ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Computers and Electrical Engineering 00 (2018) 1–14

Computers and Electrical Engineering

A comparative study of clustering ensemble algorithms

Xiuge Wu[a], Tinghuai Ma[a,b,*], Jie Cao[c], Yuan Tian[d], Mznah Al-Rodhaan[d]

[a]*School of Computer & Software, Nanjing University of information science & Technology, Jiangsu, Nanjing 210044, China*
[b]*CICAEET, Jiangsu Engineering Center of Network Monitoring, Nanjing University of information science & Technology, Nanjing 210044,China*
[c]*School of Economics & Management,Nanjing University of Information Science & Technology, Nanjing 210044, China*
[d]*Computer Science Department, College of Computer and Information Sciences, KingSaud University, Riyadh 11362, Saudi Arabia*

Abstract

Since clustering ensemble was proposed, it has rapidly attracted much attention. This paper makes an overview of recent researches on clustering ensemble about generative mechanism, selective clustering ensemble, consensus function and application. Twelve clustering ensemble algorithms are described and compared to choose a basic one. The experiment shows that using k-means with different initializations as generative mechanism and average-linkage agglomerative clustering as consensus function is the best one. As ensemble size increases, the performance of clustering ensemble improves. The basic clustering ensemble algorithm with suitable ensemble size is compared with clustering algorithms and the experiment shows that clustering ensemble is better than clustering. The influence of diversity on clustering ensemble is instructive to selecting members. The experiment shows that selecting members in high quality and big diversity for low-dimensional data sets, and selecting members in high quality and median diversity for high-dimensional data sets are better than traditional clustering ensemble.

Keywords: clustering ensemble, generative mechanism, consensus function, ensemble member, diversity, ensemble size

1. Introduction

With rapid progress of clustering technology, clustering analysis plays an important role in various fields, such as pattern recognition, image processing, business intelligence, document clustering, market research, data analysis and customer recommendation. It is difficult to find one clustering algorithm that can be applied to all data sets, so various clustering algorithms are improved and different clustering algorithms are proposed. For this problem, authors in [1] proposed the concept of clustering ensemble in 2003. Specifically, the definition of clustering ensemble is as follows: There is a data set $X = \{x_1, x_2, \ldots, x_n\}$ that has n data. Then, M clustering algorithms are used to cluster X and generate M partitions. The ensemble member set $P = \{P_1, P_2, \ldots, P_M\}$ is formed with these partitions and $P_m(m = 1, 2, \ldots, M)$ is the clustering partition obtained by the mth clustering algorithm. Subsequently, consensus function Γ will combine these ensemble members and get the final partition P^*. The intuitive illustration of clustering ensemble is shown in Fig. 1.

Clustering ensemble combines different clustering partitions about data set into a final one. The result of clustering ensemble is superior to single clustering algorithm. Single clustering algorithm has its own weakness, so it leads to one algorithm being only suitable for a specific data set. Clustering ensemble combines these clustering algorithms to

[*]Corresponding author: thma@nuist.edu.cn

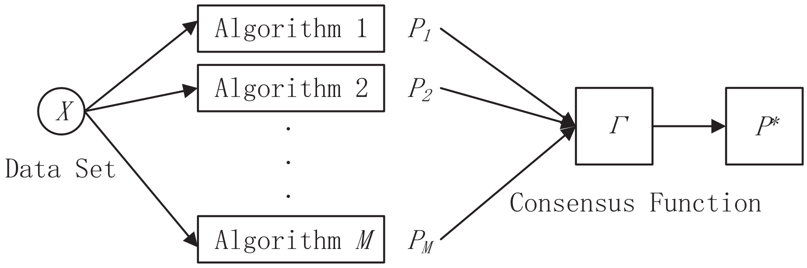


Fig. 1. Framework of clustering ensemble.

avoid the shortcoming of single clustering algorithm. It fits more data sets than clustering and it is also robust against noise and outliers [2].

In order to make a comprehensive research on clustering ensemble and choose a basic algorithm for our further researches, we introduce twelve clustering ensemble algorithms. The twelve algorithms are composed of three generative mechanisms and four consensus functions. Then, the comparative experiment of these algorithms finds the best one as the basic algorithm for further researches. The basic algorithm generates ensemble members using k-means with different initializations and combines members using average-linkage agglomerative clustering. Next, the influence of ensemble size on clustering ensemble is analysed to find an appropriate ensemble size. Then the basic algorithm with suitable ensemble size is compared with standard clustering algorithms. In addition, the relation of diversity and performance of clustering ensemble is explored to guide the selection of ensemble members. Finally, the selective clustering ensemble based on quality and diversity is compared with traditional clustering ensemble.

The rest of this paper is organized as follows. Section 2 reviews the recent researches on clustering ensemble. Three generative mechanisms and four consensus functions are described in Section 3. Section 4 compares twelve clustering ensemble algorithms on six data sets, analyzes the influence of ensemble size and diversity, and compares clustering ensemble with standard clustering algorithms and selective clustering ensemble. This paper is concluded in Section 5 with discussion about future works of clustering ensemble.

2. Literature review on clustering ensemble

There are two main phases in clustering ensemble. The first stage is producing ensemble members while the second stage is combining these ensemble members to get the final partition. As indicated in Fig. 2, the left side shows different generative mechanisms and the right side displays different consensus functions. By selecting different clustering algorithms, setting different initializations for same clustering algorithm, using sampling data or using feature subsets, we can produce different ensemble members. Whereas the consensus functions include voting approach, hierarchical clustering, graph method, information theory and mixture model.

Accordingly, the recent researches mainly focus on four aspects. (1) Generative mechanism: the approach to get the ensemble members [3–12]. (2) Selective clustering ensemble: selecting effective ensemble members before consensus function [13–18]. (3) Consensus function: the method of combining ensemble members [19–23]. (4) Application: the practical applications of clustering ensemble [24–27].

2.1. Generative mechanism

Adopting different clustering algorithms to generate ensemble members is one of the common generative mechanisms. Authors in [1] propose to apply different clustering algorithms on the same data set. In [3], authors use self-organizing maps and k-means, the two well-known clustering algorithms in neural network and statistical field, to generate ensemble members.

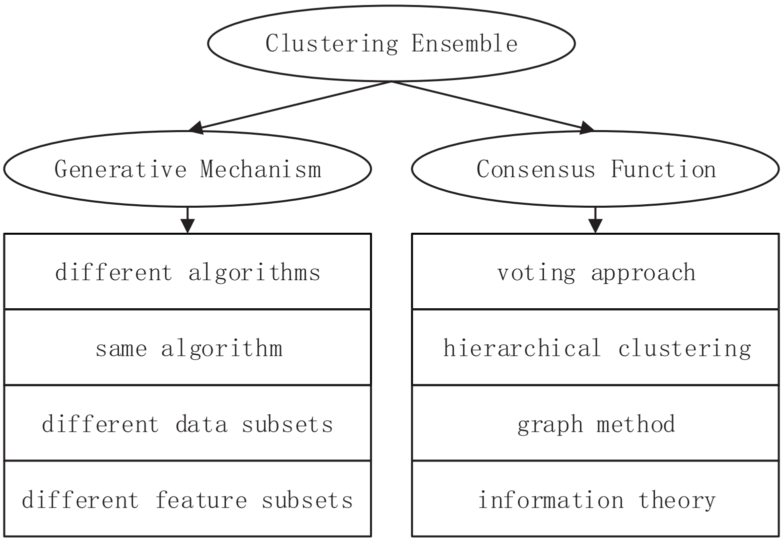


Fig. 2. Clustering ensemble algorithms classification from generative mechanism and consensus function.

Setting different initial parameters is the main idea of using the same clustering algorithm. In [5], authors repeat k-means algorithm with different k values. Authors in [6] repeat k-means with the same k value but they randomly choose initial cluster centers each time. In [7], spectral clustering with random scaling parameter is used as generative mechanism to generate different ensemble members.

Using different data subsets is suitable for big data. Authors in [8] use the bagging technique to generate data subsets and propose the structure ensemble approach based on probabilistic bagging. In [10] a new hierarchical clustering ensemble method based on boosting is introduced, and this method uses several boosting iterations to create ensemble members.

For high-dimensional data sets, it is suitable to generate members using different feature subsets. Authors in [11] propose a new clustering ensemble approach based on fuzzy c-means clustering with random projection. In [12], authors compare random projection, principal component analysis and random sampling the three dimensionality reduction methods. The results of clustering ensemble are always affected by noise and outliers. In [2], authors use a feature selection algorithm to remove the noise attributes in data set.

2.2. *Selective clustering ensemble*

A clustering ensemble algorithm typically produces many ensemble members. However, it is not good to combine all available members. So it is necessary to select suitable ensemble members. Selective clustering ensemble is an algorithm that combines partial ensemble members rather than combining all ensemble members. In [13], authors investigate the diversity among ensemble members. They find that combining members in big diversity is better than combining members in small diversity even if the latter contains more accurate members. Nevertheless, authors in [14] carry out a further research. They find that the relation between diversity and quality of members is not linear. And selecting members in median diversity is always better than selecting members in big diversity.

In [15], authors regard selective clustering ensemble as the two-objective optimization problem of quality and diversity, so they propose an algorithm based on random sampling. This algorithm estimates the qualities and diversities of ensemble members through resampling technique and selects members to build new ensemble member set. Authors in [5] investigate several methods to evaluate and select ensemble members based on relative clustering validity indexes. These indexes calculate the relationship between cluster and partition, and select ensemble members that are of high quality to participate in clustering ensemble. They combine these relative indexes and create a final evaluation criterion for selecting the high quality members to participate in rather than combining all ensemble members.

Authors in [16] are inspired by the concept of attribute importance in rough set theory. They regard all members of clustering ensemble as features of data set, and transformed selective clustering ensemble into unsupervised feature selection. First, the selective clustering ensemble based on rough set theory calculates the importance of each attribute relative to all attributes according to information entropy. Then it selects members that correspond to important attributes. In [18], authors present an incremental semi-supervised clustering ensemble algorithm which removes redundant ensemble members according to two cost functions. The first cost function considers not only the similarity between two subspaces but also the cost of ensemble members, while the second cost function calculates the cost that combines the increasing selected members into the final partition.

2.3. *Consensus function*

The method based on co-association matrix was proposed in 2001 before the appearance of clustering ensemble [19]. In [21], authors use voting spatial clustering ensembles based on co-association matrix to protect privacy. The co-association matrix calculates the similarities among data according to the frequencies of data in same cluster. In [5], authors propose evidence accumulation which partitions data x_i and x_j into same cluster once their votes are more than the others. To combine the ensemble members, authors in [6] propose the extended evidence accumulation clustering, a new consensus function based on co-association matrix.

Graph method is a popular consensus function. In [18], authors regard the problem of combining members as graph partitioning and deal it with normalized cut algorithm. Authors in [22] use spectral clustering algorithm on similarity matrix to produce the final partition. In addition, authors in [1] propose three hyper graph methods: cluster-based similarity partitioning algorithm, hyper-graph partitioning algorithm and meta-clustering algorithm. The three algorithms transform data into hyper graph based on clustering members, and the hyper edge between data means the data belong to the same cluster.

Information theory and mixture model are two new consensus functions. In [2], authors adopt a probabilistic model and a finite mixture of multinomial distributions in cluster space as consensus function. Authors in [10] combine the ensemble members through an information theoretic approach. In [23], authors propose a clustering ensemble algorithm based on Dempster-Shafer evidence theory. This algorithm uses the neighbors to describe the data and takes the surrounding information of cluster structure about data into consideration. First, they find the neighbors of each data and generate its label probability outputs in each member. Second, these label probability outputs are integrated based on Dempster-Shafer theory to produce the final result.

2.4. *Application*

The application of clustering ensemble is extensive, especially in image. This is mainly because some consensus functions are based on graph. In [22] the clustering ensemble algorithm is applied to image segmentation and it outperforms most existing image segmentation methods. Authors in [24] introduce clustering ensemble algorithm into visual object categorization and face image grouping with multiple feature representations.

Clustering ensemble supplies significant contributions to medical research. Accuracy is extremely important in medical science. Authors in [18] propose five clustering ensemble algorithms to discover the types of cancer: adoptive clustering ensemble algorithm, clustering ensemble algorithm based on knowledge, clustering ensemble algorithm based on neural gas, hybrid fuzzy clustering ensemble framework and clustering ensemble algorithm based on fuzzy theory. In [25], authors design a new medical system which combines clustering ensemble with text summarization for comprehensive gene expression data analysis.

Clustering ensemble is widely used in cloud classification. In 2012, authors in [27] propose a high resolution satellite precipitation estimation algorithm (HSPE) which uses the link-based clustering ensemble method (LCE) to cluster cloud patches. HSPE includes four steps: split infrared cloud images into patches, cloud patch feature extraction, cluster cloud patches using LCE, dynamic application of brightness temperature and rain rate relationships derived using satellite observations. The addition of clustering ensemble makes HSPE outperform LCE with self organizing map in rainfall estimate.

3. Common clustering ensemble algorithms

In this paper, we compare twelve common clustering ensemble algorithms composed of three generative mechanisms and four consensus functions to choose a basic one for further researches on clustering ensemble. The three generative mechanisms are same algorithm with different initializations, using different data subsets and using different feature subsets in Fig. 2. While the four consensus functions are voting approach and three agglomerative hierarchical clustering algorithms based on co-association matrix in Fig. 2.

3.1. Generative mechanisms

3.1.1. Same algorithm with different initializations

Using one algorithm but setting different initializations is one of the most common generative mechanisms in clustering ensemble. And k-means is usually used in this method because it is easy to implement and it has low complexity [6]. In this paper, we use k-means with different initializations as the first generative mechanism. The progress of using k-means as generative mechanism is described in Algorithm 1.

Algorithm 1 Generative mechanism: k-means with different initializations

```
Input: $k$, number of clusters in each ensemble member; $X$, data set with $n$ data; $M$, number of ensemble members
Output: $M$ ensemble members
set $m = 0$;
while $m < M$ do
    randomly select $k$ initial cluster centers;
    repeat
        (re)assign each data into the cluster according to Eq. (1);
        update the cluster centers according to Eq. (2);
    until there is no change with clusters or Eq. (3) is minimal;
    $m = m + 1$;
end while
```

Generally speaking, the generative mechanism of k-means with different initializations is running k-means M times with fixed k value and different initial cluster centers. First, it randomly chooses k data from X as initial cluster centers. Then, it assigns data into the cluster where its nearest cluster center is. This assignment is based on the Euclidean distance between data and cluster center in Eq. (1).

$$dist(x_j, c_i) = \sqrt{(x_{j1} - c_{i1})^2 + (x_{j2} - c_{i2})^2 + \cdots + (x_{js} - c_{is})^2} \tag{1}$$

where $x_j = (x_{j1}, x_{j2}, \ldots, x_{js})$ is a piece of data in X, $c_i = (c_{i1}, c_{i2}, \ldots, c_{is})$ is cluster center and their dimensions are s. Since the data are reassigned to other clusters, the cluster centers will change. So it is necessary to calculate the new cluster center in each cluster according to Eq. (2).

$$c_i = \frac{1}{\mid C_i \mid} \sum_{x_j \in C_i} x_j \tag{2}$$

where $\mid C_i \mid$ is the total number of data in cluster C_i. Repeating above reassignment and recalculation until the cluster centers do not change or the objective function in Eq. (3) is minimized.

$$E = \sum_{i=1}^{k} \sum_{x_j \in C_i} dist(x_j, c_i) \tag{3}$$

Repeating the above operations M times and getting M ensemble members.

3.1.2. Different data subsets

Using different data subsets as generative mechanism is suitable for big data. In this paper, we adopt resampling method without replacement to generate the subsets of original data [10]. Then, using k-means on these data subsets to generate multiple ensemble members. The generative mechanism of using different data subsets is summarized in Algorithm 2.

Algorithm 2 Generative mechanism: k-means with different data subsets

```
Input: $k$, number of clusters in each ensemble member; $X$, data set with $n$ data; $M$, number of ensemble members
Output: $M$ ensemble members
set $m = 0$ and calculate $n^*$ according to Eq. (4);
while $m < M$ do
    loop sampling without replacement until $n^*$ data are randomly selected from $X$;
    randomly select $k$ initial cluster centers from the selected $n^*$ data;
    repeat
        (re)assign each data into the cluster according to Eq. (1);
        update the cluster centers according to Eq. (2);
    until there is no change with clusters or Eq. (3) is minimal;
    $m = m + 1$;
end while
```

The generative mechanism of using different data subsets is running k-means M times with different initial cluster centers on resampling data. First, it calculates the number of resampling data based on the Law of Large Numbers and the Central Limit Theorems. The formula is described in Eq. (4).

$$n^* = \frac{n \times t^2 \times \delta^2}{\Delta_{\overline{x}}^2 \times n + t^2 \times \delta^2} \tag{4}$$

where n is the total number of data, $t = 1.96$ is the probability degree because the confidence coefficient we used in our experiments is 0.95, δ is the standard deviation, and $\Delta_{\overline{x}} = 0.1\delta$ is error limitation. Then, it randomly samples n^* data from X without replacement. Next, k-means randomly chooses k data from the above n^* data as initial cluster centers. Then, it reassigns data into clusters according to Eq. (1) and recalculates the cluster centers according to Eq. (2). Reassigning and recalculating until the cluster centers do not change or Eq. (3) is minimized. Repeating above operations until M ensemble members are gotten.

3.1.3. Different feature subsets

It is suitable for high-dimensional data sets to use different feature subsets as generative mechanism. In this paper, we adopt ReliefF algorithm [12] for dimensionality reduction and generate different data sets with selected features. Then k-means runs on these new data sets that are composed of the selected feature subsets to generate multiple ensemble members. The generative mechanism of using different feature subsets is presented in Algorithm 3.

The generative mechanism of using different feature subsets mainly includes two steps: projecting data into a random subspace of lower dimension and clustering new data sets with k-means. First, it finds the neighbors in same cluster with the data and denotes them as H. Second, it finds the neighbors in different cluster from the data and denotes them as M. Then, it calculates the weight of feature on each data according to Eq. (5).

$$W(A_j) = W(A_j) - \sum_{i=1}^{r} \frac{dif(A, x, H_i)}{mr} + \sum_{C \in class(x_j)} \frac{\frac{p(C)}{1-p(class(x_j))} \sum_{i=1}^{r} dif(A, x, M_i(C))}{mr} \tag{5}$$

where x_j is a data in X, $W(A)$ is the weight of feature A on data x_j, m is the number of resampling frequency, r is the number of neighbors, H_i is the ith data in H, $M_j(C)$ denotes the jth data of cluster C in M, $class(x_j)$ is the cluster label of x_j, $p(C)$ denotes the probability of cluster C, $dif(A, x_i, x_j)$ indicates the difference of feature A on data x_i and x_j and it is shown in Eq. (6).

$$dif(A, x_i, x_j) = \begin{cases} \frac{|x_i[A] - x_j[A]|}{max(A) - min(A)}, & \text{if } A \text{ is continuous} \\ 0, & \text{if } A \text{ is discrete and } x_i[A] = x_j[A] \\ 1, & \text{if } A \text{ is discrete and } x_i[A] \neq x_j[A] \end{cases} \tag{6}$$

Algorithm 3 Generative mechanism: k-means with different feature subsets

```
Input: $k$, number of clusters in each ensemble member; $X$, data set with $n$ data; $M$, number of ensemble members; $\delta$, the threshold of feature weight
Output: $M$ ensemble members
while $m < M$ do
    set all feature weights as 0;
    repeat
        find $H$ and $M$ of each data and calculate the weight of feature on each data according to Eq. (5);
        calculate the average weight of feature according to Eq. (7);
        if $W(A) \geq \delta$ then
            retain feature $A$;
        else
            delete feature $A$;
        end if
    until all features are compared with $\delta$;
    generate new data set with retained features and randomly select $k$ initial cluster centers;
    repeat
        (re)assign each data into the cluster according to Eq. (1);
        update the cluster centers according to Eq. (2);
    until there is no change with clusters or Eq. (3) is minimal;
    $m = m + 1$;
end while
```

where $x_i[A]$ and $x_j[A]$ are the values of feature A on x_i and x_j respectively, $max(A)$ and $min(A)$ are respectively the maximum value and minimum value of feature A on X. Next, it calculates the average weight of each feature according to Eq. (7).

$$W(A) = \frac{1}{p}\sum_{i=1}^{p} W(A_j) \tag{7}$$

where $W(A)$ is the average weight of feature A. And if $W(A) \geq \delta$, retain feature A. Otherwise, delete feature A. Then, it produces new data set using the retained features and randomly selects k data from the new data set as initial cluster centers. Reassigning data into clusters according to Eq. (1) and recalculating the cluster centers according to Eq. (2) until the cluster centers do not change or Eq. (3) is minimized. Repeating above operations until M ensemble members are generated.

3.2. Consensus functions

Voting approach, the first consensus function we used in this paper, is based on co-association matrix [19]. Co-association matrix is a symmetrical matrix that reflects the relations of data in M different ensemble members. The similarity of data x_i and x_j in co-association matrix is calculated as Eq. (8).

$$CO_{ij} = \frac{c_{ij}}{M} \tag{8}$$

where c_{ij} counts the number of data x_i and x_j appear in same cluster. Based on the definition of co-association matrix, $CO_{ij} = CO_{ji}$. Voting approach is the earliest proposed consensus function and has been used frequently. After calculating co-association matrix, voting approach compares the values in it with the defined threshold θ. If $CO_{ij} > \theta$, data x_i and x_j are partitioned into one cluster. Otherwise, they are in different clusters. In our experiments, we set θ as 0.5 because the previous researches usually set θ as 0.5.

The three agglomerative clustering consensus functions are based on graph [22]. The vertexes of graph are data. At first, the vertexes are independent of each other and there are no edges among the vertexes. The three agglomerative algorithms combine vertexes into clusters gradually according to cluster proximity. The first agglomerative algorithm is single-linkage consensus function which defines cluster proximity as the closest distance of data in two clusters.

The second agglomerative algorithm is complete-linkage consensus function which takes the farthest distance of data in two clusters as cluster proximity. The third agglomerative algorithm is average-linkage consensus function which defines cluster proximity as the average distance among data in two clusters. The three cluster proximities are presented in Eq. (9).

$$PM(C_m, C_n) = \begin{cases} \arg\min_{x_i \in C_m, x_j \in C_n} dist(x_i, x_j), & \text{single-linkage} \\ \arg\max_{x_i \in C_m, x_j \in C_n} dist(x_i, x_j), & \text{complete-linkage} \\ \frac{\sum_{x_i \in C_m} \sum_{x_j \in C_n} dist(x_i, x_j)}{|C_m||C_n|}, & \text{average-linkage} \end{cases} \tag{9}$$

where $dist(x_i, x_j)$ is the Euclidean distance between data x_i and x_j, $\mid C_m \mid$ is the number of data in cluster C_m, $\mid C_n \mid$ is the number of data in cluster C_n, $PM(C_m, C_n)$ is the cluster proximity between cluster C_m and C_n.

4. Experiments

In this paper, we design five sets of experiments to analyze clustering ensemble. First, we compare twelve common clustering ensemble algorithms to choose a basic one for further experiments. Second, we explore the relationship between algorithm performance and ensemble size. Third, the basic clustering ensemble with suitable ensemble size is compared with three standard clustering algorithms to prove its efficiency. Fourth, we study the relation between diversity and performance of clustering ensemble to guide selective clustering ensemble. At last, selective clustering ensemble based on quality and diversity is compared with traditional clustering ensemble. Our experiments are based on six University of California Irvine Machine Learning Repository data sets. Table 1 shows the detailed information about data sets.

Table 1: Summary of data sets where n is amount of data, d is number of features and k is number of classes.

data set	n	d	k
Image Segmentation (IS)	2310	19	7
Ionosphere	351	34	2
Iris	150	4	3
Page Blocks Classification (PBC)	5473	10	5
Statlog of Vehicle Silhouettes (SVS)	846	18	4
Wine	178	13	3

In clustering ensemble, the diversity between ensemble members can be measured by (Adjusted) Rand Index, Jaccard Index and (Normalized) Mutual Information [5]. And these indicators have same meaning and similar trend. In this paper, we evaluate diversity in ensemble members with Jaccard Index (JI) which is defined as Eq. (10).

$$JI(P_i, P_j) = \frac{n_{11}}{n_{11} + n_{01} + n_{10}} \tag{10}$$

where n_{11} denotes the number of data pairs which are not only in the same cluster in P_i, but also in the same cluster in P_j. And n_{01} is the number of data pairs in different clusters in P_i but in the same cluster in P_j. On the contrary, n_{10} denotes the number of data pairs in the same cluster in P_i but in different clusters in P_j. In general, n_{11} counts the consistency among P_i and P_j while n_{01} and n_{10} count the inconsistency of P_i and P_j. If the partition of P_i is the same as P_j, then $JI(P_i, P_j)$ takes its maximum value of 1.

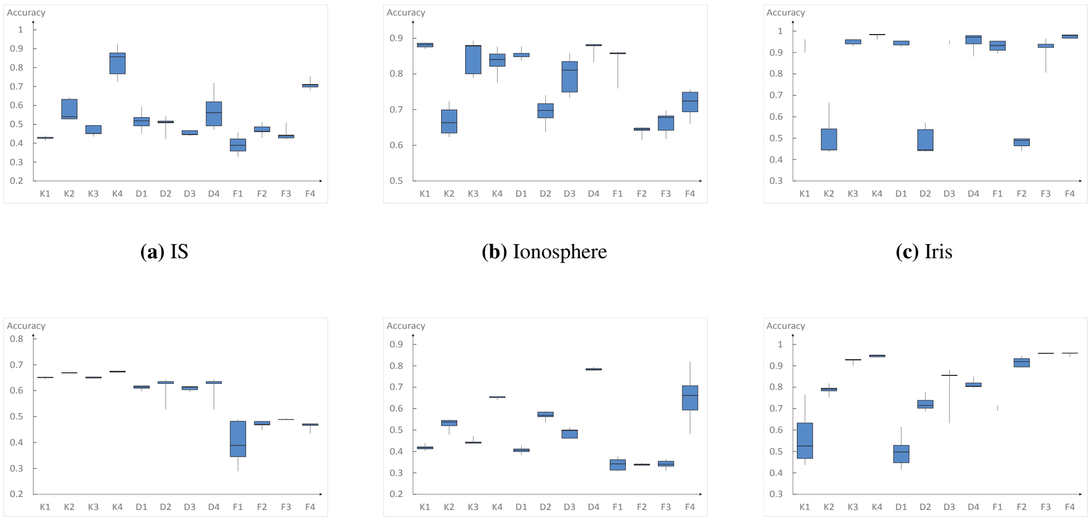
In most cases, the real class labels are adopted as standard for measuring the performance of clustering ensemble. Error Rate, Accuracy and (Normalized) Mutual Information are used to evaluate the performance of clustering ensemble algorithm [5]. In this paper, we use Accuracy estimating the algorithm performance. Accuracy counts the probability of data in right partition and is defined as Eq. (11).

$$Accuracy = \frac{\sum_{i=1}^{n} e_i}{n} \tag{11}$$

If data x_i has same partition in our result with the real partition, $e_i = 1$. Otherwise, $e_i = 0$. The bigger the Accuracy, the greater the performance of clustering ensemble.

4.1. Comparison of common clustering ensemble algorithms

In order to choose a basic clustering ensemble algorithm for further researches on selective clustering ensemble, we compare twelve clustering ensemble algorithms. The twelve clustering ensemble algorithms are composed of three generative mechanisms and four consensus functions described in Section 3. The three generative mechanisms are k-means with different initializations, method based on different data subsets and method based on different feature subsets. The four consensus functions are voting approach and three agglomerative clustering algorithms respectively with single-linkage (SL), complete-linkage (CL) and average-linkage (AL). The ensemble size in each algorithm is 5, 8, 13, 20 and 30 respectively. Each clustering ensemble algorithm with different ensemble sizes runs 20 times. The statistical results about twelve clustering ensemble algorithms in 100 runs are shown in Fig. 3 in the form of box-plot. The box-plot in Fig. 3 not only shows the best result, the average result and the worst result, but also shows the distribution of all results.



(a) IS **(b)** Ionosphere **(c)** Iris

(d) PBC **(e)** SVS **(f)** Wine

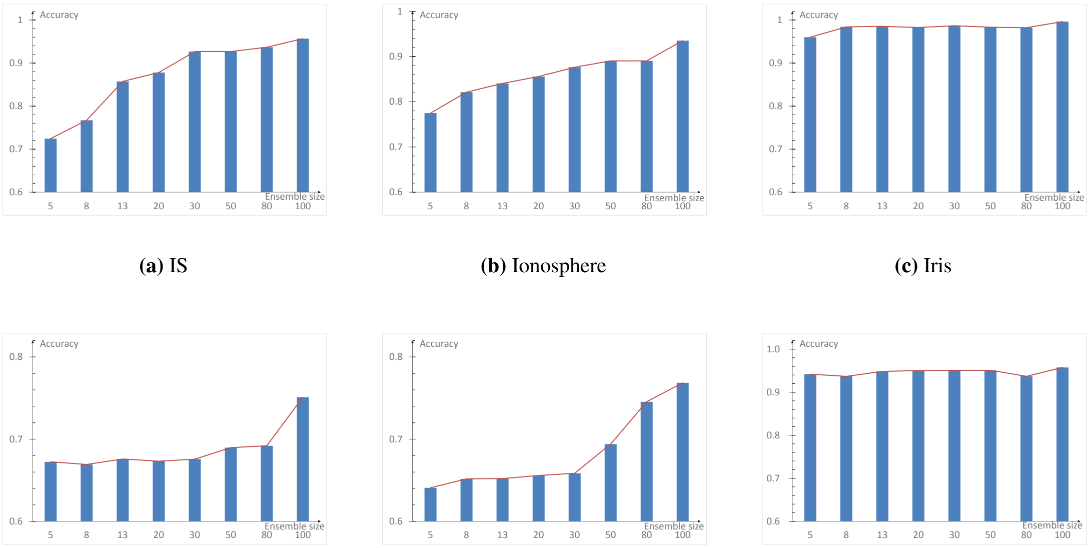
Fig. 3. All results of 12 clustering ensemble algorithms on 6 data sets in 100 runs. Accuracy represents the algorithm performance. K1∼K4 respectively denote k-means generative mechanism with voting approach, SL, CL and AL. D1∼D4 are combinations of using different data subsets with voting approach, SL, CL and AL. F1∼F4 combine using different feature subsets with voting approach, SL, CL and AL respectively.

From Fig. 3 we can intuitively see the following information. (1) Compared with D1 and F1, K1 performs well on data set (b)∼(f). Compared with D2 and F2, K2 achieves the best result on data set (a), (c), (d). Compared with D3 and F3, K3 performs well on data set (a)∼(d). Similarly, K4 performs better than D4 and F4 on data set (a), (c), (d), (f). According to these phenomena, we can get the conclusion that k-means with different initializations is the best generative mechanism. Using data subsets is suitable for big data sets and using feature subsets fits data sets that have many attributes. (2) Compared with K1∼K3, K4 achieves the best result on data set (a), (c)∼(f). Compared with D1∼D3, D4 performs well on data set (a)∼(e). F4 performs better than F1∼F3 on data set (a), (c), (e), (f). AL is the best consensus function. Voting approach considers votes of all members and it is easily affected by the quality of ensemble members. The cluster proximity in SL is the shortest distance among data in two clusters and it is sensitive to noise and outliers. The cluster proximity in CL is the farthest distance among data in two clusters and it breaks large clusters. Compared with SL and CL, AL calculates the average distance of data in two clusters as cluster proximity and it avoids the above weaknesses. (3) The clustering ensemble algorithm K4 that uses k-means as generative mechanism and AL agglomerative clustering as consensus function performs well on most data sets. And our other researches about clustering ensemble are based on algorithm K4.

In addition, we find that the Accuracy of some algorithms is less than 60%. This is resulted by ensemble size and consensus function. Once the ensemble size is small, each ensemble member will have a big influence on the final result. And if the qualities of ensemble members are poor, the result of clustering ensemble is affected accordingly. So, it is necessary to increase the ensemble size to reduce the influence of members that are of poor quality. Besides, a good consensus function is essential. Although AL is the best consensus function in this experiment, there are still many consensus functions waiting for us to study. Although clustering ensemble is widely used in image, the results of most clustering ensemble algorithms on IS data set are not as good as we excepted. In addition to the impact of ensemble size, this is because the best consensus function in image processing is Normalized cut algorithm.

4.2. Influence of ensemble size on performance of clustering ensemble

In this experiment, we analyze the influence of ensemble size on performance of clustering ensemble. According to the experiment results in Section 4.1, the clustering ensemble algorithm we used is K4 that generates ensemble members with k-means and combines ensemble members with average-linkage agglomerative clustering. We set the number of ensemble members as 5, 8, 13, 20, 30, 50, 80 and 100. The average Accuracy of K4 with different ensemble sizes in 20 runs are calculated to denote the performance of clustering ensemble. The results are shown in Fig. 4.



(a) IS

(b) Ionosphere

(c) Iris

(d) PBC

(e) SVS

(f) Wine

Fig. 4. Influence of ensemble size on clustering ensemble. The ensemble sizes are 5, 8, 13, 20, 30, 50, 80 and 100. Accuracy denotes the performance of clustering ensemble. The line charts and bar charts are all expressions of performance with different ensemble sizes.

From Fig. 4 we can see that the algorithm achieves the best when ensemble size is 100 on all data sets. And on most data sets, the performance of clustering ensemble increases with the ensemble size increases. With the increment of ensemble size, there are more ensemble members than previous and the influence of members that are of low quality becomes smaller. In addition, we also see that the performance of the clustering ensemble on Iris and Wine data set is superior to it on IS and Ionosphere data sets. And the clustering ensemble algorithm is general on PBC and SCS data sets. Because Iris and Wine are in small sizes with few instances and few attributes, the performance of clustering ensemble algorithm reaches the best easily on these data sets. According to the conclusion in this experiment, the clustering ensemble algorithm improves gradually with the ensemble size increases. Generally speaking, the bigger the ensemble size, the better the performance of clustering ensemble.

4.3. Comparison of clustering ensemble algorithm and standard clustering algorithms

In this experiment, we compare clustering ensemble algorithm with standard clustering algorithms. The clustering ensemble algorithm we used is algorithm K4 in Section 4.1 and the ensemble size is 100. The standard clustering algorithms we used in this experiment are standard k-means clustering algorithm (k-means) [28], fuzzy c-means clustering algorithm (FCM) [29] and self organizing maps clustering algorithm (SOM) [30]. The number of clusters in k-means, FCM and SOM is the real class number of data sets. The learning gain coefficient in SOM is $a(t) = 1 - \frac{t}{n}$ where t is the current time and n is the number of data. The reaction neighbourhood in SOM is half of the network bandwidth. All algorithms run 20 times and the average Accuracy of each algorithm on six data sets is calculated and shown in Fig. 5.

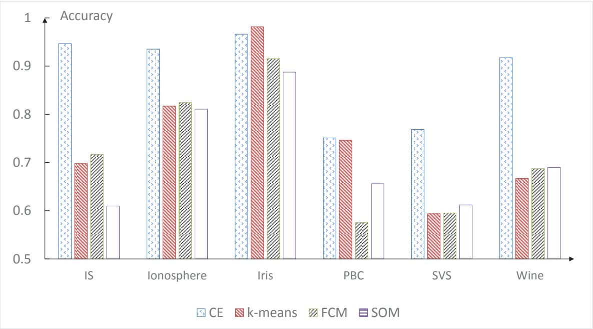


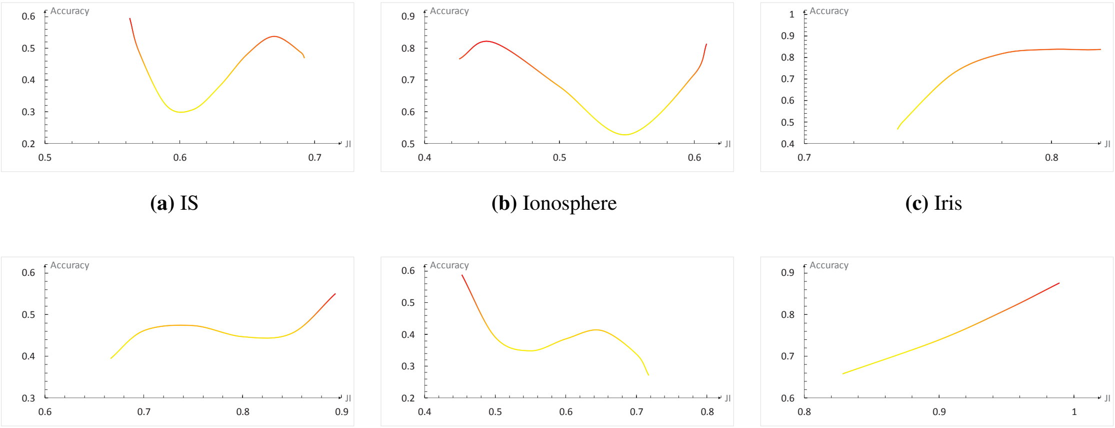
Fig. 5. The comparison of clustering ensemble algorithm and standard clustering algorithms. CE is clustering ensemble algorithm, k-means is standard k-means clustering algorithm, FCM is fuzzy c-means clustering algorithm and SOM is self organizing maps clustering algorithm. Accuracy represents the algorithm performance.

From Fig. 5 we can see that compared with standard clustering algorithms, CE is better than FCM and SOM on all data sets and it is superior to k-means on five data sets. The performance of clustering ensemble is better than standard clustering algorithms because clustering ensemble produces final result using all ensemble members that are generated by k-means. On PBC data set, the result of CE is similar to the result of k-means. And the result of k-means is better than the result of CE on Iris data set. This is because k-means performs well on low-dimension data sets. But, k-means is not as good as FCM and SOM on data sets that have more attributes. Compared with FCM, SOM is superior to FCM on SVS and Wine data sets, but it is worse than FCM on IS and Ionosphere data sets. This is because SOM partitions data as the human brain. As the dimension of the data increases, the complexity increases and the accuracy is reduced. It can be seen that there is no clustering algorithm that performs well on all data sets, and this is the reason for proposing clustering ensemble. Clustering ensemble combines the results of different clustering algorithms and it has better performance than single clustering algorithm.

4.4. *Influence of diversity among members on selective clustering ensemble*

Quality and diversity are two factors in selective clustering ensemble while selecting ensemble members. From research in [18] and above experiments, we know ensemble members that are of high quality have positive influence on the final result. This experiment analysed the influence of diversity among ensemble members on selective clustering ensemble. We record the results of twelve clustering ensemble algorithms with different ensemble sizes in 20 runs. In this experiment, the ensemble size is 5, 8, 13, 20, 30. According to the statistics on average diversity and Accuracy of members, the trend chart of diversity and quality is drawn in Fig. 6.

From (c), (d) and (f) in Fig. 6 we can see that the performance of algorithms on Iris, PBC and Wine data sets improves as diversity of ensemble members increases. It is contrary on IS, Ionosphere and SVS data sets. The relationship between diversity and performance is not linearly increased on these data sets. It is generally declining, even though it finally rises on the Ionosphere data set. Therefore, the median diversity among ensemble members can get better results than big diversity on these data sets. The reason for this phenomenon is that Iris, PBC and Wine data sets have little attributes. If the diversity among ensemble members is small on data sets that have few attributes, these members can not fully reflect the structure of original data set. And this discovery is useful in selective clustering ensemble. On low-dimensional data sets, selecting ensemble members in big diversity is helpful to the final result. While on high-dimensional data sets, it is better to select members in median diversity.



(a) IS **(b)** Ionosphere **(c)** Iris

(d) PBC **(e)** SVS **(f)** Wine

Fig. 6. Relation between diversity among members and performance of clustering ensemble. JI denotes the diversity among the ensemble members. Accuracy evaluates the performance of clustering ensemble. The red part of line charts shows the results in high performance and the yellow part shows those in low performance.

4.5. Comparison of clustering ensemble and selective clustering ensemble

In this experiment, we compare the clustering ensemble algorithm with selective clustering ensemble algorithm on six data sets. The selective clustering ensemble selects ensemble members based on the conclusion in Section 4.4. It selects members in high quality and big diversity on Iris, PBC and Wine data sets. On IS, Ionosphere and SVS data sets, it selects ensemble members in high quality and median diversity. The generative mechanism in these algorithms is k-means and the consensus function is average-linkage agglomerative clustering. The ensemble size in clustering ensemble is 100. And the number of selected members in selective clustering ensemble is respectively 25, 50, and 75. Each algorithm runs 20 times and the average Accuracy of the results are calculated and shown in Fig. 7.

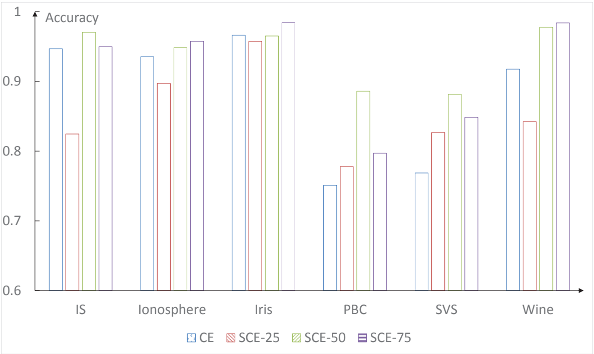


Fig. 7. The comparison of clustering ensemble and selective clustering ensemble. CE is the clustering ensemble algorithm, SCE-25 is selective clustering ensemble that selects 25 ensemble members, SCE-50 selects 50 ensemble members and SCE-75 selects 75 ensemble members from 100 ensemble members. Accuracy represents the algorithm performance.

From Fig. 7 we can see that SCE-50 and SCE-75 are superior to CE and SCE-25. It proves that selective clustering ensemble is generally better than traditional clustering ensemble. Some ensemble members have positive influence on the final result while some ensemble members have negative effect. Selecting the ensemble members that have positive influence can avoid the influence of members that have negative influence. However, SCE-25 is not as good as CE on most data sets. The reason for this result is that the number of selected members is small compared with ensemble size. And it can not fully reflect the real partition of data set. Therefore, selective clustering ensemble is better than traditional clustering ensemble that combines all ensemble members. And the suitable number of selected ensemble members needs to be researched in depth.

5. Conclusion and future works

In this paper, we review previous researches on clustering ensemble and introduce three generative mechanisms and four consensus functions. These algorithms are compared on data sets to choose a basic one for further researches on clustering ensemble. The best one in twelve clustering ensemble algorithms is algorithm K4 that uses k-means with different initiations as generative mechanism and average-linkage agglomerative clustering as consensus function. Generally, the performance of clustering ensemble improves with ensemble size increases. Clustering ensemble combines the results of single clustering algorithm and its result is better than the single one. Quality and diversity of ensemble members are two important factors in selective clustering ensemble. The influence of diversity among ensemble members on clustering ensemble is different according to attributes of data sets. On low-dimensional data sets, selecting ensemble members in high quality and big diversity is better than combining all ensemble members. For high-dimensional data sets, selecting ensemble members in high quality and median diversity is better than traditional clustering ensemble.

Through the research, comparison and analysis of clustering ensemble, our future researches will focus on the following aspects. First, the number of selected members is important in selective clustering ensemble expect quality and diversity. Our further research is aimed at designing a selective clustering ensemble algorithm based on quality and diversity. And the number of selected members is automatically generated rather than artificially set. Second, consensus function is important in combining ensemble members. In future, we are devoted to finding and studying a better consensus function to realize a better performance of clustering ensemble. Third, clustering ensemble has been applied to many application scenarios, but it has not been widely used in every field. Extending the applications of clustering ensemble is an important research point.

6. Acknowledgments

This work was supported in part by National Science Foundation of China (No. 61572259, No. U1736105) and Special Public Sector Research Program of China (No. GYHY201506080) and was also supported by PAPD.

This research project was supported by a grant from the “Research Center of the Female Scientific and Medical Colleges”, Deanship of Scientific Research, King Saud University.

References

[1] Strehl A, Ghosh J. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research 2003;3:583-617.

[2] Topchy A, Jain AK, Punch W. A mixture model of clustering ensembles. Proc. SIAM Intl. Conf. on Data Mining 2004;379-390.

[3] Tsai CF, Hung C. Cluster ensembles in collaborative filtering recommendation. Applied Soft Computing 2012;12(4):1417-1425.

[4] Ma TH, Zhang YL, Cao J, Shen J, Tang ML, Tian Y, Al-Dhelaan A, Al-Rodhaan M. KDVEM: a k-degree anonymity with vertex and edge modification algorithm. Computing 2015;97(12):1165-1184.

[5] Naldi MC, Carvalho AC, Campello RJ. Cluster ensemble selection based on relative validity indexes. Data Mining & Knowledge Discovery 2013;27(2):259-289.

[6] Alizadeh H, Minaei-Bidgoli B, Parvin H. Cluster ensemble selection based on a new cluster stability measure. Intelligent Data Analysis 2014;18(3):389-408.

[7] Jia J, Xiao X, Liu B, Jiao L. Bagging-based spectral clustering ensemble selection. Pattern Recognition Letters 2011;32(10):1456-1467.

[8] Yu Z, You J, Wong HS, Han G. From cluster ensemble to structure ensemble. Information Sciences An International Journal 2012;198(3):81-99.

[9] Rong H, Ma TH, Tang ML, Cao J. A novel subgraph K$^+$-isomorphism method in social network based on graph similarity detection. Soft Computing 2018;22(8):2853-2601.

[10] Rashedi E, Mirzaei A. A hierarchical clusterer ensemble method based on boosting theory. Knowledge-Based Systems 2013;45(3):83-93.

[11] Ye M, Liu W, Wei J, Wei J, Hu X. Fuzzy c-Means and Cluster Ensemble with Random Projection for Big Data Clustering. Mathematical Problems in Engineering 2016;1-13.

[12] Fern XZ, Brodley CE. Cluster ensembles for high dimensional clustering: an empirical study. Corvallis Or Oregon State University Dept of Computer Science 2004;1-26.

[13] Kuncheva LI, Hadjitodorov ST. Using diversity in cluster ensembles. IEEE International Conference on Systems, Man and Cybernetics 2004;2:1214-1219.

[14] Hadjitodorov ST, Kuncheva LI, Todorova LP. Moderate diversity for better cluster ensembles. Information Fusion 2005;7(3):264-275.

[15] Hong Y, Kwong S, Wang H, Ren Q. Resampling-based selective clustering ensembles. Pattern Recognition Letters 2009;30(3):298-305.

[16] Wang X, Han D, Han C. Rough set based cluster ensemble selection. Information Fusion 2013;438-444.

[17] Ma TH, Wang Y, Tang ML, Cao, J, Tian, Y, Al-Dhelaan, A, Al-Rodhaan M. LED: A fast overlapping communities detection algorithm based on structural clustering. Neurocomputing 2016;207:488-500.

[18] Yu Z, Luo P, You J, Wong HS, Leung H, Wu S, Zhang J, Han G. Incremental Semi-Supervised Clustering Ensemble for High Dimensional Data Clustering. IEEE Transactions on Knowledge & Data Engineering 2016;28(3):701-714.

[19] Fred ALN. Finding Consistent Clusters in Data Partitions. Multiple Classifier Systems 2001;2096:309-318.

[20] Ma TH, Rong H, Ying CH, Tian Y, Al-Dhelaan A, Al-Rodhaan M. Detect structural-connected communities based on BSCHEF in C-DBLP. Concurrency & Computation Practice & Experience 2016;28(2):311-330.

[21] Anandhi RJ, Natarajan S. Privacy Protected Mining Using Heuristic Based Inherent Voting Spatial Cluster Ensembles.Springer 2014;236:1183-1193.

[22] Wang L, Zhang G. Cluster Ensemble Based Image Segmentation Algorithm. Eighth International Conference on Internet Computing for Science and Engineering 2016;10(4):68-73.

[23] Li F, Qian Y, Wang J, Liang J. Multigranulation information fusion: a Dempster-Shafer evidence theory-based clustering ensemble method. Information Sciences 2016;1:58-63.

[24] Tsai JT, Lin YY, Liao HYM. Per-Cluster Ensemble Kernel Learning for Multi-Modal Image Clustering With Group-Dependent Feature Selection. IEEE Transactions on Multimedia 2014;16(8):2229-2241.

[25] Hu X, Park EK, Zhang X. Microarray gene cluster identification and annotation through cluster ensemble and EM-based informative textual summarization. IEEE Transactions on Information Technology in Biomedicine 2009;13(5):832-840.

[26] Lv YH, Ma TH, Tang ML, Cao J, Tian Yuan, Al-Dhelaan A, Al-Rodhaan M. An efficient and scalable density-based clustering algorithm for datasets with complex structures. Neurocomputing 2016;171(C):9-22.

[27] Mahrooghy M, Younan NH, Anantharaj VG, Aanstoos J, Yarahmadian S. On the Use of a Cluster Ensemble Cloud Classification Technique in Satellite Precipitation Estimation. IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing 2012;5(5):1356-1363.

[28] Ahmadian S, Norouzi-Fard A, Svensson O, Ward J. Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms. Foundations of Computer Science 2017;61-72.

[29] Zhang L, Lu W, Liu X, Pedrycz W, Zhong C. Fuzzy C-Means clustering of incomplete data based on probabilistic information granules of missing values. Knowledge-Based Systems 2016;99(C):51-70.

[30] Mcelwee S, Cannady J. Improving the performance of self-organizing maps for intrusion detection. IEEE Journal of Southeastcon 2016;1-6.

Xiuge Wu received her Bachelor degree in Network Engineering from Nanjing University of Information Science & Technology (NUIST), China, in 2015. Currently, she is a candidate for the degree of Master of Computer Science and Engineering in NUIST. Her research interests include data mining, social network, etc.

Tinghuai Ma is a professor in Computer Sciences and Engineering at NUIST. He received his Bachelor (HUST, China, 1997), Master (HUST, China, 2000), PhD (Chinese Academy of Science, 2003) and was Post-doctoral associate (AJOU University, 2004). His research interests are data mining, Cloud Computing, ubiquitous computing, privacy preserving etc. He has published more than 100 journal/conference papers.

Jie Cao received his Ph.D. (Southeast University, 2005). He was an associate professor from 1999 to 2006. From 2006 to 2009, he was a Post-Doctoral Fellow at Academy of Mathematics and Systems Science, Chinese Academy of Science. From 2009, he is a professor in School of management and economic, NUIST. His research interests are system engineering, management science and technology.

Yuan Tian received her master and Ph.D degree from KyungHee University. She is currently working as Assistant Professor at College of Computer and Information Sciences, King Saud University (KSU). She is member of technical committees of several international conferences. In addition, she is an active reviewer of many international journals. Her research interests are privacy and security.

Mznah Al-Rodhaan received her BS in Computer Applications (KSU, 1999), MS in Computer Science (KSU, 2003) and Ph.D. in Computer Science (University of Glasgow, 2009). She is currently working as the Vice Chair of the Computer Science Department in KSU. Her current research interest includes: Mobile Ad Hoc Networks, Wireless Sensor Networks, Multimedia Sensor networks, Cognitive Networks, and Network Security.