Institute of Engineering & Technology,Patiala, Punjab, India -

Abstract- Ensembling techniques are now a well-defined area in machine learning, leading towards mod- els that are much more accurate and robust, and mostly used in the domains that deal mainly with forecasting tasks. Ensemble models consist of a bunch of standard and generally popular models com- bined in such a way that the resultant model is expected to generate much better results as compared to one single model. In this paper, we propose a two-tier ensemble classification model to combine the predictions made by various models into a unified model for more accurate classification. We then use the proposed model on a variety of datasets to evaluate its performance. The results obtained by evaluation justify the design decisions regarding the learning of the resultant ensemble model. The results also help to conclude that the proposed ensemble yields significantly more accurate predictions as compared to the individual models.

Keywords *Ensemble Model · Two-Tier · Multilevel · Classification · Best-of-N*

1 Introduction

Figure 1 shows the diagrammatic ow of a classification. Initially, data will be loaded into the systems workspace and partition of data will be done between training and testing data. Then, machine learning- based techniques come in action to train a model by considering training data. In the end, the trained model is used to predict the accuracy by using the testing data.

Figure 2 illustrates the basic structure of an ensemble model (regression or classification). The example mentioned uses neural networks as its base classifiers, although as per the definition of ensemble methods, any random classifier method (e.g., random forest,support-vector machine, etc.) can be used in place of the neural networks. Each network in Figure 2's ensemble (network 1 through network N in this case) is trained using the training instances for that neural network. Then, for each instance, the predicted output of each of these networks (o in Figure 2) is combined to produce the output of the ensemble (in Figure 2). A good number of researchers (Alpaydin, 1993; Breiman, 1996c; Krogh & Vedelsby, 1995; Lincoln & Skrzypek, 1989) have independently conducted studies and come to the conclusion that an efficient combination scheme is to simply average the predictions obtained from base classifiers. However, there are many other possible ways in which the outputs from individual networks can be combined to produce a single output.

Ensemble models which combine the predictions of individual classifiers [1], [2] have been ma jorly successful in generating accurate predictions for a lot of complex classification problems [3], [4] . The efficient working of these methods is attributed to their ability to both generate accurate result and correct errors across a number of diverse base classifiers. It has been observed that in most of the cases, diversity is the key to performance of the final ensemble, i.e., if there is complete agreeableness among

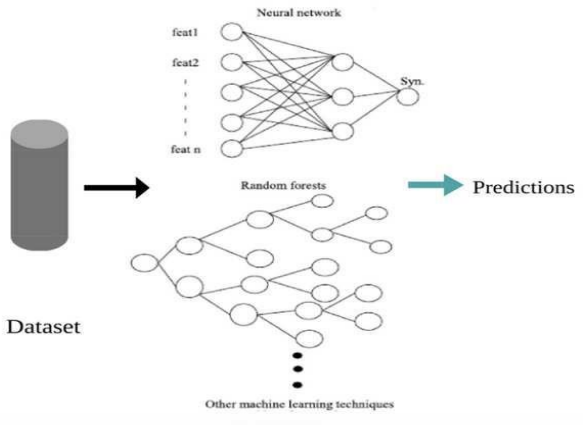

Fig. 1: Flowchart of a Generic Classifier

the base classifiers, the ensemble cannot outperform the best base classifier, however, an ensemble lacking any agreeableness among base classifiers will again have very thin chances of performing well due to its weak base. Successful ensemble methods maintain a balance between the diversity and accuracy of the ensemble.

Some of the more popular ensemble methods like bagging and boosting [5], [6] are able to maintain variety by sampling from or assigning varying importance to training examples but generally use one single type of base classifier to make the ensemble. However, when a choice of base classifier is not clear, the concept of homogeneous ensemblers may not be a good choice. One can instead opt for an ensemble from the results generated from a wide variety of heterogeneous base classifiers such as support vector machines, neural

networks, and decision trees. Two of the most popular heterogeneous approaches to ensembling include a form of meta-learning called stacking [7] [8] as well as ensemble selection [9] [10]. Stacking involves formation of an upper-level predictive model over the predictions of base classifiers, while the technique of ensemble selection deals in incremental strategy to select base predictors for the ensemble while maintaining a balance between diversity and accuracy. Due to their ability to utilize heterogeneous base classifiers, these approaches have superior performance across several application domains [4] [11]. The knowledge obtained from the analysis of the proposed ensemble model for the classification problems should have wide applicability across various applications of ensemble learning.

We start by describing the datasets used and the ensemble methods studied (namely ensemble selec- tion and stacking) in Section 2.3. This is followed by a discussion of the proposed ensemble classifier in Section 3.2. Evaluation metrics used in this study are discussed in section 4 while section 5 deals with analysis of comparison between standard and proposed models on the basis of evaluation metrics.

2 Materials and Methods

2.1 Datasets

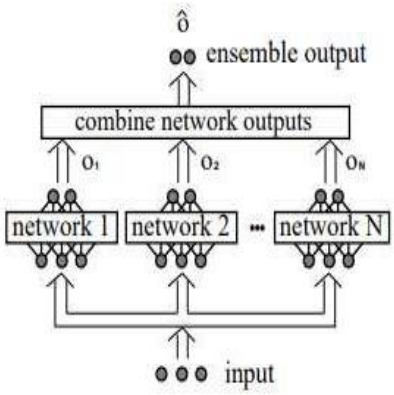

Fig. 2: A classifier ensemble of neural networks

Table 1: Datasets used in Study

Dataset Used	Instances	Features	Classes
YouTube Spam Collection Data Set	1956	5	2
Banknote Authentication Data Set	1372	5	2
Occupancy Detection Data Set	20560	7	2
HTRU Data Set	17898	9	2
Sports Articles for Objectivity Analysis	1000	59	2

To evaluate the accuracy of the proposed ensemble classifier, we collected a number of data sets from UCI data set repository (Murphy & Aha, 1994). These data sets were selected after filtering them through certain parameters such that they (a) were related to the problems of real scenarios, (b) had different dimensions and other characteristics, and (c) have been categorised as important by previous researches. Table 1 provides the details of these data sets.

2.2 Machine Learning Methods

The models used as basic classifiers are : Random Forest Classifier, Decision Tree Classifier and Support Vector Classifier . Random forest : Random forests or random decision forests are an ensemble learning method for prediction modelling tasks, that operate by generation of a number of decision trees at training time and then selecting the class cited as prediction by the maximum number of trees (classification) , i.e., the predictions obtained from the individual decision tress or mean prediction (regression) of the individual trees. Random decision forests are a good replacement for decision trees which generally result in overfitting. [12] [13]

Decision Tree : A decision tree is a decision support tool which makes use of a tree-like graph or model of decisions and their possible results, including chance event outcomes, costs of the resources used, and functionality. It is one of the ways to show an algorithm that makes use of just conditional control statements. [14]

SVM : Support vector machines (SVMs also knows as support vector networks) are supervised ma- chine learning analysis models that are used to analyse data for predictive modelling tasks. For example, a training set (every instance has an associated label) when input to the SVM algorithm, generates a model that makes predictions on new instances input to the model as part of test data. On analysing the approach, we can conclude that SVM is a non-probabilistic classifier. [15]

2.3 Ensemble Methods

1. Simple Aggregation : The predictions given by each base classifier is appended in a new column in the data set itself at a location i, j where i refers to an instance while j refers to the new column corresponding to the result of that specific classifier. Once all the base classifiers are done with their initial run, all the results are averaged and the result is this resultant mean value.

2. Meta-Learning : Meta learning is a general methodology that says that better prediction algorithms can be generated by using meta information provided by other base classifiers. Stacking is an example of meta-learning [7]. Stacked generalization (or stacking) involves training a higher-level classifier model on the predictions generated by a low-level classifier model Using the standard formulation given by Ting and Witten [16], we perform meta-learning on logistic regression classifier at higher level which is trained on the predictions generated by a number of heterogenous low-level classifiers. Even though there are many possible models available to use instead of logistic regression classifier, the choice is justified as it helps to avoid overfitting. Also, weights assigned by this higher-level classifier to underlying low-level classifiers give insights into their performances.

3. Cluster-Based Meta-Learning : Another variation of old-school stacked generalisation is to combine classifiers at base level having similar outputs (predictions) and then to apply different higher-level

classifiers for each separate cluster. Alternatively, classifiers can be first combined to by generating average values and then training a higher-level classifier on these group os clusters averaged predic- tions. For simplicity, the former approach is termed as intra-cluster stacking while the latter is referred to as inter-cluster stacking. The methodology behind both approaches though involves formation of clusters and to resolve the differences encountered between those predictions. Hence, the performance of a classifier is affected by diversity of predictions generated by various classifiers within a cluster.

4. Ensemble Selection : Ensemble selection begins with choosing random subsets of classifiers that when put to work in combination generate an efficient ensemble as including every single classifier would just result in decrement of the performance of the overall ensemble. However, going through every possible combination for every new data set is infeasible from a practical point of view and hence heuristics are put to use to make justified assumptions for the optimal subsets. Another point to observe is that the performance of the ensemble can only increase as compared to the best of its base classifier and hence the precondition is that the base classifiers being used should also have a good amount of accurate predictions and so good selection methods satisfy this precondition.

3 Methodology

3.1 Flow of the proposed Model

Figure 3 shows the flowchart of the proposed model.

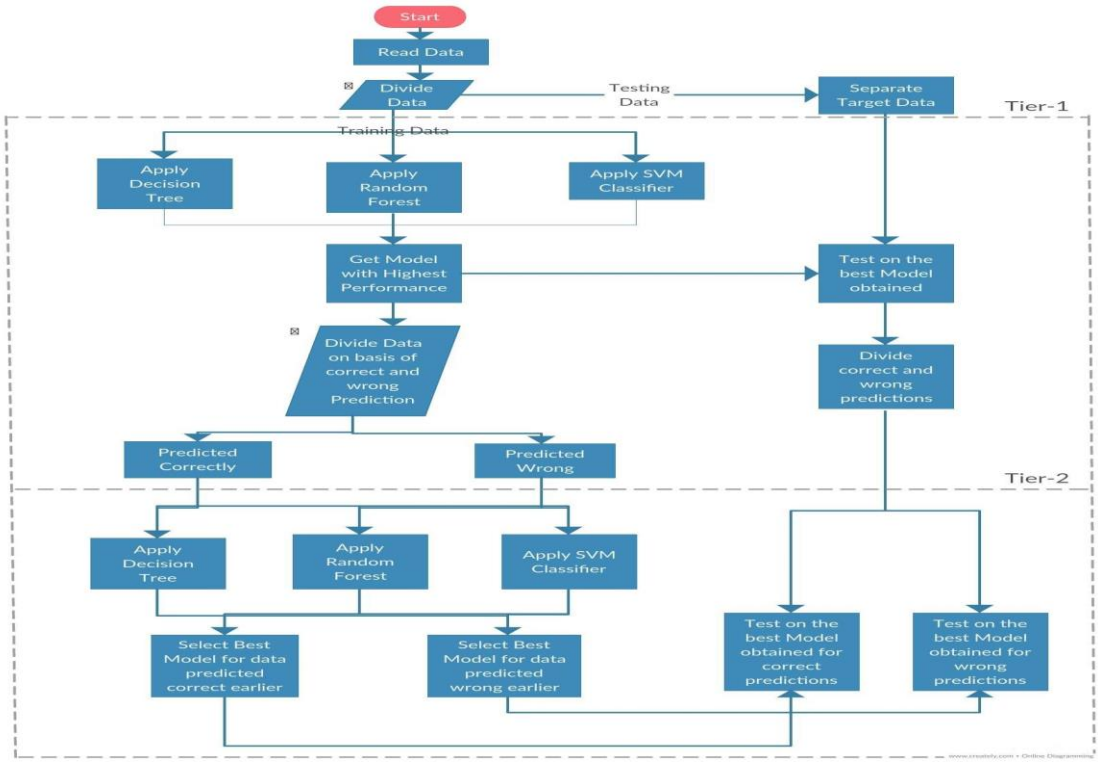

Fig. 3: Flow of the proposed Model

3.2 Proposed Two-Tier Ensemble Model

First of all, a few standard models are selected (namely, Random Forest, Decision Tree and Support Vector Machine) for base level (or Level-1). The standard models are trained on the training data and tested to select a model as base classifier. The model performing the best is chosen as the Level-1 classifier. The examples predicted correctly and examples not predicted correctly by the base classifier are then separated and stored as 2 different datasets. Now, we again run the standard models to test their performance on the 2 new datasets generated by division of examples and choose the best performing model for each of the datasets. These newly selected models may or may not be similar to the base classifier. These models function as level-2 classifiers and hence the examples having more chance of being predicted wrong by the base classifier are again tested by the Level-2 classifiers depending upon their prediction by the base classifier. The advantage of the process is that the examples having a chance of being predicted wrong are instead predicted using a different model which increases the accuracy of the overall ensemble model as the probability of predicting examples wrong is reduced.

4 Model Evaluation

There are a number of parameters available for evaluating the performance of a model such as gini index, accuracy, area under curve (ROC curve), specicity and sensitivity, recall, precision and many more [17].

4.1 Performance evaluation

The current study makes use of parameters such as AUC score, sensitivity, specificity and lastly, the accuracy for evaluating and hence comparing the performance of all models.

4.1.1 AUC

Area under the curve (AUC) gives a measure of the how efficient a classifier is. The area covered under the receiver operating characteristics (ROC) curve is termed as AUC. The model scoring a higher AUC as compared to other models is considered as more efcient. Its value lies in the range of [0, 1]. The quality of model is good if it has AUC value near to 1.

Table 2: Occupancy Detection Data Set

Dataset	Model	Accuracy	AUC	Sensitivity	Specificity
Occupancy Detection	Random Forest	80.11	0.53	1.00	0.06
	Decision Tree	79.29	0.50	1.00	0.00
	SVM	95.28	0.98	0.98	0.98
	Proposed Model	98.08	0.96	1.00	0.92
Youtube Spam Collection	Random Forest	59.46	0.60	0.51	0.69
	Decision Tree	65.77	0.67	0.37	0.96
	SVM	48.65	0.50	0.00	1.00
	Proposed Model	73.87	0.74	0.53	0.96
Sports Articles for Objectivity Analysis	Random Forest	70.00	0.70	0.70	0.71
	Decision Tree	70.00	0.61	0.97	0.24
	SVM	65.67	0.55	0.98	0.12
	Proposed Model	74.33	0.66	1.00	0.31
HTRU2 Data Set	Random Forest	91.64	0.89	0.92	0.85
	Decision Tree	90.97	0.50	1.00	0.00
	SVM	90.89	0.50	1.00	0.00
	Proposed Model	98.64	0.92	1.00	0.85
Banknote Authentication	Random Forest	57.77	0.60	0.29	0.91
	Decision Tree	46.85	0.50	0.00	1.00
	SVM	46.85	0.50	0.00	1.00
	Proposed Model	62.14	0.64	0.29	1.00

4.1.2 Accuracy

Accuracy is calculated as the number of examples classified correctly by the model. The accuracy can be calculated by:

$$Accuracy = 100 * (TP + TN)/TotalData \quad (1)$$

4.1.3 Sensitivity

Sensitivity, also known as true positive rate (TPR), is given by number of actual positives (instances that are actually positive) divided by the number of instances classified by the model as positive. It is computed as:

$$Sensitivity = TP/(TP + FN) \quad (2)$$

4.1.4 Specificity

Specicity is also known as true negative Rate (TNR). It relates to the classiers ability to identify negative results and is computed as:

$$Specificity = TN/(TN + FP) \quad (3)$$

5 Result analysis, comparison and Discussion

The proposed model is tested on 5 different datasets and the results are tabulated in Tables 2.

On observing the experimental results of testing the proposed model on different types of datasets, we see that there is no one single standard model that always performs better than other models under study. The proposed model, however, is able to get more accurate results as compared to other models as instead of following a completely different concept for predicting class of examples, it builds up on the models that outperform others on a specific part of the datasets and hence on combining the results from individual models, the overall accuracy of the ensemble surpasses that of individual models used. The reason for increase in the accuracy of ensemble is because it predicts class of different parts of data set using different models (one which performs best on that specific part of data set). Another interesting

point to observe from results is that as the size of the data set increases, the model gives much more accurate results. As the factor of size of data set increases, the ensemble model gives even better results.

6 Conclusion

The objective of ensemble techniques is to combine different variety of classifiers in an efficient way so as to improve the overall performance of the final ensemble model than any of the underlying base classifiers. Since the task of going through all possible combinations of standard classifiers quickly results in infeasible ensembles for even the smallest of ensemblers, other techniques for generating efficient ensembles has been widely researched in the last few years.

In this paper, we tried to generate a new ensembler based on ensemble techniques such as stacking and ensemble selection and analysed the results. We find that both stacking and ensemble selection approaches show great improvements in performance as compared to standard models like SVM and decision tree and even moderate improvements over random forest classifiers which are deemed as effective in most of the domains . Here, even small improvements in accuracy can contribute directly to big improvements in a lot of fields such as medical analysis and image enhancement techniques .

References

1. Lior Rokach. Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1-2):1–39, 2010.
2. Giovanni Seni and John F Elder. Ensemble methods in data mining: improving accuracy through combining predictions.*Synthesis Lectures on Data Mining and Knowledge Discovery*, 2(1):1–126, 2010.
3. Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1297–1304. Ieee, 2011.
4. André Altmann, Michal Rosen-Zvi, Mattia Prosperi, Ehud Aharoni, Hani Neuvirth, Eugen Schülter, Joachim Büch, Daniel Struck, Yardena Peres, Francesca Incardona, et al. Comparison of classifier fusion methods for predicting response to anti hiv-1 therapy. *PloS one*, 3(10):e3470, 2008.
5. Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.
6. Robert E Schapire and Yoav Freund. *Boosting: Foundations and algorithms*. MIT press, 2012.
7. David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.
8. Christopher J Merz. Using correspondence analysis to combine classifiers. *Machine Learning*, 36(1-2):33–58, 1999.
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. Rich Caruana, Art Munson, and Alexandru Niculescu-Mizil. Getting the most out of ensemble selection. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 828–833. IEEE, 2006.
11. Alexandru Niculescu-Mizil, Claudia Perlich, Grzegorz Swirszcz, Vikas Sindhwani, Yan Liu, Prem Melville, Dong Wang, Jing Xiao, Jianying Hu, Moninder Singh, et al. Winning the kdd cup orange challenge with ensemble selection. In *Proceedings of the 2009 International Conference on KDD-Cup 2009-Volume 7*, pages 23–34. JMLR. org, 2009.
12. Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.
13. Carolin Strobl, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, 8(1):25, 2007.
14. S Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, 21(3):660–674, 1991.
15. Thorsten Joachims. Making large-scale svm learning practical. Technical report, Technical report, SFB 475: Kom- plexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, 1998.
16. Kai Ming Ting and Ian H Witten. Issues in stacked generalization. *Journal of artificial intelligence research*, 10:271–289, 1999.
17. Divya Khanna and Prashant Singh Rana. Multilevel ensemble model for prediction of iga and igg antibodies. *Im- munology letters*, 184:51–60, 2017.