A Novel Ensemble Classifier Approach using Weak Classifier Learning on Overlapping Clusters

Ashfaqur Rahman and Brijesh Verma

Abstract— **This paper presents a novel approach for creating and training of an ensemble classifier. The approach is based on creating atomic and non-atomic clusters at different levels, training of weak classifiers on overlapping clusters and fusion of their decisions. The subsets of data are obtained by clustering of original training data sets into multiple partitions. As each partition represents highly correlated patterns from different classes, the proposed approach trains weak classifiers on difficult–to–classify patterns and combines the decision at various levels. The approach is tested on six benchmark datasets from UCI machine learning repository. The results show that the proposed approach achieves better classification accuracy than the existing approaches.**

I. Introduction

An ensemble classifier refers to a group of weak or base classifiers that separately learn the class boundaries from a data set and their decisions on a test pattern are combined to predict the class. Ensemble classifier is also known as multiple classifier systems, mixture of experts and committee of classifiers. The goal of ensemble classifier is to perform better than its weak counterparts. The achievement of this objective depends on ensemble classifier creation and efficient fusion of the weak classifier decisions. The ensemble creation step is constrained by *diversity* [1] that forces the weak classifier errors to be uncorrelated. The fusion step is guided by rules to combine the decisions of the weak classifiers.

A number of research efforts are observed towards creation of ensemble classifiers in order to achieve diversity. Bootstrap aggregating or *bagging* [2] is one of the earliest ensemble classifier creation methods. The weak classifiers in bagging are trained on different subsets of the training data. The subsets are randomly drawn (with replacement) from the training set and their errors are uncorrelated. The weak classifiers are homogeneous in nature. There are a number of variants of bagging including random forests [3], pasting small votes [4], adaptive generation and aggregation approach [5], and fuzzy bagging [6].

Another commonly used ensemble creation method is called *boosting* [7]. Boosting creates an ensemble of classifiers by re-sampling the training data, however, by providing most informative training data for each consecutive classifier [7][8]. AdaBoost [9] is a more generalized version of boosting. It trains a classifier on instances that previous classifiers fail to classify. A number of variants of boosting can be observed in the literature including weighted instance selection [10], boosting recombined weak classifiers [11], Learn++ [12] and its variant Learn++.NC [13].

The other important aspect of ensemble classifiers is the decision fusion methods. Decisions provided by the weak classifiers are combined by the fusion methods to provide a final verdict. Majority voting is the most commonly used fusion method [1][14] where the ensemble choose the class that receives the highest number of votes. The aim of this paper is to present an ensemble classifier creation approach and we thus refrain from elaborating the fusion methods.

A careful scrutiny of the ensemble classifier creation approaches reveals that the weak classifiers are trained on subsets of the training data and the subset selection algorithm varies among the approaches. In this paper we present a novel approach to create the training subsets by using clustering. Given the clustering parameters the clustering algorithm partitions the data set into a set of non–overlapping segments. Each partition contains highly correlated data points from multiple classes that are difficult––to–classify. We propose to learn the decision boundaries in each cluster using a neural network.

The final outcome of some clustering algorithms (e.g. *k*–means) however depends on the initialization of the clustering parameters. In this regard we bring in the concept of *level*. Clustering at *n* levels implies that the data set is partitioned *n* times using *n* different sets of clustering parameters (e.g. cluster centres in *k*–means clustering). As the data is partitioned, identical data points belong to different clusters at different levels and are involved in the learning process of the different weak classifiers. The weak classifiers are thus trained on subsets of training data containing difficult–to–classify overlapping patterns and their errors are uncorrelated.

The aim of the research presented in this paper is to (i) develop an ensemble classifier based on the overlapping clustering philosophy mentioned above; (ii) explore the influence of level wise clustering on the ensemble classifier learning and prediction; and (iii) obtain a comparative performance analysis of the proposed and commonly used ensemble classifier creation methods.

Ashfaqur Rahman is with the CQUniversity, Australia (phone: +61749306508; e-mail: a.rahman@cqu.edu.au).

Brijesh Verma is with the CQUniversity, Australia (phone: +61749309058; e-mail: b.verma@cqu.edu.au).

II. The Proposed Ensemble Classifier

A. Overlapping Clusters and Ensemble Classifier

The proposed ensemble classifier creation approach is based on the notion of clustering. The objective is to partition the data set into multiple clusters and deploy a set of weak classifiers to learn the decision boundaries within each cluster. The clustering process partitions a data set into segments of highly correlated data points. The correlated data points are very close geometrically and are difficult to classify especially when patterns from multiple classes overlap. When clustering is used to partition labelled data sets (i.e. data where each pattern is associated with a class) the resultant segments can be of two types – *atomic* and *non–atomic*. An atomic cluster contains patterns that belong to the same class whereas a non–atomic cluster is composed of patterns from multiple classes. Fig 1 demonstrates an example of a data set partitioned into three clusters. Out of the three clusters at *level one* $\mathrm{C}_{1,2}$ is an atomic cluster whereas $\mathrm{C}_{1,1}$ and $\mathrm{C}_{1,3}$ are non–atomic clusters.

Once the clustering is finished the weak classifiers can be trained on non–atomic clusters whereas the class label can be memorized for the atomic clusters for later classification. The class of a test example can be predicted by first finding the appropriate cluster based on its distance from the cluster centres and then using the corresponding classifier (for a non–atomic cluster) or the class label (for an atomic cluster).

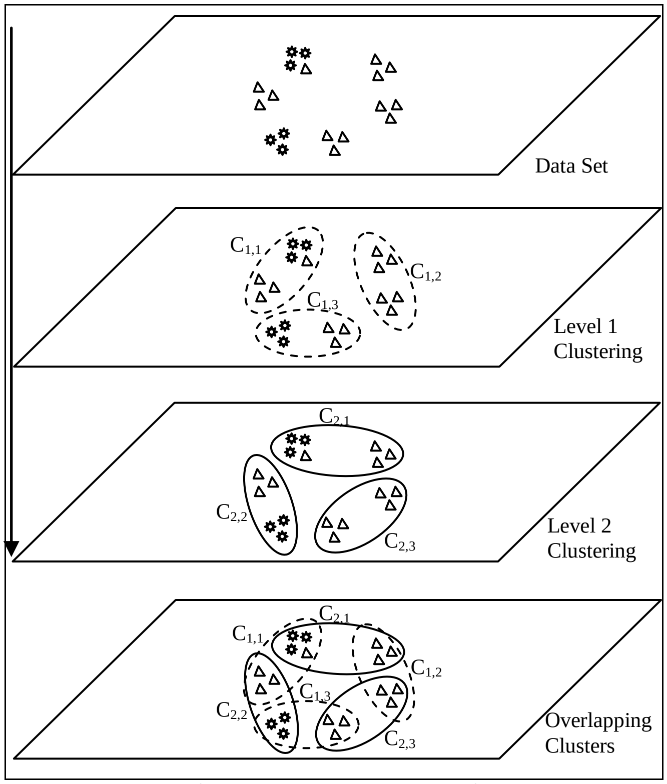


Fig 1: Clustering of data set containing patterns from two classes at different levels. At level one the data set is partitioned into three clusters with $\mathrm{C}_{1,2}$ being an atomic cluster. At level two the data set is partitioned into three different clusters with $\mathrm{C}_{2,3}$ being the atomic cluster. When overlapped it can be seen that identical data patterns belong to different clusters at different levels.

As a pattern can belong to one cluster the decision on a test example is based on the prediction of only one classifier. Although clustering identifies difficult–to–classify patterns the decision based on only one classifier prediction leaves space for improvement.

The final content of the segments in clustering algorithms depends on the initial clustering parameter settings. For example, in *k*–means clustering algorithm the final clusters depend on the initialization of the seeds (i.e. the initial state of the cluster centres). We aim to incorporate this phenomenon to improve the abovementioned decision making process in the proposed ensemble classifier. The idea is to train multiple weak classifiers on similar patterns. At this point let's introduce the concept of levels. A level indicates the partitioning of the data set based on one set of seed parameters. For example, level one clustering of the data set into three clusters is presented in Fig 1 based on the initial values of the clustering parameters Υ_1. The clusters are indexed by the level number followed by the cluster number. For example, the second cluster at level one is represented by $\mathrm{C}_{1,2}$. An alternate clustering of the same data set at level two into three segments is also parented in Fig 1 based on another set of initial clustering parameters Υ_2. Note that clusters at different level overlap and identical patterns belong to different clusters at different levels.

Classifiers are now trained independently on the non–atomic clusters at different levels. As the clusters at different levels overlap, the same patterns are included in the training process of multiple classifiers. Moreover different training subsets are used in the training of these weak classifiers and thus diversity is achieved. A test pattern also belongs to different clusters at different levels and thus gets decisions from different weak classifiers that can be fused to obtain the final verdict on its class. We use this idea for creating the ensemble of classifiers and fusing their decisions. The *novelty* of the proposed approach lies in the introduction of level–wise clustering to partition data set into alternative clusters for weak classifier learning to achieve diversity.

B. Theoretical Modelling

Let the training patterns in the data set are represented by $\Gamma = \{(\vec{x}_1, t_1), (\vec{x}_2, t_2), \dots, (\vec{x}_{|\Gamma|}, t_{|\Gamma|})\}$ where each pattern is described by a vector of n continuous valued features $\vec{x}_j = < x_{j1}, x_{j2}, \dots, x_{jn} >$ and a class label t_j with $t_j \in \{class_1, class_2, \dots, class_{N_{class}}\}$. A level is denoted by l and the K clusters at level l are denoted by $\mathrm{C}_{l,1}, \mathrm{C}_{l,2}, \dots, \mathrm{C}_{l,K}$ where $1 \leq l \leq N_{levels}$.

A pattern in the training set can be considered as a point in the Euclidian space of dimension *n*. The objective of the clustering algorithm is to group data points that are geometrically close. Given two patterns $(\vec{x}_i, t_i)$ and $(\vec{x}_j, t_j)$ in the training set a distance function d between them is defined in terms of their Euclidian distance as –

$$d(\vec{x}_i, \vec{x}_j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2}\ , \qquad (1)$$

where $\vec{x}_i = < x_{i1}, x_{i2}, \dots, x_{in} >$ and $\vec{x}_j = < x_{j1}, x_{j2}, \dots, x_{jn} >$. Assuming a set of K clusters $\{\Omega_{l,1}, \Omega_{l,2}, \dots, \Omega_{l,K}\}$ at level l, the associated cluster centres $\Upsilon_l = \{\vec{\kappa}_{l,1}, \vec{\kappa}_{l,2}, \dots, \vec{\kappa}_{l,K}\}$ are initialized randomly and the clustering algorithm aims to minimize an objective function

$$J_{L_i} = \sum_{k=1}^{K} \sum_{\forall \vec{x}_j \in \Omega_{i,k}} d(\vec{x}_j, \vec{\kappa}_{i,k}) \qquad (2)$$

for all the data points in the training set Γ.

At the end of the clustering process at level l each pattern $(\vec{x}_i, t_i)$ belongs to a cluster $\mathrm{C}_{l,k}$ where $1 \leq k \leq K$. At this point the clusters are separated into atomic and non–atomic clusters. A class label is memorized for an atomic cluster $\mathrm{C}_{l,k}$. A neural network $\theta_{l,k}$ is set up at this stage for each non–atomic cluster $\mathrm{C}_{l,k}$ to learn the decision boundaries on its patterns. Given a training pattern $(\vec{x}_i, t_i)$ that belongs to a non–atomic cluster $\mathrm{C}_{l,k}$, $\vec{x}_i$ acts as an input to the neural network whereas t_i acts as the target. Considering one hidden layer with N_H units there are a total of $n \times N_H$ links between the input layer and the hidden layer where n is the number of features. The weights of the links are represented by a weight matrix $\omega_{I,H}$. Similarly there are a total of $N_H \times N_{class}$ links between the hidden layer and the output layer and the corresponding weight matrix is represented by $\omega_{H,O}$. The weight matrices are initialized randomly and updated using back–propagation method for finding an appropriate mapping between the data points and the corresponding class as supervised by all the patterns that belong to $\mathrm{C}_{l,k}$.

A test pattern $\vec{x}$ is classified by first finding the appropriate cluster at each level. For this the distance between $\vec{x}$ and the centre of each cluster $\vec{\kappa}_{l,k}$ is computed using (1) and the appropriate cluster at level l is selected as –

$$\hat{C}_{l,k} = \underset{\vec{\kappa}_{l,k}}{\arg\max}\; d(\vec{x}, \vec{\kappa}_{l,k}) . \qquad (3)$$

If $\hat{\mathrm{C}}_{l,k}$ is an atomic cluster the memorized class label P_l is predicted at level l. If $\hat{\mathrm{C}}_{l,k}$ is a non–atomic cluster the corresponding neural network $\theta_{l,k}$ trained on $\hat{\mathrm{C}}_{l,k}$ is used to predict the class label P_l at level l. Upon receiving the predictions $\{P_l\}$ from all the N_{levels} levels, the decisions are fused into a final verdict using the majority voting fusion rule.

C. Learning and Prediction

Based on the above philosophy the learning and prediction phase of the proposed approach are presented in Fig 2 and Fig 3 respectively. The training data set is clustered at L separate levels. At each level the data is segmented into K clusters based on clustering parameters (e.g. initial state of the cluster centres). A cluster analyser then identifies atomic and non–atomic clusters. The class label is recorded for atomic clusters. A neural network is trained on the patterns of a non–atomic cluster.

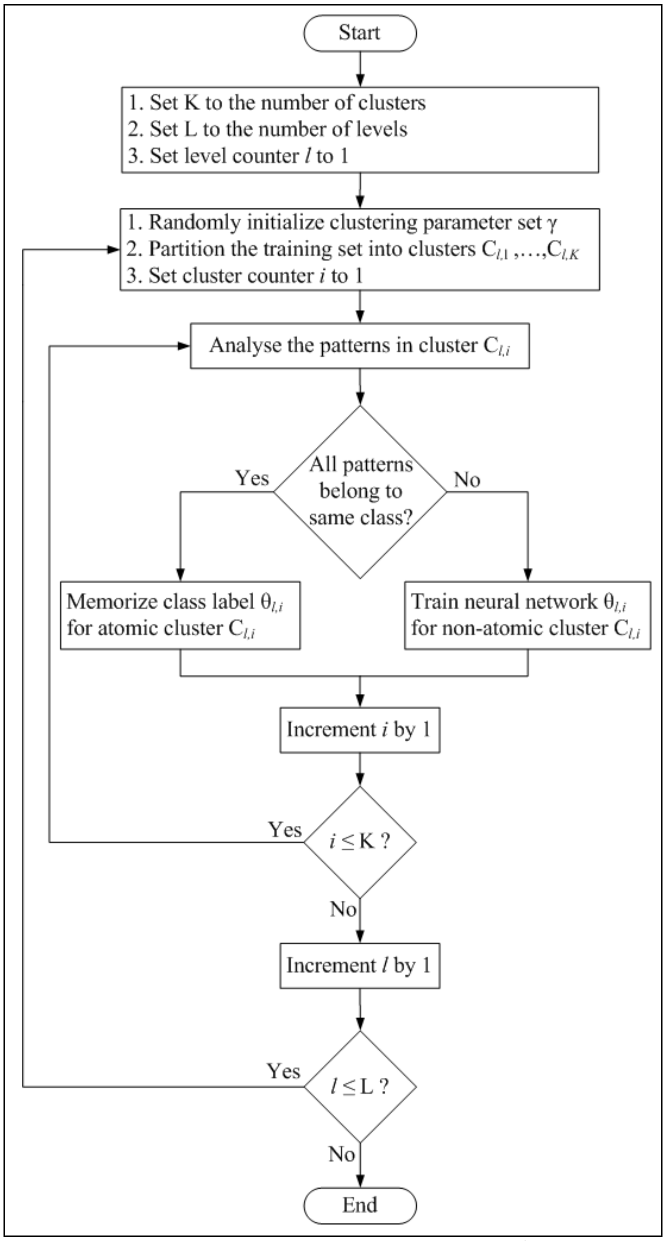

Start
1. Set K to the number of clusters
2. Set L to the number of levels
3. Set level counter l to 1
1. Randomly initialize clustering parameter set γ
2. Partition the training set into clusters $\mathrm{C}_{l,1}, \dots, \mathrm{C}_{l,K}$
3. Set cluster counter i to 1
Analyse the patterns in cluster $\mathrm{C}_{l,i}$
All patterns belong to same class?
Yes
No
Memorize class label $\theta_{l,i}$ for atomic cluster $\mathrm{C}_{l,i}$
Train neural network $\theta_{l,i}$ for non-atomic cluster $\mathrm{C}_{l,i}$
Increment i by 1
$i \leq$ K ?
Yes
No
Increment l by 1
$l \leq$ L ?
Yes
No
End

Fig 2: Learning in the proposed ensemble classifier.

During prediction (Fig 3) the appropriate cluster for the test pattern is identified at each level. If the selected cluster is atomic the pre–recorded class is predicted as P_l. If the cluster is non–atomic the corresponding neural network predicts the class P_l. Once the prediction is received from all the L levels the final verdict is obtained from $\{P_l\}$ using the majority voting rule implemented using statistical mode function.

III. Experimental Setup

We have conducted a number of experiments on benchmark data sets to verify the strength of the proposed ensemble classifier. We have compiled the datasets as used in contemporary research works [11][14] from the UCI Machine Learning Repository [15]. A summary of the data sets is presented in Table I. We used 10–fold cross

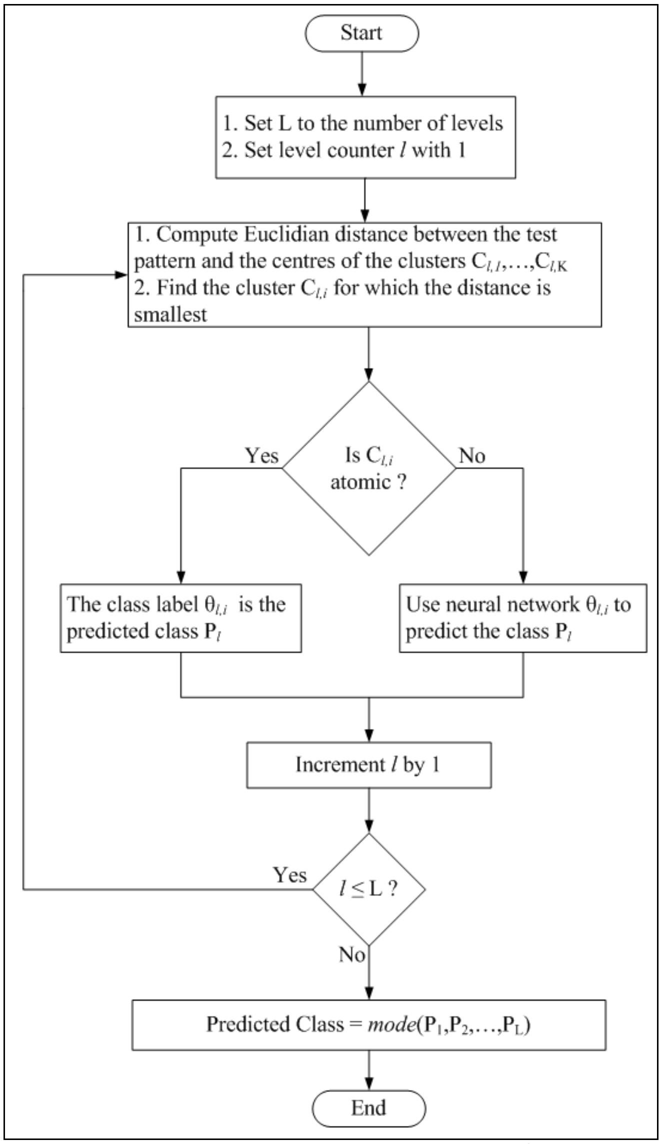

Start
1. Set L to the number of levels
2. Set level counter l with 1
1. Compute Euclidian distance between the test pattern and the centres of the clusters $C_{l,1},\ldots,C_{l,K}$
2. Find the cluster $C_{l,i}$ for which the distance is smallest
Is $C_{l,i}$ atomic ?
Yes
No
The class label $\theta_{l,i}$ is the predicted class P_l
Use neural network $\theta_{l,i}$ to predict the class P_l
Increment l by 1
$l \leq L$?
Yes
No
Predicted Class = $mode(P_1,P_2,\ldots,P_L)$
End

Fig 3: Prediction in the proposed ensemble classifier.

validation approach for all the data sets for reporting the results. For clustering we used the *k*–means clustering algorithm.

A neural network with a single hidden layer was used in the experiments. Training of the weights was achieved using a backpropagation learning algorithm. The following parameter settings were used during the training process for all data sets: (a) No. of hidden units =5, (b) Learning rate = 0.01, (c) Momentum = 0.4, (d) Epochs i.e. No. of iterations = 25 and (e) RMS goal = 0.00001. Note that the main objective of the experiment was to find the influence of overlapping clustering on ensemble classifier learning and we thus restrict ourselves to parameter settings that perform best on all the data sets found by trial–and–error basis.

As we are using majority voting rule, odd number of levels were used. Experiments were conducted with 1, 3, 5, 7, and 9 levels. At each level, data was partitioned into ten clusters. To reduce the variability of the classification accuracies, all the experiments were conducted ten times on each data set and their average performance is reported in this paper. All the experiments were conducted on MATLAB 7.5.0.

IV. Results and Discussion

In this section we present some experimental results to demonstrate the effectiveness and superiority of the proposed approach. The concept of overlapping clustering is based on the changes in cluster content at different initialization of clustering parameters and we present some results to demonstrate the changes w.r.t. changes in initialization of cluster centres using *k*–means clustering. We also demonstrate some results to show how the change in number of layers influence the classification accuracy of the proposed ensemble classifier. The classification results on benchmark data sets obtained with the proposed ensemble classifier are compared against single as well as classical ensemble classifiers like *bagging* and *boosting* at the end of this section. Neural network was used as the weak classifier with bagging and boosting and the experiments were conducted using WEKA [16].

A. Impact of Levels on Ensemble Classifier Learning

Fig 4 represents the class–cluster co–occurrence matrices obtained at different levels for some data sets namely *Breast Cancer*, *Ionosphere*, and *Sonar*. Note that the content of the clusters changes in all cases as the clustering parameters are initialized randomly at different levels. Identical patterns belonging to different clusters indicate dissimilar learning of the weak classifiers making their errors uncorrelated thus achieving diversity.

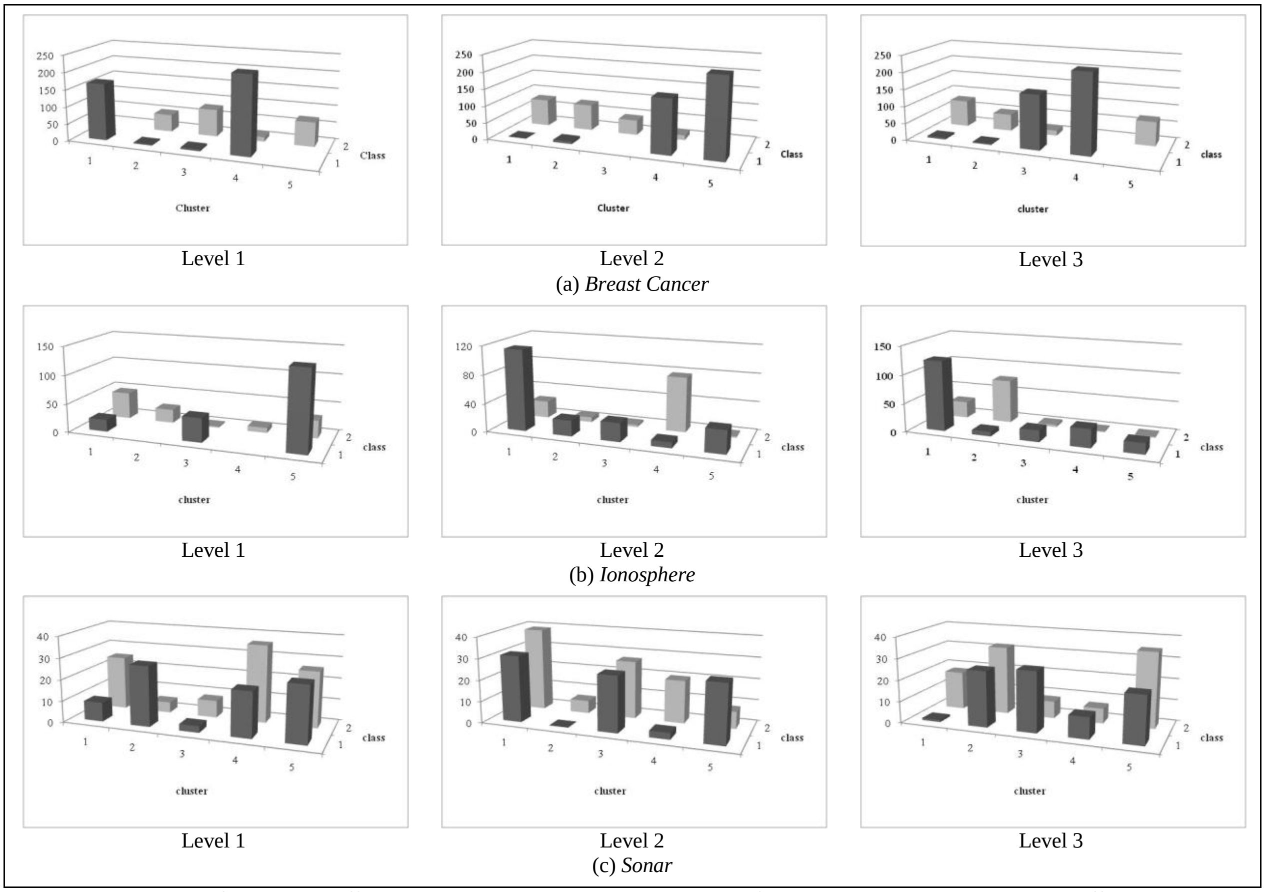
Fig 5 represents the best classification accuracies achieved as the data sets are partitioned into 1, 3, 5, 7 and 9 levels. All the graphs in Fig 5 provide a positive change of classification accuracy at higher number of cluster levels. At higher number of levels, more experts (i.e. Neural Networks) are trained on identical but disjoint patterns and thus achieve diversity leading to higher classification accuracy. Table II represents the standard deviation of the classification accuracy w.r.t the change of number of levels for the data sets in Table I. The variation is highest for the *Vowel* data set. Data sets like *Sonar* and *Ionosphere* also enjoy relatively higher variation than the other data sets. This implies that these data sets have adequate overlapping patterns and (i) it becomes easier to learn the decision boundaries with clustering and (ii) as identical and overlapping patterns are learned by multiple classifiers at different levels the final verdict becomes accurate. Relatively smaller although positive impact is observed for some data sets like *Breast*

Table I: Data sets used in the experiments.

Dataset	Instances	Attributes	Classes	Test process
Breast Cancer	699	9	2	10–fold cv
Ionosphere	351	33	2	10–fold cv
Vowel	528	13	11	10–fold cv
Sonar	208	60	2	10–fold cv
Waveform	5000	21	3	10–fold cv
Wine	178	13	3	10–fold cv

Cancer and *Waveform*.

Table III depicts the number of clusters and number of levels at which the best classification accuracies as presented in Fig 5 are achieved on the test cases. In general for all the data sets higher number of levels (minimum is 7 in Table III) imply better classification accuracy. This can be attributed to the fact that the inclusion of more cluster levels implies the learning of identical but disjoint patterns by multiple classifiers. This phenomenon is called diversity in ensemble classifiers and leads to better classification accuracy. Data sets like *Ionosphere* and *Waveform* where the patterns are already well separated the best classification accuracies are achieved at no clustering. *Breast Cancer*, *Sonar* and *Vowel* For data sets enjoy the best accuracies at relatively large number of clusters.



Level 1 Level 2 Level 3

(a) *Breast Cancer*

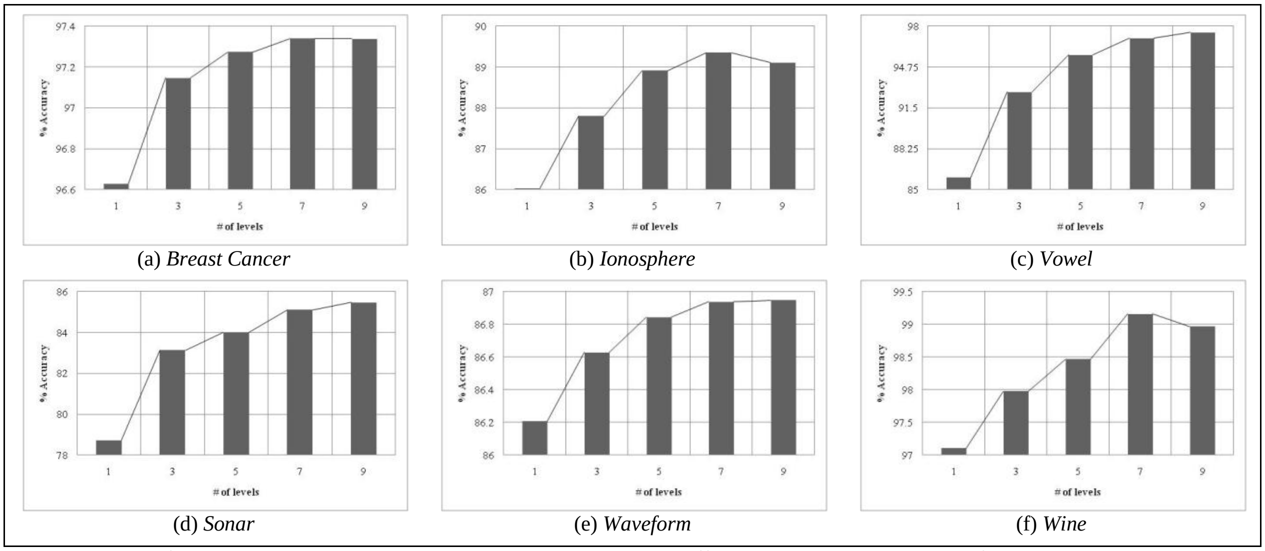
Level 1 Level 2 Level 3

(b) *Ionosphere*

Level 1 Level 2 Level 3

(c) *Sonar*

Fig 4: Clustering of data sets at different levels with the proposed ensemble classifier as the patterns change clusters at different levels.



(a) *Breast Cancer* (b) *Ionosphere* (c) *Vowel*

(d) *Sonar* (e) *Waveform* (f) *Wine*

Fig 5: Best classification accuracies achieved as the data sets are clustered at different levels and their decisions fused using majority voting.

Table II: Standard deviation of classification accuracy for different data sets with respect to the number of levels.

Data Set	std (Accuracy)
Breast Cancer	0.30
Ionosphere	1.37
Vowel	4.75
Sonar	2.71
Waveform	0.31
Wine	0.83

Table III: Number of levels and number of clusters at which the classification accuracies presented in Fig 5 are achieved.

Data Set	Level	Cluster
Breast Cancer	7	8
Ionosphere	7	1
Vowel	9	10
Sonar	9	9
Waveform	9	1
Wine	7	2

B. *Comparative Analysis*

In order to position the proposed ensemble classifier we have compared the performance against single as well as ensemble classifiers on the data sets in Table I. Neural network was used as a single classifier to learn the data sets and the performance is compared against the proposed ensemble classifier in Table IV. The ensemble classifier significantly outperforms the single classifier and the average improvement is 10.84%. As the decisions from a set of diverse and accurate base classifiers are fused to the final class verdict in the proposed approach, the ensemble classifier performs better than the single classifier.

Table V provides a comparison of classification accuracy between the proposed approach and two commonly used ensemble classifier creation methods namely *bagging* and *boosting*. In all six cases the proposed approach performs better than the other ensemble classifier creation methods. Overall the proposed approach performs 2.83% better than bagging and 2.55% better than boosting. Detection of overlapping and thus difficult–to–classify patterns by clustering and then training of the weak classifiers on identical but disjoint training patterns makes the proposed ensemble classifier approach more diverse than the other approaches leading to better classification accuracy.

V. Conclusion

In this paper, we have presented and analysed a novel approach towards creating and training of an ensemble classifier. The proposed approach trains weak classifiers on different partitions of the data set obtained by explicit clustering. In order to obtain multiple decisions on a pattern, overlapping clusters are created at different levels using different clustering parameters and weak classifiers are trained on them. The decisions obtained on a test pattern at different levels are fused using majority voting. The proposed approach has been tested on six well known benchmark data sets. The results show that the use of overlapping clusters at different levels contribute to gain accurate decisions from complementary weak classifiers and classification accuracy increases at higher number of levels. The proposed approach has been compared with existing approaches and it has outperformed the commonly used ensemble classifier approaches. This is due to the fact that patterns belong to different clusters at different levels in the proposed approach and thus the weak classifiers are trained on complementary and difficult–to–classify subsets of the data leading to achievement of better diversity compared to other approaches. In future we aim to investigate the influence of data imbalance at non–atomic clusters on classification accuracy of the ensemble classifier.

Table IV: Comparative classification accuracy (%) between the proposed ensemble classifier and individual neural network.

Data Set	Single Classifier (Neural Network)	Proposed Ensemble Classifier
Breast Cancer	95.44	**97.52**
Ionosphere	83.80	**89.35**
Vowel	74.02	**98.17**
Sonar	71.88	**85.47**
Waveform	86.21	**86.95**
Wine	95.50	**99.16**

Table V: Comparative classification accuracy (%) between proposed and existing ensemble classifiers with majority voting fusion.

Data Set	Bagging	Boosting	Proposed Ensemble Classifier
Breast Cancer	96.12	95.80	**97.52**
Ionosphere	87.60	86.84	**89.35**
Vowel	89.90	97.67	**98.17**
Sonar	83.90	82.03	**85.47**
Waveform	85.95	83.70	**86.95**
Wine	97.89	97.20	**99.16**

References

[1] R. Polikar, “Ensemble based systems in decision making,” IEEE Circuits and Systems Magazine, vol. 6, no. 3, pp. 21–45, 2006.

[2] L. Breiman, “Bagging predictors,” Machine Learning, vol. 24, no. 2, pp. 123–140, 1996.

[3] L. Breiman, “Random Forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.

[4] L. Breiman, “Pasting small votes for classification in large databases and on-line,” Machine Learning, vol. 36, pp. 85–103, 1999.

[5] L. Chen and M. S. Kamel, “A generalized adaptive ensemble generation and aggregation approach for multiple classifiers systems,” Pattern Recognition, vol. 42, pp. 629–644, 2009.

[6] L. Nanni and A. Lumini, “Fuzzy bagging: a novel ensemble of classifiers,” Pattern Recognition, vol. 39, pp. 488–490, 2006.

[7] R. E. Schapire, “The strength of weak learnability,” Machine Learning,” vol. 5, no. 2, pp. 197–227, 1990.

[8] H. Drucker, C. Cortes, L. D. Jackel, Y. LeCun, and V. Vapnik, “Boosting and other ensemble methods,” Neural Computation, vol. 6, no. 6, pp. 1289–1301, 1994.

[9] Y. Freund and R. E. Schapire, “Decision-theoretic generalization of on-line learning and an application to boosting,” Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119–139, 1997.

[10] N. G. Pedrajas, “Constructing ensembles of classifiers by means of weighted instance selection,” IEEE Transaction on Neural Networks, vol. 20, no. 2, pp. 258–277, 2009.

[11] J. J. Rodriguez and J. Maudes, “Boosting recombined weak classifiers,” Pattern Recognition Letters, vol. 29, pp. 1049–1059, 2008.

[12] D. Parikh and R. Polikar, “Ensemble based incrimental learning approach to data fusion,” IEEE Transaction on Systeams, Man, and Cybernetics, vol. 37, no. 2, pp. 437–450, 2007.

[13] M. D. Muhlbaier, A. Topalis, and R. Polikar, “Learn++.NC: Combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes,”

IEEE Transaction on Neural Networks, vol. 20, no. 1, pp. 152–168, 2009.

[14] A. H. R. Ko, R. Sabourin, A. de S. Britto, and L. Oliveira, “ Pairwise fusion matrix for combining classifiers,” Pattern Recognition, vol. 40, pp. 2198–2210, 2007.

[15] UCI Machine Learning Database, http://archive.ics.uci.edu/ml/, accessed on 6th October 2009.

[16] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA Data Mining Software: An Update,” SIGKDD Explorations, vol. 11, no. 1, 2009.