A Weighted Majority Voting Ensemble Approach for Classification

Alican Dogan
The Graduate School of Natural and Applied Sciences
Dokuz Eylul University
Izmir, Turkey
alican.dogan@deu.edu.tr

Derya Birant
Department of Computer Engineering
Dokuz Eylul University
Izmir, Turkey
derya@cs.deu.edu.tr

***Abstract*—Ensemble learning combines a series of base classifiers and the final result is assigned to the corresponding class by using a majority voting mechanism. However, the base classifiers in the ensemble cannot perform equally well. Therefore, the base classifiers should be assigned different weights in order to increase classification performance. In this study, a novel Weighted Majority Voting Ensemble (WMVE) approach is proposed, which evaluates individual performances of each classifier in the ensemble and adjusts their contributions to class decision. In the proposed weight adjustment model, only reward mechanism is provided, so punishment is not included. Classifiers that correctly classify observations which are not correctly classified by most of the classifiers gain more weights in the ensemble. In the experimental studies, increasing value of weight was calculated for each classifier in a heterogeneous collection of classification algorithms, including C4.5 decision tree, support vector machine, k-nearest neighbor, k-star, and naive Bayes. The proposed method (WMVE) was compared with the Simple Majority Voting Ensemble (SMVE) approach in terms of classification accuracy on 28 benchmark datasets. The effectiveness of the proposed method is confirmed by the experimental results.**

Keywords—classification*, *ensemble learning*, *weighted majority voting*, *machine learning

I. Introduction

Ensemble learning (EL), which is a type of machine learning, is a combination of multiple solution approaches rather than usage of an individual model with the aim of solving a specific problem. The logic behind this strategy is focused on creating a collection of distinct solutions in order to obtain more reasonably predicted results than a single technique can acquire. The source of more effective performance of the ensemble approach is the diversity of methods and management of the variations.

Ensemble learners can be analyzed into four categories which are Bagging, Boosting, Stacking and Voting. First of all, *bagging* (or bootstrap aggregation) utilizes bootstrap sampling for generating data subsets to feed base classifiers. Secondly, *boosting* is defined as a set of approaches, which is capable of transforming weak learning models to strong learning models by weighting data instances so that data instances on which previous model were incorrect are more important. The most commonly used boosting algorithm is AdaBoost which is an abbreviation for Adaptive Boosting. The next ensemble category is *stacking*, in which the whole training data is used to feed base classifiers and the single learning model is trained with the outputs they generate. Since base classifiers are different from each other, stacking models are generally heterogeneous. Finally, *voting* benefits from diverse machine learning solutions to determine the correct label of a data and makes a prediction as the decision receiving most votes.

This study focuses on voting procedure of ensemble members (multiple classifiers) to make final prediction for a given ensemble. In the simplest voting form, called majority voting, each classifier is treated equally and contributes a single vote. The final prediction is given by the majority of the votes, i.e., the class that obtains the highest number of votes (the most frequent vote) is the final prediction. On the other hand, in weighted voting, the classifiers have varying degrees of effect on the final prediction. Each classifier is associated with a coefficient (weight), usually proportional to its classification performance on a validation set. The final decision is made by summing up all weighted votes and by selecting the class with the highest aggregate. In this study, a novel weighted majority voting approach is proposed for ensemble learning.

The main contributions of this paper can be summarized as follows: (i) firstly, it provides a brief survey of weighted majority voting methods, which have been introduced to improve prediction performance of the standard ensemble learning approaches; (ii) it proposes a novel Weighted Majority Voting Ensemble (WMVE) that considers increasing the effect of models making correct predictions according to the failure rate of other models; iii) it presents a number of experimental studies that compared on twenty-eight benchmark datasets to demonstrate that the proposed WMVE method generally produce better classification results than both simple majority voting ensemble (SMVE) approach and individual standard classification algorithms in terms of accuracy. The classification algorithms used in the experimental analysis are C4.5 decision tree, support vector machine (SVM), k-nearest neighbor (KNN), k-star, and naive Bayes (NB). In this research, we preferred these algorithms because of their popularities.

The remainder of this paper is organized as follows. In Section 1, related and previous works on the subject is summarized. In Section 3, the proposed Weighted Majority Voting Ensemble (WMVE) approach is explained in detail. Section 4 gives the characteristics of benchmark datasets used in this study. It also presents the performance of the proposed method on the datasets. Finally, in the last section, some concluding remarks and possible future works are given.

II. Related Work

Weighted majority voting has been utilized in various areas such as health [1], environment [2], intrusion detection [3], facial expression recognition [4], text mining [5] and

978-1-7281-3964-7/19/$31.00 ©2019 IEEE

software engineering [6] [7]. It has helped to classify EEG signals for the task of Epileptic Seizure Detection [1], to detect pollutants in surroundings automatically [2], to enhance predictive capability in sentiment analysis [5] and to estimate possible bugs in a software project [6].

Many different approaches have been proposed to adjust the weights of base classifiers in ensembles so far, such as based on fuzzy sets [8], particle swarm optimization (PSO) [9], and genetic algorithm (GA) [10]. For instance, reference [8] used interval valued fuzzy set to represent weights. Reference [9] demonstrated a rigid performance of PSO method on some datasets to optimize the weight of each learner. Similarly, differential evolution was used for weight optimization to have strong generalization ability [11]. There also exists a study that considers probabilistic framework for weight determination in an ensemble system [12]. Apart from the analytical optimization approaches to optimize weights, meta-heuristic way of thinking was also preferred. Optimum weighted model of several classifiers is determined by GA in another study [10].

Until now, many different classification algorithms have been used as base classifiers in weighted ensemble studies. Support vector machines (SVM), neural networks (NN) and decision trees are some examples of algorithms used in weight adjustment approaches found in the literature. To exemplify, SVM was implemented in an ensemble voting scheme because of its promising results on remote sensing data [13]. The NN algorithm was also used to form an ensemble model with the aim of detecting failure cause [14] [2]. The other study of weighted voting based ensemble learning benefits from decision trees in order to derive a consensus decision [15].

Application area of weighted majority voting is not limited with standard classification tasks. It is also applied to solve multi-label learning problems in which each instance may belong to more than one class. Several methods were proposed to cope with multi-label challenges such as high dimensionality and imbalanced data [16]. They proposed a dynamic weighting method using Shepherd rule to determine the weights of classifiers. In some review papers [17], the advantages of weighted majority voting approach were clearly emphasized.

Our approach differs from the previous works in various aspects, such as strategy and methodology. First, the weights in the existing methods in the literature generally vary according to reward and punishment strategies. In our model, only reward mechanism is provided, so punishment is not included. Second, in the existing methods, generally the accuracy of classifiers on the dataset is taken into consideration. However, we propose a different methodology in which the weight of a classifier that clasify an instance correctly is increased by the percentage of ensemble members that incorrectly classify an instance in the validation set.

III. Materials and Method

A. Simple Majority Voting Ensemble

Ensemble learning involves the combination of predictions made by different classifiers. The simplest yet most popular combination method is majority voting. Although majority voting is applied in various styles, the most common one considers the highest number of votes, i.e., assigning an instance to the class that most base classifiers agree on. In this type of voting, all classifiers have the same value of votes and they are all equal to 1. Let n be the number of classifiers in the ensemble. Assume that C_t is used to represent a classifier in the ensemble E such that $t = 1,2,...,n$ and ensemble $E = \{C_1,C_2,...,C_n\}$. The decision of the t^{th} classifier (C_t) is denoted by $d_{t,j} \in \{0,1\}$, where $j = 1,2,...,k$ and k is the number of classes. The decision will produce $d_{t,j} = 1$, if t^{th} classifier decides for class c_j, and $d_{t,j} = 0$ otherwise. The output of the ensemble, in majority voting, can be outlined with the following equation (1).

$$\max_{1\le j\le k} \sum_{t=1}^{n} d_{t,j} \tag{1}$$

However, the base classifiers in an ensemble usually cannot perform equally well and therefore, considering them equally in aggregation may not be optimal. In this case, the appropriate solution is to weight each classifier based on its performance. The most important key issue in weighting schemes is how to appropriately determine the weights of classifiers, which can strongly influence the performance of the ensemble. In this study, a novel weighting mechanism is proposed.

B. Weighted Majority Voting Ensemble

In the proposed approach, called Weighted Majority Voting Ensemble (WMVE), weights are determined according to the classification performances of classifiers. Some classifiers could have higher votes if they had revealed more reliable work in previous estimates. If it is proven that any machine learning algorithm can make more confident inferences for a specific dataset than the others, it is advisable to increase their value of votes in order to obtain more successful results compared to classical majority voting.

Fig. 1 shows the proposed Weighted Majority Voting Ensemble (WMVE) process. The algorithm associates each learned classifier with a different weight according to its classification performance in the validation set. The final prediction for each instance is done based on highest weighted votes.

In the proposed approach (WMVE), there are three phases. The first phase is to train classifiers on training set. The second phase is to determine the weights of the classifiers using validation set. In this phase, each classifier generates a decision pointing to predicted class label of a single instance and then these decisions are evaluated to update weights. The third phase is to combine the outputs of individual classifiers by considering their weights.

So, the whole dataset is split into three subsets which are training, validation and test sets. The cardinality of the validation set is the same as the size of test set and is equal to 1/11 of dataset. Let m be the number of instances in the validation set. Table I visualizes correct and incorrect predictions of classifiers for each instance in the validation set, where

$$p_{ij} = \begin{cases} 1, & \text{if } j^{th} \text{ classifier makes a correct prediction for } i^{th} \text{ instance} \\ 0, & \text{if } j^{th} \text{ classifier makes an incorrect prediction for } i^{th} \text{ instance} \end{cases}$$

TABLE I. THE PREDICTIONS OF CLASSIFIERS FOR EACH INSTANCE IN THE VALIDATION SET

Instances \ Classifiers	C_1	C_2	...	C_n
1	p_{11}	p_{12}	...	p_{1n}
2	p_{21}	p_{22}	...	p_{2n}
.	.	.	.	.
m	p_{m1}	p_{m2}	...	p_{mn}

In the proposed approach, the weights are updated for each instance in the validation set. Initially, all weights are equal to 1. All instances in the validation dataset are traversed and processed through n classifiers once. The weights of the classifiers that correctly predict class label of an instance are incremented by the ratio of the number of incorrectly predicting classifiers to the whole number of classifiers (n). This weight gain value cannot exceed 1 for a single instance operation. The weights calculated for each instance in the validation set are represented in Table II.

In Table II, w_{ij} is the weight of j^{th} classifier as a result of the operation realized on i^{th} instance, where $i = 1,2,...,m$ and $j = 1,2,...,n$. After the classifiers in the ensemble are trained in parallel using training set, weight w_{ij} is recalculated step by step for each instance in the validation set as follows:

$$w_{ij} = \begin{cases} w_{i-1,j} + \alpha_i & if\ j^{th}\ classifier\ makes\ a\ correct \\ & prediction\ for\ i^{th}\ instance \\ w_{i-1,j} & if\ j^{th}\ classifier\ makes\ an\ incorrect \\ & prediction\ for\ i^{th}\ instance \end{cases}$$

where α_i is the change in weight and calculated as $\alpha_i = Y_i / n$, where Y_i is the number of incorrect predictions for i^{th} instance and n is the number of classifiers.

TABLE II. THE WEIGHTS CALCULATES FOR EACH INSTANCE IN THE VALIDATION SET

Instances \ Classifiers	C_1	C_2	...	C_n
0	1.0	1.0	...	1.0
1	w_{11}	w_{12}	...	w_{1n}
2	w_{21}	w_{22}	...	w_{2n}
.	.	.	.	.
m	w_{m1}	w_{m2}	...	w_{mn}

When all records in the validation set are processed through n classifiers once, the final values are stored as weights. Those weights are used as the effect of vote for each classifier to predict class labels of the instances in the test set. In the final decision, all weighted votes are summed for each class and the class receiving the most weighted votes becomes predicted class of an instance to be classified, as given in (2).

$$\max_{1 \le j \le k} \sum_{t=1}^{n} w_t\, d_{t,j} \tag{2}$$

The proposed method is a generic classification approach; therefore it is independent from the field. So it can be applied to different domains easily.

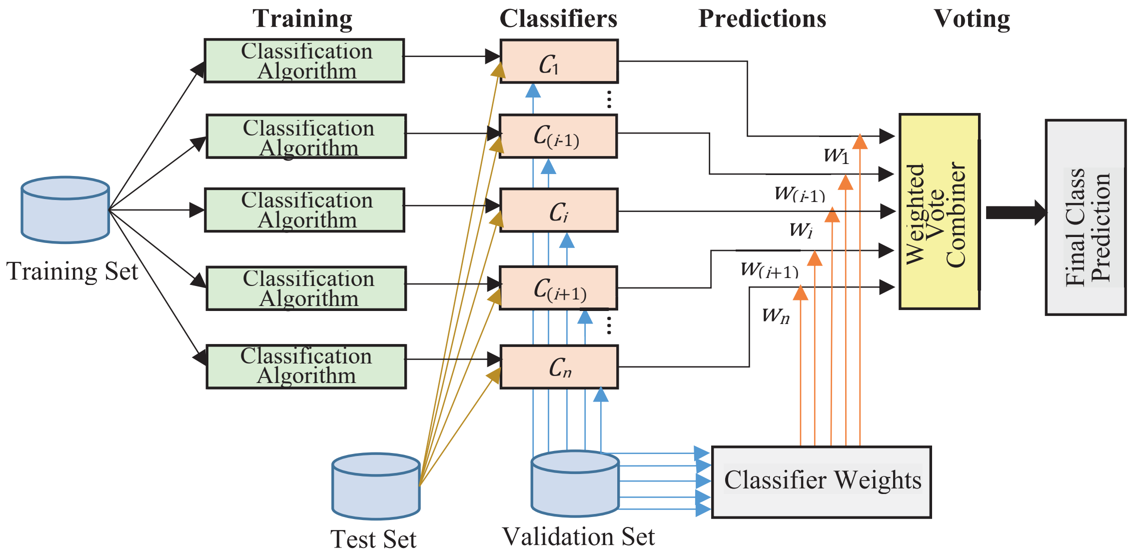

Fig. 1. Weighted majority voting ensemble (WMVE)

C. An Example for WMVE

In this section, the proposed approach (WMVE) is illustrated with an example. A sample data is given in Table III.

In the example senario (Table III), there are three classifiers (C_1, C_2, and C_3) in the ensemble and their predictions on five instances in the validation set are listed. The final column in the table shows the actual class labels of the instances. There two possible class labels: *X* and *Y*.

TABLE III. AN EXAMPLE DATASET

Instances \ Classes	Predicted Classes C_1 Prediction	C_2 Prediction	C_3 Prediction	Actual Class
1	X	X	Y	Y
2	X	Y	Y	Y
3	X	X	X	X
4	X	Y	X	X
5	X	Y	X	Y

Table IV shows the changes in the weights of the classifiers for each instance. To begin with, all weights are initialized to 1. In each iteration through instances in the validation set, the weights of correctly predicting classifiers are only incremented, so the weights of the others are not modified. The weights are incremented by the ratio of the number of incorrectly predicting classifiers to the whole number of classifiers (n = 3). For example; there are two incorrect matches for the first instance, so the weight of the 3rd classifier is updated by 2/3. This incremental process finishes with the last item.

TABLE IV. CHANGES IN THE WEIGHTS FOR EACH INSTANCE

Instances \ Classifiers	C_1	C_2	C_3
Initial Weights	1.0	1.0	1.0
1	1.0 + 0 = 1.0	1.0 + 0 = 1.0	1.0 + 2/3 = 1.67
2	1.0 + 0 = 1.0	1.0 + 1/3 = 1.33	1.67 + 1/3 = 2.0
3	1.0 + 0/3 = 1.0	1.33 + 0/3 = 1.33	2.0 + 0/3 = 2.0
4	1.0 + 1/3 = 1.33	1.33 + 0 = 1.33	2.0 + 1/3 = 2.33
5	1.33 + 0 = 1.33	1.33 + 2/3 = 2	2.33 + 0 = 2.33
Final Weights	1.33	2	2.33

IV. EXPERIMENTAL STUDIES

The proposed method (WMVE) was implemented by using Weka open source data mining library [18] on Visual Studio with C# programming language. The ensemble was constructed by 5 classifiers and each classifier was trained by a different classification algorithm, including C4.5 decision tree, support vector machine, k-nearest neighbor, k-star and naive Bayes. The default parameters of the algorithms in Weka were used in all experiments; expect k parameter of the KNN algorithm which was set to 5.

The proposed method (WMVE) was compared with the existing SMVE approach and with individual standard classification algorithms in terms of classification accuracy. The methods were applied on 28 benchmark datasets from various fields such as medicine, automotive, financial and so on. First, validation group (1/11 of the whole data) was randomly selected from the dataset with the aim of being used in weighting process. Next, 10-fold cross validation technique is used to divide the remainder data into training and test parts.

A. Dataset Description

In the experimental studies, a set of datasets were used to demonstrate the ability of the proposed method. Accurately, 28 different benchmark datasets that are well-known and broadly used in machine learning were utilized to compare the methods. Main characteristics of the datasets utilized in this study are given in Table V, including the number of attributes, instances, classes and domains.

These datasets are publicaly available and they were obtained from UCI Machine Learning Repository [19]. Some datasets are appropriate for binary classification, while some of them are for multi-class classification. They are typically balanced datasets. They are from different domains such as health, financial, education, transportation, game, biology, environment, food and zoology. So, this diversity demonstrates the independence of the proposed approach from the domain.

TABLE V. THE BASIC CHARACTERISTICS OF THE DATASETS

ID	Dataset Name	Attributes	Instances	Classes	Domain
1	arrhythmia	279	452	16	medicine
2	audiology	69	226	24	audiology
3	breast-cancer	9	286	2	medicine
4	breast-w	9	699	2	medicine
5	car evaluation	6	1728	4	automotive
6	credit-a	15	690	2	financial
7	dermatology	34	366	6	medicine
8	ecoli	7	336	4	biology
9	glass	10	214	7	chemical
10	hypothyroid	21	7200	3	medicine
11	ionosphere	34	351	2	radar
12	iris	5	150	3	environment
13	kr-vs-kp	36	3196	2	game
14	labor	16	57	2	social
15	letter	16	20000	26	alphabet
16	liver disorders	6	345	2	medicine
17	lymphography	20	148	4	medicine
18	nursery	8	12960	5	education
19	page-blocks	10	5473	5	document
20	segment	21	2310	7	image
21	sick	30	3772	2	medicine
22	sonar	208	60	2	physical
23	soybean	35	307	19	biology
24	spambase	57	4601	2	e-mail
25	tic-tac-toe	9	958	2	game
26	vehicle	18	946	4	image
27	wine	15	178	3	food
28	zoo	18	101	7	zoology

B. Experimental Results

In this study, classification accuracy was used as evaluation metric to compare the performances of the methods. Accuracy is a performance measure that shows the percentage of correctly predicted observations and calculated as *(TP + TN) / (TP + TN + FP + FN),* where *TP, TN, FP* and *FN* represent the number of true positives, true negatives, false positives and false negatives, respectively.

Table VI shows the comparative results for the proposed method (WMVE) with SMVE and five single classifiers, including C4.5 decision tree, support vector machine (SVM), k-nearest neighbor (KNN), k-star and naive Bayes (NB). The best accuracy values in each row are highlighted in bold. The obtained results show that the proposed method (WMVE) generally provides higher accuracy values than others. It outperformed the others in 20 out of 28 datasets. According to the average accuracy results, the proposed WMVE approach has the best accuracy score with 90.17%.

The graph given in Fig. 2 shows the average ranks of the algorithms. In the ranking procedure, alternative methods are rated by their performances on the datasets. Firstly, the rank 1 is given to the method that obtained the best result, rank 2 to the second best, and so on with the rank m being assigned to the worst one. If two or more methods have the same performance, average rank value is assigned to each method. Lastly, the average ranks per method are calculated as final rank values. According to the average rank values given in Fig. 2, the proposed WMVE approach is superior to others since it has the lowest rank value (1.57).

TABLE VI. COMPARISON OF CLASSIFICATION ACCURACIES (%)

Dataset Name	C4.5	SVM	KNN	K-Star	NB	SMVE	WMVE
arrhythmia	65.61	69.51	58.78	56.10	61.71	66.59	**70.24**
audiology	77.56	80.98	60.00	80.98	72.20	79.95	**81.4**
breast-cancer	72.31	70.00	75.00	72.31	70.38	74.23	**75.38**
breast-w	94.96	97.01	96.69	95.75	96.22	**97.64**	**97.64**
car evaluation	90.70	92.93	92.23	86.88	85.03	90.70	**93.38**
credit-a	86.28	86.12	85.65	79.27	77.51	**86.45**	**86.45**
dermatology	93.67	96.08	97.29	94.58	97.59	**98.18**	**98.18**
ecoli	85.25	83.61	85.9	81.64	84.59	87.17	**88.16**
glass	69.59	52.58	65.98	**75.26**	48.97	69.11	71.61
hypothyroid	**99.48**	93.67	93.26	94.66	95.42	94.81	**99.48**
ionosphere	91.22	88.71	84.95	83.70	82.13	**91.25**	**91.25**
iris	94.12	**95.59**	**95.59**	94.12	**95.59**	94.78	94.78
kr-vs-kp	**99.38**	95.97	95.73	96.49	87.37	98.04	98.11
labor	88.24	92.16	86.27	92.16	92.16	**94.33**	**94.33**
letter	87.57	82.15	95.09	95.70	64.08	94.42	**96.25**
liver-disorders	63.9	58.15	61.34	65.18	54.63	69.95	**70.59**
lymphography	80.6	88.06	83.58	88.06	82.09	88.02	**90.22**
nursery	96.7	93.03	**98.05**	96.65	90.20	97.12	97.58
page-blocks	96.88	93.11	95.96	96.8	90.91	96.60	**97.19**
segment	96.62	91.90	94.71	96.95	80.05	97.19	**97.52**
sick	**98.75**	93.93	96.03	95.65	92.53	96.97	97.46
sonar	69.31	77.25	83.60	**86.24**	67.72	84.62	84.06
soybean	90.48	92.58	90.32	88.39	**93.06**	92.90	92.90
spambase	92.73	90.39	90.05	90.55	78.86	94.12	**94.19**
tic-tac-toe	84.71	98.62	**98.97**	95.75	70.34	97.70	98.74
vehicle	72.30	73.86	68.53	69.44	44.21	74.26	**74.39**
wine	93.17	98.14	96.89	98.14	98.14	**98.79**	**98.79**
zoo	92.31	94.51	90.11	95.60	94.51	**94.56**	**94.56**
Average	**86.59**	**86.45**	**86.31**	**87.25**	**80.29**	**89.30**	**90.17**

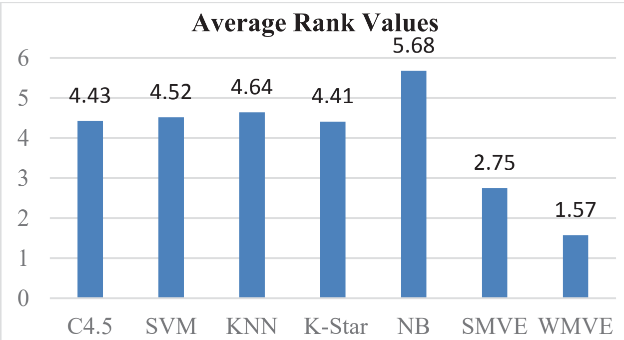

Fig. 2. The average ranks of the methods

The matrix presented in Table VII gives all pairwise comparisons of the methods. Each cell in the matrix presents the number of wins, ties and losses between two methods located in that row and that column. For instance, in the pairwise of WMVE and C4.5 algorithms, 25-1-2 indicates that the proposed WMVE approach is better than C4.5 algorithm on 25 datasets, it is worse on 2 datasets, and they are equal in only one dataset. When the results given in the matrix is considere, it is possible to say that our proposed approach WMVE outperforms all other methods.

TABLE VII. THE PAIRWISE COMPARISONS OF THE METHODS

	C4.5	SVM	KNN	K-Star	NB	SMVE	WMVE
C4.5	0-28-0	14-0-14	16-0-12	13-2-13	20-0-8	5-1-22	2-1-25
SVM	14-0-14	0-28-0	15-1-12	11-4-13	19-4-5	6-0-22	1-0-27
KNN	12-0-16	12-1-15	0-28-0	13-0-15	19-1-8	6-0-22	3-0-25
K-Star	13-2-13	13-4-11	15-0-13	0-28-0	19-2-7	7-0-21	3-0-25
NB	8-0-20	5-4-19	8-1-19	7-2-19	0-28-0	3-0-25	2-0-26
SMVE	22-1-5	22-0-6	22-0-6	21-0-7	25-0-3	0-28-0	1-9-18
WMVE	25-1-2	27-0-1	25-0-3	25-0-3	26-0-2	18-9-1	0-28-0

Compared with SMVE, 27 out of the 28 cases WMVE approach is ranked as the highest or equal, 96% of all cases. WMVE was fallen behind SMVE in only one dataset. This results show clearly that assigned weights to each classifier based on its performance consequently improve the classification accuracy. Here, we note that more significant differences between WMVE and SMVE are expected to be observed if the optimal input parameters can be identified for each algorithm and for each dataset, instead of using default parameters.

V. CONCLUSION AND FUTURE WORK

In this study, a novel weighted majority voting approach is proposed for ensemble learning to increase vote effects of correctly classifying algorithms. The proposed approach, called WMVE, consists of three phases. The first phase is to train classifiers on training set. The second phase is to determine the weights of the classifiers using validation set. In this phase, the weights of the classifiers that correctly predict class label of an instance are incremented by the ratio of the number of incorrectly predicting classifiers to the whole number of classifiers. The third phase is to combine the outputs of individual classifiers by considering their weights.

The proposed approach (WMVE) was applied on 28 commonly used datasets of different sizes. It was compared with SMVE and five popular classification algorithms. Among the methods applied in this study, WMVE has superiority in terms of classification performance. The results obtained from the experimental studies favor the usage of ensembles and claims that the proposed WMVE approach is promising. It is apparent that this approach is a great candidate to improve SMVE in many domains.

As a future work, instead of using default parameters, the optimal input parameters can be identified for each algorithm and for each dataset. In this way, classification accuracy of the proposed approach can be improved slightly. As future work, WMVE approach can be further tested in ensembles with different sizes and by using different classification algorithms. Another future work may be conducted by applying feature selection techniques in WMVE.

REFERENCES

[1] S. Satapathy, A. Jagadev, and S. Dehuri, “Weighted Majority Voting Based Ensemble of Classifiers Using Different Machine Learning Techniques for Classification of EEG Signal to Detect Epileptic Seizure”, Informatica, vol. 41, pp. 99-110, 2017.

[2] P. Baez, C. Araujo, and P. Lopez P., “Neural Network Ensembles with Missing Data Processing and Data Fusion Capacities”, Applications in Medicine and in the Environment, International Work-Conference on Artificial Neural Networks, Malaga, Spain, pp. 169-176, 8-10 June 2011.

[3] A. Aburomman and M. Reaz, “A survey of intrusion detection systems based on ensemble and hybrid classifiers”, Computers & Security, vol. 65, pp. 135-152, 2017.

[4] M. Zia, M. Hussain, and M. Jaffar, “A novel spontaneous facial expression recognition using dynamically weighted majority voting based ensemble classifier”, Multimedia Tools Application, vol. 77, pp. 25537-25567, 2018.

[5] A. Onan, S. Korukoğlu, and H. Bulut, “A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification”, Expert Systems with Applications, vol. 62, pp. 1-16, 2016.

[6] S. Moustafa, M. ElNainay, N. Makky and M. Abougabal, “Software bug prediction using weighted majority voting techniques”, Alexandria Engineering Journal, vol. 7, pp. 2763-2774, 2018.

[7] M. Awad, M. ElNainay, and M. Abougabal, “Predicting Bug Severity using Customized Weighted Majority Voting Algorithms”, Japan-Africa Conference on Electronics, Alexandria, Egypt, pp. 170-175, 18-20 December 2017.

[8] R. Burduk, “Recognition Task with Feature Selection and Weighted Majority Voting Based on Interval-Valued Fuzzy Sets”, International Conference on Computational Collective Intelligence, Ho Chi Minh City, Vietnam, pp. 204-209, 28-30 November 2012.

[9] A. Kausar, M. Ishtiaq, M. Jaffar, and A. Mirza, “Optimization of Ensemble based Decision using PSO”, World Congress on Engineering, London, UK, vol. 1, pp. 671-676, June 30-July 2 2010.

[10] Y. Mehmood, M. Ishtiaq, M. Tariq, and M. Jaffar, “Classifier Ensemble Optimization for Gender Classification using Genetic Algorithm”, International Conference on Information and Emerging Technologies, Karachi, Pakistan, pp. 39-44, 14-16 June 2010.

[11] Y. Zhang, H. Zhang, J. Cai, and B. Yang, “A Weighted Voting Classifier Based on Differential Evolution”, Abstract and Applied Analysis, vol. 2, pp. 1-6, 2014.

[12] L. Kuncheva and J. Rodriguez, “A weighted voting framework for classifiers ensembles”, Knowledge Information Systems, vol. 38, pp. 259-275, 2012.

[13] J. Gutierrez, D. Mateos-Garcia, M. Garcia, and J. Riquelme-Santos, “An evolutionary weighted majority voting and support vector machines applied to contextual classification of LiDAR and imagery data fusion”, Neurocomputing, vol. 163, pp. 17-24, 2015.

[14] F. Ye, “Board-Level Functional Fault Diagnosis Using Artificial Neural Networks, Support-Vector Machines, and Weighted-Majority Voting”, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 31 issue 5, pp. 723-736, 2013.

[15] N. Toth and B. Pataki, “Classification confidence weighted majority voting using decision tree classifiers”, International Journal of Intelligent Computing and Cybernetics, vol. 1 issue 2, pp. 169-192, 2007.

[16] M. Tahir, J. Kittler, and A. Bouridane, “Multilabel classification using heterogeneous ensemble of multi-label classifiers”, Pattern Recognition Letters, vol. 33, pp. 513-523, 2012.

[17] R. Polikar, “Ensemble based systems in decision making,” IEEE Circuits and Systems Magazine, vol. 6, issue 3, pp. 21-45, 2006.

[18] E. Frank, M. Hall, and I. Witten, The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques", 4th ed, Morgan Kaufmann: Cambridge, USA, 2016, ISBN 9780128043578.

[19] D. Dua and C. Graff, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science. Accessed on April 2019.