INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OPEN ACCESS

Received: 12-01-2023

Accepted: 08-07-2023

Published: 01-09-2023

Citation: Sekhar RA, Sreeni KG (2023) An Entropy Based Alternate Approach to Develop an Ensemble Classifier. Indian Journal of Science and Technology 16(33): 2601-2608. https://doi.org/10.17485/IJST/v16i33.85

[*] **Corresponding author**.

arvndskhr@gmail.com, aravindskhr.87@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

An Entropy Based Alternate Approach to Develop an Ensemble Classifier

R Aravind Sekhar[1*], **K G Sreeni**[1]

1 College of Engineering, Trivandrum, India

Abstract

Objectives: To develop an alternative approach to improve the performance of ensemble classifiers. **Methods:** The Principle of Maximum Entropy and a precision based diversity measure are used to develop the model. Here joint entropy values are used which strongly reflect the combined uncertainties in the actual prediction and the prediction obtained during the training phase. Towards that purpose probabilistic confusion matrix has been used. The performance of the proposed method is tested against twenty different datasets. Each dataset is different from one another in terms of the number of classes and features. The results have been compared with several existing state of art ensemble methods uch as Bagging, AdaBoost, Naive Bayes (NB), Weighted Majority Vote (WMV) and TransEnsemble Classifier (TrEnL). **Findings:**The results show that the proposed method can select the best among the classifiers and perform better when compared to conventional classifiers. **Novelty:** The maximum entropy principle is applied at the final prediction probability matrix so that the method can be used as a generalized technique to improve the performance of ensemble classifiers.

Keywords: Ensemble; Maximum Entropy; Confusion Matrix; Weighted Entropy; Probabilistic Confusion Matrix

1 Introduction

Ensemble methods are techniques that combine several classifiers to obtain the best results[1]. Such methods are widely used in the field of machine learning and deep learning to improve accuracy to a great extent[2,3]. In ensemble learning individual classifiers known as the base learners are developed and then combined to obtain the final prediction. Once the base learners are developed they are combined using a scored or decision level ensemble which then gives the final prediction for each testing sample. Ensemble methods have been found very much useful while dealing with datasets with high dimensions[4,5]. Some of the areas where ensemble classifiers are used include medical fields, weather forecast, metro traction energy prediction etc.[6,7]. In ensemble classifiers, accuracy and diversity are considered the main factors for better classification[8]. Also, the decision on selecting the individual classifiers depends on the weights assigned to them. The weight vector thus plays a crucial role in selecting the best among the classifiers. Many ensemble-based methods are developed to give a better weight vector to the individual classifiers[9]. Such methods include diversity analysis, sparse regularisation, transformed ensemble learning, and so on.

One of the main challenges in ensemble classification is the size of the ensemble pool [10,11]. If ensemble size increases it in turn increases the computational complexity and computational time [12]. Achieving better classification accuracy with reduced ensemble size is a research area that needs to be well investigated [13,14]. The main limitation of the state of art ensemble classifier is the trade off between accuracy and ensemble size. An increase in accuracy can be achieved only at the cost of increased ensemble size. Here the proposed method aims to achieve better classification accuracy with a reduced ensemble size. Another challenge in ensemble classification is the tradeoff between diversity and accuracy. The proposed method aims to find a solution to the challenges faced through a novel ensemble classifier using a weighted maximum entropy approach. Here unlike the other entropy based approaches the proposed method uses joint entropy values of the prediction probability matrix and probabilistic confusion matrix for the final prediction. A comparative analysis is made between the proposed classifier and the state of art classifiers such as Bagging, AdaBoost, Naive Bayes (NB), Weighted Majority Vote (WMV) and TransEnsemble Classifier (TrEnL) on the twenty different datasets mentioned in Table 1.

The remaining paper is organized as follows. Section 2 explains the proposed method followed by the results and discussions in section 3. Conclusions and future work are presented in section 4.

2 Methodology

This section explains a Maximum Entropy Ensemble Classifier (MEEC) used in the proposed method. The classifier model is shown in Figure 1. Here the test input is given to individual classifiers, and corresponding prediction vectors are generated. For each classifier, a joint probability vector is found from the probabilistic confusion matrix and the classifier prediction vector. The entropy values are further calculated from the joint probability vectors. Then a novel diversity measure is determined based on the class precision which selects only diverse classifiers from the ensemble pool. The determination of diversity measure is based on Algorithm 1 which is explained in section II-A1. Finally, the weighted entropy values are calculated and a decision is made based on the maximum entropy principle as per Algorithm 2 (see section II-A1), which selects the classifier with the best prediction.

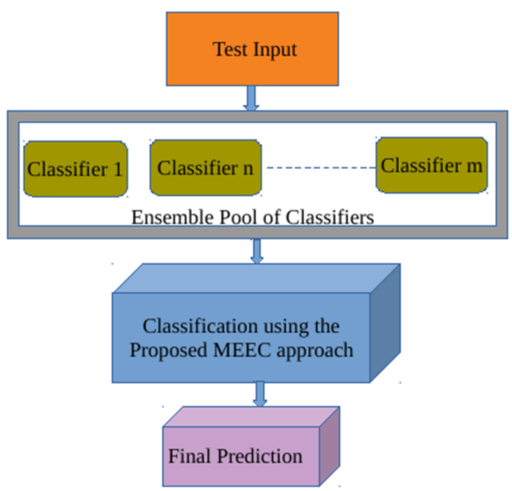

Fig 1. Architecture of the Proposed Ensemble Classifier

2.1 MEEC Approach

In the block diagram shown in Figure 1 there are m different classifiers, each classifier produces a prediction probability vector corresponding to a given test input. The predicted class label is the class corresponding to the one with the maximum probability.

1) Diversity Measure: In the proposed approach, a novel diversity measure has been generated using the class precision values which are obtained from the confusion matrices of the corresponding classifiers. Two classifiers are said to be diverse if their predictions are entirely different for a given test input. For each trained classifier there corresponds a confusion matrix from which several performance metrics can be evaluated. One such metric is the individual class precision score which represents the ability of a classifier to predict the classes accurately. Class precision is computed as the ratio of true predictions to the sum

of true predictions and false predictions corresponding to a class. The maximum value of precision is 1 and the minimum value is 0. An increase in class precision of a classifier indicates an increase in its ability to predict the corresponding class . In a confusion matrix, every row corresponds to the expected class and every column represents the actual class. Let 'CF_m ' denote the confusion matrices of the mth classifier as shown in equation 1.

$$CF_m = \begin{matrix} C_m(0,0)\ldots & C_m(0,k)\ldots & C_m(0,n-1) \\ C_m(1,0)\ldots & C_m(1,k)\ldots & C_m(1,n-1) \\ . & . & . \\ . & . & . \\ C_m(n-1,0)\ldots & C_m(n-1,k)\ldots & C_m(n-1,n-1) \end{matrix} \quad (1)$$

Let α_{mk} denotes the precision score of m th classifier to predict the class label 'k '. The determination of α_{mk} from CF_m is given in equation 2.

$$a_{mk} = C_m\,(k_1,k)\,/\,(C_m(0,k) + (C_m(1,k) + \ldots\ldots (C_m(n-1,k)) \tag{2}$$

Similarly, class precision scores can be generated for all class labels. Using the class precision scores, a diversity measure is determined as per Algorithm 1.

Algorithm 1: Selecting the Diverse Classifiers for a given test input.

Input:Predicted class labels of the base classifiers for a test input.

Output:Selecting only diverse classifiers from the ensemble pool.

1. A test input is given to all the classifiers and their corresponding predicted class labels are noted.

2. Form n groups, denoted by X_0 , X_1 , ...X_i , ...X_{n-1} corresponding to n classes where X_i denote the group containing the classifiers whose predicted class label is i .

3. For the set of classifiers in the group X_i select only those classifier/classifiers with maximum class precision score (which was obtained during the training phase). Repeat the process to select classifiers from every other group.

4. Selected classifiers are added to the ensemble pool such that the pool contains only diverse classifiers.

Thus for a given test input, the ensemble pool contains only diverse classifiers so that the computational complexity can be reduced to a great extent.

2) Probabilistic Confusion Matrix and Entropy: The probabilistic confusion matrix is obtained by dividing each row of the confusion matrix by the corresponding number of test inputs. The elements, 'δ_m(i , j)' of the probabilistic confusion matrix, Δ_m is given in equation 3.

$$\delta_{mk}(i,j) = [C_m(i,j)]\,/t(i); \quad for alli,j,\varepsilon\{0,1\ldots..n-1\} \tag{3}$$

where *t(i)* denotes the number of test inputs belonging to class *i* . Here '*i* ' and '*j* ' denote the rows and columns, respectively of the matrix, *CFm* . In a probabilistic confusion matrix for every i^{th} row

$\sum \quad \sum_{j=0}^{n-1} \delta_m\,(i_\pm j) = 1.$

Let us consider an unknown test input given to the classifiers in the ensemble pool. The prediction of each classifier and their corresponding prediction probability vectors are noted. Let the corresponding predicted class label of the mth classifier Cl_m be represented by k and the corresponding predicted probability vector by p_{mk} . Let the joint probability vector of the classifier Cl_m with predicted class label k be denoted by $\mathbf{h}_{mk}$. Let the kth row of the probabilistic confusion matrix, Δ_m be denoted by δ_{mk} , where δ_{mk} = [δ_m(k , 0), δ_m(k , 1)...δ_m(k , n − 1)] and

$$\Delta_m = \begin{matrix} \delta_{m0} \\ \delta_{m1} \\ \ldots\ldots \\ \delta_{mk} \\ ---- \\ \delta_{mn} \end{matrix}$$

Now the joint probability vector can be determined from p_{mk} and δ_{mk} as per the equation 4.

$$\mathbf{h}_{mk} = p_{mk}\delta_{mk} \tag{4}$$

where $\mathbf{h}_{mk}$ = [h_{mk}^{0}, h_{mk}^{1}.........h_{mk}^{n-1}], represents the joint probability prediction vector of classifier C_{lm} with predicted class label k . Similarly, for every prediction corresponding to a classifier the joint probability prediction vector can be determined. The entropy corresponding to the joint prediction vector $\mathbf{h}_{mk}$ be denoted by e_{mk} and is calculated as per equation 5.

$$e_{mk} = \sum^{n-1}{}_{i=0} h^{i}_{mk} \log\left(h^{i}_{mk}\right) \tag{5}$$

3) Weighted Entropy: In the proposed Maximum Entropy Ensemble Classifier (MEEC) the classifier with maximum weighted entropy is taken for the final prediction. The weights are determined from the equation 2. The weighted entropy, denoted by'ε_{mk}' corresponding to e_{mk} is determined as per equation 6. Similarly, the weighted entropy values are calculated for every classifier corresponding to each prediction. Now a decision rule is formulated as per Algorithm 2 based on the principle of maximum entropy.

$$\varepsilon_{mk} = \alpha_{mk} e_{mk} \tag{6}$$

Algorithm 2: Selecting the best Classifier for a test input using the entropy-based Decision Rule.
Input: Entropy Values of the Classifiers Corresponding to a Predicted Class.
Output: Prediction Vector Corresponding to the Classifier with the best Prediction.
1. Compute the joint probability vector for each classifier corresponding given test input.
2. Determine the entropy for each joint probability vector.
3. Convert the entropy to weighted entropy as per equation 6
4. Select the prediction vector of the classifier with maximum weighted entropy.

Table 1. The details of different datasets used for the study

Dataset	Number of classes	Number of Features
Iris	3	4
Cancer	2	30
Diabetes	2	8
Air	3	64
Vehicle	4	18
Glass	6	24
Wine	3	13
Letter	26	16
Sonar	2	60
Heart	2	13
Ionosphere	2	34
Musk	2	166
Page	5	10
Phy	2	78
Waveform	3	21
Wbcd	2	9
Wdbc	2	30
Segment	6	18
Spambase	2	57
Zoo	7	16

2.2 Datasets

The experimental evaluation is done with twenty different datasets, the details are given in Table 1 . The number of features used and the number of classes for each dataset are also shown in the table. The datasets can be downloaded from http://www.ics.uci.edu/mlearn/MLRepository.

3 Result and Discussion

The proposed Maximum Entropy Ensemble Classifier (MEEC) has been implemented using python3.7/jupyter notebook. The proposed method is compared against five different ensemble classifiers such as Bagging, AdaBoost, Weighted Majority Vote (WMV), Naive Bayes Classifier (NBC), and Transformed Ensemble Learning Classifier (TrEnL), and is shown in Table 2 . A Support Vector Machine (SVM) is used as the base learner for developing the ensemble classifiers in all cases. In bagging the base classifiers are fitted each on a random subset of original data and then form the final prediction by aggregating base classifiers predictions. On the other hand in Ada Boost classifier the output of the base classifiers is combined iteratively to produce the final output based on an optimized differentiable loss function. In WMV the classifiers are combined using the weights based on the probability of correct classification while in NBC the prediction is done based on the maximum probability determined using the individual classification probability for each class. In TrEnL multiple base learners are converted to a linear transformation of base learners and then assigned optimal weights.

Table 2. Comparison of the accuracy values (in %) of the proposed MEEC approach with other state of art ensemble classifiers. Bold entries in each row indicate the best performance with the corresponding dataset

Dataset	Bagging	AdaBoost	NBC	WMV	TrEnL	MEEC
Iris	97.7	94.8	90.1	97.1	97.83	**98.2**
Cancer	73.8	69.9	72.3	73.7	**77.13**	76.5
Diabetes	75.7	76.1	77.1	75.4	77.8	**78.7**
Air	93.4	**98.6**	84.3	92.4	92.3	98.3
Vehicle	75.7	76.1	54.7	75.6	75.9	**77.7**
Glass	57.3	57.6	40.1	56.8	56.5	**60.3**
Wine	96.1	**96.2**	72.3	**96.2**	**96.2**	**96.2**
Letter	90.9	95.1	65.2	91.8	92.6	**96.4**
Sonar	79.7	81.1	77.1	79.4	**81.7**	80.7
Heart	81.9	78.2	80.7	81.4	81.5	**84.7**
Ionosphere	92.1	91.2	92.7	92.4	93.2	**94.1**
Musk	77.3	79.3	78.4	77.9	79.7	**80.6**
Page	94.2	94.5	93.8	94.1	94.6	**96.1**
Phy	70.9	69.3	70.1	70.4	70.8	**72.2**
Waveform	83.9	85.2	83.7	83.4	83.8	**85.7**
Wbcd	98.2	98.3	98.2	**98.4**	**98.4**	**98.4**
Wdbc	94.8	94.8	93.5	94.4	95.6	**95.7**
Segment	88.3	89.1	80.7	88.2	90.4	**91.2**
Spambase	89.9	86.2	89.5	89.4	89.7	**91.6**
Zoo	87.1	**89.5**	80.9	87.4	87.8	**89.5**

All these classifiers are individually tested on the twenty different datasets as shown in Table 2. From the table, it can be seen that the proposed MEEC approach provides much better classification accuracy in the majority of the cases.

The performance of the proposed MEEC has been tested on the datasets by computing entropy with and without weights. From Figure 2 it can be seen that the proposed MEEC approach without weights, has no significant advantage in a majority of the cases. But the use of weighted entropy values in prediction improves the accuracy significantly as shown in the figure. This specifies the importance of adding weights to the entropy values.

Table 2 shows the comparison of classification accuracy of the state of art methods with the proposed MEEC approach. The proposed MEEC approach outperforms the other classifiers in a majority of cases. Among the twenty datasets tested, with the Air quality dataset, the AdaBoost performs better while with the Sonar dataset, TrEnL provides the best classification accuracy. In all other cases, the proposed MEEC has better classification accuracy compared to other classifiers. The performance of the proposed MEEC was also evaluated using Mathew's Correlation Coefficient (MCC).

Table 3 shows the MCC of the various ensemble classifiers before and after applying the proposed MEEC technique. MCC is a measure of the quality of the classifier's performance on both binary and multiclass data. It takes into account both true and false positives and negatives and can generally produce a balanced measure even for classes of different sizes. From Table 3 it can be seen that the proposed technique improves the MCC score to a great extent.

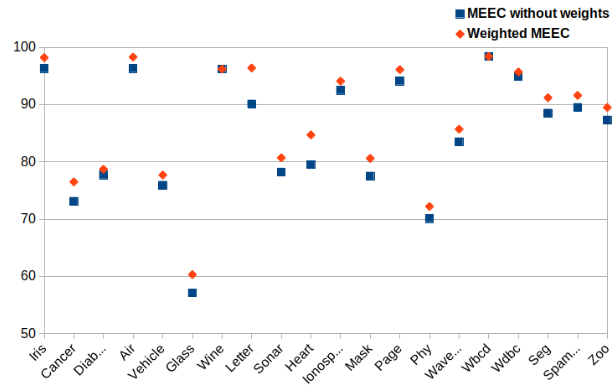

Fig 2. Comparison of the classification accuracy of the proposed MEEC approach on different datasets by using weighted entropy and entropy without using the weights

Table 4 shows the performance in terms of the accuracy of the proposed MEEC based on different ensemble sizes. The ensemble sizes used are 25, 50, 75, 100, 200, 300, and 500, and the highest accuracies in each case are indicated in bold entries. The MEEC was able to achieve the best performance (16 out of 20 datasets) with an ensemble size of 200. With an ensemble size of 500, the MEEC has the highest accuracy for four datasets. This shows that increasing the ensemble size has less effect on the MEEC performance. Achieving the best performance with a reduced ensemble size also helps to reduce computational complexity.

Table 3. Comparison of the Mathew's Correlation Coefficients (MCC) of MEEC with other state of art ensemble classifiers on different datasets. Bold entries indicate the best performance with each dataset

	MCC of Different Classifiers on each dataset					
Dataset	**Bagging**	**AdaBoost**	**WMV**	**NBC**	**TrEnL**	**MEEC**
Iris	0.953	0.945	0.901	0.982	0.977	**0.991**
Cancer	0.721	0.695	0.733	0.729	0.731	**0.755**
Diabetes	0.743	0.766	0.752	0.732	0.755	**0.795**
Air	0.882	0.953	0.914	0.812	0.921	**0.975**
Vehicle	0.744	0.753	0.736	0.548	0.755	**0.765**
Glass	0.601	0.612	0.615	0.477	0.622	**0.637**
Wine	0.932	0.965	0.711	0.962	0.966	**0.968**
Letter	0.882	0.915	0.663	0.892	0.922	**0.943**
Sonar	0.812	**0.823**	0.761	0.798	0.798	0.802
Heart	0.816	0.775	0.793	0.813	0.804	**0.856**
Ionosphere	0.912	0.914	0.919	0.922	0.915	**0.946**
Musk	0.754	0.775	0.772	0.764	0.788	**0.797**
Page	0.935	0.929	0.927	0.933	0.944	**0.952**
Phy	0.698	0.701	0.705	0.697	0.702	**0.718**
Waveform	0.812	0.833	0.817	0.821	0.821	**0.848**
Wbcd	0.978	0.977	0.978	**0.984**	**0.984**	**0.984**
Wdbc	0.913	0.917	0.897	0.921	0.921	**0.943**
Segment	0.853	0.867	0.799	0.866	0.867	**0.895**
Spambase	0.878	0.858	0.894	0.896	0.898	**0.905**
Zoo	0.857	**0.887**	0.811	0.862	0.878	0.885

Table 4. Performance comparison of MEEC in terms of classification accuracy on different ensemble sizes. The bold entries in each row indicate the highest accuracy with the corresponding dataset

	Accuracy values of each classifier on datasets for different ensemble sizes						
Dataset	**(Ensemble size=25)**	**(Ensemble size=50)**	**(Ensemble size=75)**	**(Ensemble size=100)**	**(Ensemble size=200)**	**(Ensemble size=300)**	**(Ensemble size=500)**
Iris	97.72	97.84	97.87	98.41	**98.78**	98.74	98.33
Cancer	72.32	72.46	72.77	73.47	**73.93**	73.88	73.77
Diabetes	74.27	74.43	74.78	75.21	75.48	75.67	**75.88**
Air	93.42	93.46	94.13	94.42	94.57	94.74	**94.89**
Vehicle	74.27	74.56	75.17	75.38	75.49	75.66	**75.82**
Glass	57.13	57.24	57.41	57.48	**58.54**	58.13	57.95
Wine	95.71	95.82	96.23	96.52	**96.82**	96.74	96.62
Letter	90.11	90.41	90.62	90.71	**90.84**	90.76	90.54
Sonar	79.57	79.83	79.98	80.14	**80.87**	80.42	80.11
Heart	83.23	83.42	84.17	84.45	**84.82**	84.59	84.13
Ionosphere	93.51	93.72	93.97	94.14	**94.85**	94.52	94.23
Musk	79.53	79.78	80.14	80.39	**80.87**	80.52	80.16
Page	95.62	95.95	96.18	96.31	**96.86**	96.55	96.26
Phy	71.79	71.93	80.21	80.34	**80.98**	80.72	80.24
Waveform	84.79	84.92	85.17	85.34	**85.97**	85.85	85.27
Wbcd	97.42	97.93	98.33	98.67	**98.97**	98.58	98.17
Wdbc	94.48	94.83	95.25	95.36	**95.93**	95.66	95.33
Segment	90.13	90.42	90.87	91.32	91.44	91.53	**91.88**
Spambase	90.39	90.42	90.85	91.24	**91.87**	91.55	91.26
Zoo	88.44	88.76	89.19	89.34	**89.98**	89.73	89.25

4 Conclusion

The study proposes a weighted entropy-based approach to combine multiple classifiers for better classification accuracy. Here, entropy values are used which strongly reflect the combined uncertainties in the actual prediction and the prediction obtained during the training phase. Further, for each class, the entropy values are scaled with the precision values as the corresponding weights. The decision rule is formulated which selects the best prediction from the ensemble corresponding to a given test input. Also, here a precision based diversity measure was used without affecting the accuracy. The effectiveness of the method was tested on different datasets. and the experimental results confirm that the proposed method was able to achieve better accuracy in majority of the cases. Also it can be seen that the proposed MEEC approach was able to achieve better classification accuracy with an ensemble sie of 200 while other classifiers require an ensemble size of 500 to achieve better performance. As a future scope, the method can be extended to multimodal-based ensemble classifiers.

References

1) Mienye YID, Y S. A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects. 2022. Available from: https://ieeexplore.ieee.org/document/9893798.
2) Yazdizadeh A, Patterson Z, Farooq B. Ensemble Convolutional Neural Networks for Mode Inference in Smartphone Travel Survey. *IEEE Transactions on Intelligent Transportation Systems*. 2020;21(6):2232–2239. Available from: https://doi.org/10.1109/TITS.2019.2918923.
3) Jan MZ, Munoz JC, Ali MA. A novel method for creating an optimized ensemble classifier by introducing cluster size reduction and diversity. *IEEE Transactions on Knowledge and Data Engineering*. 2020;2022(7):1–1. Available from: https://doi.org/10.1109/TKDE.2020.3025173.
4) Sridhar S, Sanagavarapu S. Detection and Prognosis Evaluation of Diabetic Retinopathy using Ensemble Deep Convolutional Neural Networks. *2020 International Electronics Symposium (IES)*. 2020. Available from: https://doi.org/10.1109/IES50839.2020.9231789.
5) Sravanthi N, Venkat ML, Harshini S, Ashesh K. An Ensemble Approach to Predict Weather Forecast using Machine Learning. *2020 International Conference on Smart Electronics and Communication (ICOSEC)*. 2020. Available from: https://doi.org/10.1109/ICOSEC49089.2020.9215444.
6) Tripathi MAP, Ansari TK, Gandhi. Ensemble Computational Intelligent for Insomnia Sleep Stage Detection via the Sleep ECG Signal. *IEEE Access*. 2022;10. Available from: https://doi.org/10.1109/ACCESS.2022.3212120.
7) Wang K, Hu T, Zhao B. An Unsupervised Deep Neural Network Approach Based on Ensemble Learning to Suppress Seismic Surface-Related Multiples. *IEEE Transactions on Geoscience and Remote Sensing*. 2022;60:1–14. Available from: https://doi.org/10.1109/TGRS.2022.3225809.
8) Tian Y, Zhang J, Chen Q, Liu Z. A Novel Selective Ensemble Learning Method for Smartphone Sensor-Based Human Activity Recognition Based on Hybrid Diversity Enhancement and Improved Binary Glowworm Swarm Optimization. *IEEE Access*. 2022;10:125027–125041. Available from: https://doi.org/10.1109/ACCESS.2022.3225652.
9) Feng B, Gu Y, Yu H, Yang X, Gao S. DME: An Adaptive and Just-in-Time Weighted Ensemble Learning Method for Classifying Block-Based Concept Drift Steam. *IEEE Access*. 2022;10:120578–120591. Available from: https://doi.org/10.1109/ACCESS.2022.3222178.

10) Zhai P, Hou T, Ji X, Dong Z, Zhang L. Robust Adaptive Ensemble Adversary Reinforcement Learning. *IEEE Robotics and Automation Letters*. 2022;7(4):12562–12568. Available from: https://doi.org/10.1109/LRA.2022.3220531.
11) Xu H, Liu Q. An Ensemble Learning Algorithm Based on Density Peaks Clustering and Fitness for Imbalanced Data. *IEEE Access*. 2022;10:116120–116128. Available from: https://doi.org/10.1109/ACCESS.2022.3219582.
12) Vinayagam A, Veerasamy V, Tariq M, Aziz A. Heterogeneous learning method of ensemble classifiers for identification and classification of power quality events and fault transients in wind power integrated microgrid. *Sustainable Energy, Grids and Networks*. 2022;31:100752. Available from: https://doi.org/10.1016/j.segan.2022.100752.
13) Pamila JCMJ, Selvi RS, Santhi P, Nithya TM. Ensemble classifier based big data classification with hybrid optimal feature selection. *Advances in Engineering Software*. 2022;173:103183. Available from: https://doi.org/10.1016/j.advengsoft.2022.103183.
14) Yang J, Wang F. Auto-Ensemble: An Adaptive Learning Rate Scheduling Based Deep Learning Model Ensembling. *IEEE Access*. 2020;8:217499–217509. Available from: https://arxiv.org/pdf/2003.11266.pdf#:~:text=Auto%2DEnsemble%20method%20refers%20to,generaliza%2D%20tion%20performance%20of%20classification.