This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2018.2865954, IEEE Transactions on Knowledge and Data Engineering

An Information-Theoretical Framework for Cluster Ensemble

Liang Bai, Jiye Liang, Hangyuan Du, Yike Guo

Abstract—Cluster ensemble is a very important tool that aggregates several base clusterings to generate a single output clustering with improved robustness and stability. However, the quality of the final clustering is often affected by uncertainties on the generation and integration of base clusterings. In this paper, we develop an information-theoretical framework which makes an effort to obtain a final clustering with high consensus on both the original data set and the base clustering set by minimizing the two uncertainties of cluster ensemble. In this framework, we provide a weighted consensus measure based on information entropy to evaluate the quality of a clustering, the similarity between clusters and the similarity between objects. Based on the measure, we propose three weighted cluster ensemble algorithms with different ensemble strategies in the framework, including the weighted feature consensus algorithm, the weighted relabeling consensus algorithm and the weighted pairwise-similarity consensus algorithm. In the experimental analysis, we compare the proposed algorithms with other existing clustering ensemble algorithms on several data sets. The comparison results illustrate the proposed algorithms are very effective and robust.

Index Terms—Cluster analysis, cluster ensemble, information entropy, consensus function.

1 Introduction

Clustering is an important problem in statistical multivariate analysis, data mining and machine learning [1]. The goal of clustering is to group a set of objects into clusters so that the objects in the same cluster are highly similar but remarkably dissimilar with objects in other clusters [2]. To tackle this problem, various types of clustering algorithms have been developed in the literature (e.g., [3] and references therein), including partitional, hierarchical, density-based and grid-based clustering. However, there is no single clustering algorithm that is suitable to deal with all the clustering tasks. Each algorithm has its own strength and weakness [4], [5], [6], [7], [8]. Given a data set, different algorithms or the same algorithms with different input parameters often produce distinct clusterings. Therefore, it is extremely difficult for users to determine which clustering would be the proper one.

Recently, cluster ensemble techniques [9], [10] have been developed to overcome the limitations of a single clustering. The goal of cluster ensemble is to aggregate several base clusterings to generate a single output clustering with improved robustness and stability. It is a type of unsupervised ensemble learning [11]. In contrast to classifier ensemble, cluster ensemble lacks label information of data sets. Thus, it is very difficult to recognize the major strength and weakness of a clustering on a data set [12]. Currently, different types of cluster ensemble methods have been proposed, according to different scientific needs. Representative methods include the pairwise-similarity approach, the graph-based approach, the relabeling-based approach and the feature-based approach. The detailed review can be found in Section 2.

Although the existing cluster ensemble methods already have good theoretical and practical contributions, they still have some deficiencies which need be improved. We know the generation and integration of base clusterings are two major tasks of cluster ensemble. The uncertainties of the two tasks often affect the quality of the cluster ensemble result. For the generation of base clusterings, many scholars have studied the effect of diversity of base clusterings on cluster ensemble [11], [13]. They suggest to producing base clusterings with certain differences between each other, which can enhance the accuracy of cluster ensemble. For the integration of base clusterings, most of existing algorithms make an effort to obtain a final clustering with the most consensus on the base clustering set [9], [14], [15], which can improve the robustness of the clustering result. However, if a majority of base clusterings cannot effectively reflect the cluster structure on the original data set, the final clustering with the most consensus cannot be guaranteed to be good. Therefore, people need to consider the performance of the final clustering on both the original data set and the base clustering set.

In a cluster ensemble problem, each object can be seen to be embedded in two feature spaces, i.e., the original feature set and base clustering feature set. Thus, a good final clustering should have high consensus on both the feature spaces. Keeping the consensus of the final

• *This work is supported by the National Natural Science Foundation of China (Nos. 61773247, 61432011, 61573229, U1435212), the Technology Research Development Projects of Shanxi (No. 201701D221097).*
L. Bai, J. Liang and H. Du are with school of Computer and Information Technology, Shanxi University, Taiyuan, 030006, Shanxi, China (e-mail: bailiang@sxu.edu.cn, ljy@sxu.edu.cn, duhangyuan@sxu.edu.cn).
Y. Guo is with Department of Computing, Imperial College London, SW7, London, United Kingdom (e-mail: y.guo@imperial.ac.uk).

clustering on the base clustering feature set is to enhance the robustness of the clustering result. Keeping the consensus of the final clustering on the original feature set is to sufficiently reflect the cluster structure of the original data set. However, most of the existing algorithms rarely consider the clustering consensus on both the feature spaces. Although some scholars [16], [17], [18] employed clustering validity indices to measure the qualities of base clusterings, they do not construct an uniform framework to simultaneously consider the consensus of the final clustering on the two spaces. To solve this problem, we provide an information-theoretical framework for cluster ensemble. The goal of the framework is to obtain a final clustering with high consensus on both the base clustering set and original data set by minimizing the uncertainties on the generation and integration of base clusterings. In this framework, we employ information entropy to evaluate the two uncertainties and develop three weighted cluster ensemble approaches with different ensemble strategies to obtain a clustering result with high quality.

The outline of the rest of this paper is as follows. Section 2 reviews the related works. Section 3 presents a weighted consensus measure based on information entropy. Sections 4 proposes an information-theoretical framework for cluster ensemble. Section 5 demonstrates the performance of the proposed ensemble algorithms. Section 6 concludes the paper with some remarks.

2 Related Works

For cluster ensemble, there are two major research tasks: 1) constructing a generator to produce base clusterings and 2) devising a ensemble strategy to produce a final partition. They are also two major factors which affect the performance of a cluster ensemble method.

In ensemble learning, it is observed that the diversity among classification results of base classifiers or clusterers, to some extent, can enhance the performance of the ensemble learner. Currently, several heuristics have been proposed to produce different clusterings on a data set, which can be classified into three categories:

- Repeatedly run a single clustering algorithm with different initial sets of parameters to produce base clusterings [19], [20], [21]. Fred and Jain [19] applied k-means with the different numbers of clusters to produce a clustering set. Kuncheva and Vetrov [20] used k-means with randomly selected different cluster centers. Zhang et al. [21] run the spectral clustering algorithm with different kernel parameters.
- Run different types of clustering algorithms to produce base clusterings [10], [22], [23]. Gionis et al [10] used several hierarchial clustering and k-means to produce a clustering set. Law et al. [22] applied multiple clustering algorithms with different objective functions as base clusterers and transformed a clustering ensemble problem as a multi-objective optimization. Yu et al [23] integrated several types of fuzzy clusterings.
- Run one or more clustering algorithms on different subspaces or subsamples from the data set [13], [24], [25], [26], [27], [28], [29], [30]. Fischer and Buhmann [24] applied the bootstrap method to obtain several data subsets. Fern and Brodley [27] used the random projection method to obtain several feature subspaces. Zhou et al [13] used different kernel functions to describe the data. Yang et al [30] proposed a novel hybrid sampling method for cluster ensemble by combining the strengths of boosting and bagging.

According to ensemble strategy, the cluster ensemble methods can be classified into the following four categories:

- The pairwise-similarity approach that makes use of co-occurrence relationships between all pairs of data objects to aggregate multiple clusterings [31], [17], [32], [33], [34]. Fred and Jain [31] proposed an ensemble algorithm based on evidence accumulation and constructed a co-association (CO) matrix. Yang et al [17] made use of clustering validity functions as weights to construct a weighted similarity matrix. Iam-On et al [32], [33] defined a link-based similarity matrix which sufficiently considers the similarity between clusters. Yu et al. [34] measured the label consistency between two objects on different subspace clusterings to construct the pairwise-similarity matrix.
- The graph-based approach that expresses the base clustering information as an undirected graph and then derives the ensemble clustering via graph partitioning [9], [35], [36], [37]. Strehl et al. [9] proposed three hypergraph ensemble algorithms CSPA, HGPA, and MCLA. CSPA creates a similarity graph, where vertices represent objects and the weights of edges represent similarity. HGPA constructs a hypergraph, where vertices represent objects and the same-weighted hyperedges represent clusters. MCLA generates a graph where the vertices represent clusters and the weights of edges reflect the similarity between clusters. Fern and Brodley et al [35] proposed the HBGF algorithm where vertices represent both objects and clusters. Huang et al. [36] proposed a graph algorithm based on random walk to recognize uncertain links in cluster ensemble.
- The relabeling-based approach that expresses the base clustering information as label vectors and then aggregates via label alignment [16], [25], [39], [40], [38]. Its representative methods can be classified into two types: crisp label correspondence and soft label correspondence. The crisp methods [16], [25], [39] transfer the relabeling problem into a minimum cost one to-one assignment problem. Long et al [40] used an alternating optimization strategy to solve the soft label alignment problem. Rathore et al. [41] proposed an efficient fuzzy ensemble framework

which uses a cumulative agreement scheme to align fuzzy clusters.

- The feature-based approach that treats a cluster ensemble problem as a clustering problem on the base clustering set [42], [14], [43], [44], [45], [46], [47], [48]. Cristofor and Simovici [42] integrated the information theory and genetic algorithms to search for the most consensus clustering. Topchy et al [14] proposed a probabilistic framework and used the EM algorithm to find the consensus clustering. Nguyen et al [45] made use of the k-modes [46] as the consensus function for clustering ensemble. Liu and Wu et al. [47], [48] proposed a k-means-based clustering ensemble algorithm to deal with large-scale data sets.

In this paper, our study focuses on the ensemble strategy. We will develop an information-theoretical framework for cluster ensemble to minimize the uncertainties on the generation and integration of base clusterings. We know that cluster analysis is no stranger to information theory [49]. Many entropy-based clustering algorithms have been developed, since information theory is an obvious criteria to establish the clustering rule. For example, Gokcay et al [50] proposed the Renyi's entropy-based clustering algorithm for discovering non-linearly separable clusters. Karayiannis et al [51] proposed the maximum-entropy algorithm for fuzzy clustering. Some scholars used information entropy as clustering criteria to cluster categorical data, such as Coolcat [52] and ACE [53]. The Coolcat algorithm determines the arrangement of each object by minimizing the expected entropy. The ACE algorithm applies the incremental entropy as a merge criterion for hierarchically clustering categorical data. In [54], we analyzed the relations between information entropy-based clustering criterion and other criteria for categorical data. Besides, information entropy is also used to cluster network data. Rosvall et al [55] developed an information-theoretical framework for clustering network data. In [56], information entropy is used to evaluate the importance of network node and build a new community description model. According to the above introduction, we can see that the information theory is a very good tool for cluster analysis. Therefore, in this paper, we will discuss how information theory is applied to cluster ensemble.

3 The Consensus Measure Based on Information Entropy

3.1 The cluster ensemble problem

We first introduce the formulation of a cluster ensemble problem as follows. Let $X = \{\mathbf{x}_i\}_{i=1}^N$ be a set of N objects, $\mathfrak{C} = \{\mathbb{C}_h\}_{h=1}^T$ be a set of T base clusterings, $\mathbb{C}_h = \{C_{hl}\}_{l=1}^{k_h}$ be the hth base clustering where k_h is its number of clusters and C_{hl} is its lth cluster, and $\mathbb{C}_* = \{C_{*l}\}_{l=1}^k$ where k is its number of clusters and C_{*l} is its lth cluster. The cluster ensemble task is to aggregate T base clusterings of the object set X to find out the final clustering $\mathbb{C}_*$.

The main goal of existing cluster ensemble algorithms is to obtain the most consensus of the base clusterings. The more consistent the final clustering is with the base clusterings, the more robust it is thought to be. However, the qualities of base clusterings often affect the performance of the final clustering. Thus, people also should consider whether the final clustering can reflect the cluster structure on the original data set. For a good clustering, the objects in the same clusters should have high consensus on the features of the original data set. Therefore, we need to take full account of the relation of the original data set, its base clustering set and its final clustering as seen in Fig.1. In cluster ensemble,

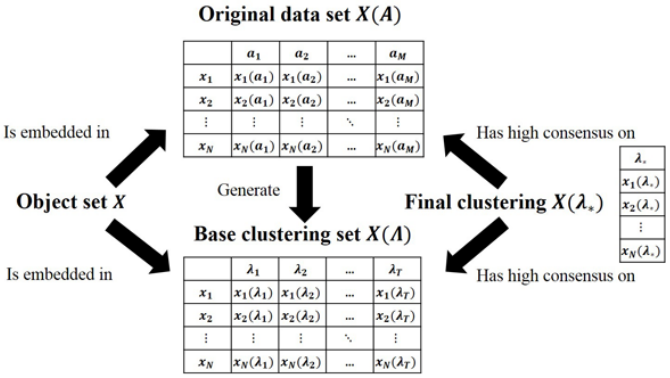

Fig. 1. Relation of the original data set, the base clustering set and the final clustering in cluster ensemble

the original data set and the base clustering set can be viewed as the representations of the object set X on different feature spaces, respectively. Therefore, in this paper, the quality of a final clustering is evaluated based on the following two criteria.

- The final clustering is good when it has high consensus on the base clustering feature set.
- The final clustering is good when it has high consensus on the original feature set.

In this case, the goal of the cluster ensemble task becomes obtaining a final clustering of the object set X which satisfies both the criteria.

For the ease of evaluating the above consensus, we formulate the two representations of X as follows. $A = \{a_j\}_{j=1}^M$ is a set of M original features. $X(A)$ is a N-by-M matrix that is the representation of X on A, where $X(a_j)$ is the jth column, $\mathbf{x}_i(A)$ is the ith row and $\mathbf{x}_i(a_j)$ is the value of object $\mathbf{x}_i$ in feature a_j. $X(A)$ is called the original data set of X. $\Lambda = \{\lambda_h\}_{h=1}^T$ is a set of T base clustering features. λ_h corresponds to the hth base clustering $\mathbb{C}_h$. $X(\Lambda)$ is a N-by-T matrix that is the representation of X on Λ. $X(\lambda_h)$ is the hth column of $X(\Lambda)$ which represents the hth base clustering. $\mathbf{x}_i(\Lambda)$ is the ith row of $X(\Lambda)$ which represents the description of the ith object on Λ. $\mathbf{x}_i(\lambda_h)$ is the cluster label of object $\mathbf{x}_i$ in base clustering feature λ_h. $X(\Lambda)$ is called the base

clustering set (matrix) of X. Besides, the final clustering $\mathbb{C}_*$ also can be represented by $X(\lambda_*)$ where λ_* is the final clustering feature. Given $\mathbb{C}_*$, $X(\lambda_*)$ can be computed as follows. If $\mathbf{x}_i \in C_{*l}$, then $\mathbf{x}_i(\lambda_*) = \lambda_{*l}$, where λ_{*l} is the cluster label of C_{*l} for $1 \leq i \leq N$. Furthermore, we provide Table 1 to summarize the main symbols used in this paper.

TABLE 1
Description of the main symbols used in this paper

Symbol	Description
$\mathbf{x}_i$	The ith object
$X = \{\mathbf{x}_i\}_{i=1}^N$	Set of N objects
$\mathbb{C}_h$	The hth base clustering
$\mathfrak{C}$	Set of T base clusterings
C_{hl}	The lth cluster of $\mathbb{C}_h$
k_h	The number of clusters in $\mathbb{C}_h$
$\mathbb{C}_*$	Final clustering
C_{*l}	The lth final cluster
k	The number of clusters in $\mathbb{C}_*$
$K = \sum_{h=1}^T k_h$	The number of all the clusters in $\mathfrak{C}$
a_j	The jth original feature
$A = \{a_j\}_{j=1}^M$	Set of M original features
$X(A)$	Original data set (matrix) of X
$X(a_j)$	The jth column of $X(A)$
$\mathbf{x}_i(A)$	The ith row of $X(A)$
$\mathbf{x}_i(a_j)$	Value of $\mathbf{x}_i$ in a_j
λ_h	The hth base clustering feature
$\Lambda = \{\lambda_h\}_{h=1}^T$	Set of T base clustering features
$X(\Lambda)$	Base clustering set (matrix) of X
$X(\lambda_h)$	The hth column of $X(\Lambda)$
$\mathbf{x}_i(\Lambda)$	The ith row of $X(\Lambda)$
$\mathbf{x}_i(\lambda_h)$	Cluster label of $\mathbf{x}_i$ in λ_h
λ_{hl}	The cluster label of C_{hl}
λ_*	Final clustering feature
λ_{*l}	The cluster label of C_{*l}
$D(\mathbf{v})$	Domain of (feature) variable $\mathbf{v}$
$H(.)$	Information entropy measure
$P(.)$	Probability of an event
$f(.)$	Probability density function
$S \subseteq X$	Object set or Cluster of X
$\mathbb{S} = \{S_l\}_{l=1}^p$	Clustering of S where S_l is the lth cluster
w_h	Importance (weight) of λ_h
$H_w(.)$	Weighted information entropy measure
$I_w(\Lambda\|\mathbb{S})$	Weighted consensus of $\mathbb{S}$ on Λ
$I(A\|\mathbb{C}_h)$	Consensus of $\mathbb{C}_h$ on A
$\kappa(.,.)$	Gaussian kernel function
$\phi(A)$	Mapped feature set of A
$\theta(.,.)$	Similarity measure between clusters
$C_{\mathbf{x}_i(\lambda_h)}$	Cluster which $\mathbf{x}_i$ belongs to in λ_h
$\Omega = \{C_{hl}\}$	Set of all the clusters from $X(\Lambda)$
$\mathbb{G} = \{G_l\}_{l=1}^k$	Partition of Ω where G_l is the lth subset
$\psi(.,.)$	Similarity measure between objects

In the next subsections, we will introduce how information entropy is used to evaluate the consensus of a clustering on two feature sets.

3.2 The basic concepts of information entropy

We give the relative definitions of the information entropy [49] which is the measure of information and uncertainty of a random variable.

Let $\mathbf{v}$ be a random variable and $D(\mathbf{v})$ be its domain which is a set including all the possible values of the variable. If $\mathbf{v}$ is discrete variable, i.e., $D(\mathbf{v}) = \{v_1, v_2, ..., v_k\}$, its information entropy (Shannon entropy) is defined as

$$H(\mathbf{v}) = -\sum_{i=1}^{k} P(v_i) \log P(v_i),$$

where $P(v_i)$ is the probability of the event $\mathbf{v} = v_i$. If $\mathbf{v}$ is continuous variable, its information entropy (Continuous entropy) is defined as

$$H(\mathbf{v}) = -\int f(\mathbf{v}) \log f(\mathbf{v}) d\mathbf{v},$$

where $f(\mathbf{v})$ is a probability density function of $\mathbf{v}$. The less the $H(\mathbf{v})$ value is, the more certain the value of the variable is. The conditional entropy quantifies the amount of uncertainty of a random variable, given that the value of another random variable is known. Let $\mathbf{u}$ be a discrete variable and $D(\mathbf{u}) = \{u_1, u_2, ..., u_{k'}\}$ be its domain. The conditional entropy of $\mathbf{v}$ given $\mathbf{u}$ is defined as

$$H(\mathbf{v}|\mathbf{u}) = \sum_{j=1}^{k'} P(u_j) H(\mathbf{v}|u_j),$$

where if $\mathbf{v}$ is discrete, then

$$H(\mathbf{v}|u_j) = -\sum_{i=1}^{k} P(v_i|u_j) \log P(v_i|u_j),$$

else

$$H(\mathbf{v}|u_j) = -\int f(\mathbf{v}|u_j) \log f(\mathbf{v}|u_j) d\mathbf{v}.$$

The information entropy will be used to evaluate the consensus of an object set on a given feature set. The more objects have the same or similar feature values, the more certain the feature values of these objects are.

3.3 The weighted consensus measure

In this subsection, we first introduce how to apply Shannon entropy to measure the consensus of a cluster on the clustering feature set Λ. We consider that each λ_h is a random variable, for $1 \leq h \leq T$. λ_h has a finite possible number of distinct label values, i.e., $D(\lambda_h) = \{\lambda_{hl}\}_{l=1}^{k_h}$ where λ_{hl} is the cluster label of C_{hl}. $P(\lambda_{hl})$ represents the probability of $\lambda_h = \lambda_{hl}$. Let S be a cluster, where $S \subseteq X$. The λ_h values of objects in S can be seen as a sample set of the discrete random variable λ_h. If we estimate $P(\lambda_{hl}|S)$ by the relative frequency of the label value λ_{hl} in S which is described as

$$P(\lambda_{hl}|S) = \frac{|\{\mathbf{x}_i|\mathbf{x}_i(\lambda_h) = \lambda_{hl}, \mathbf{x}_i \in S\}|}{|S|}, \quad (1)$$

the information entropy of λ_h given S can be written as

$$H(\lambda_h|S) = -\sum_{l=1}^{k_j} P(\lambda_{hl}|S) \log P(\lambda_{hl}|S). \quad (2)$$

We have $0 \leq H(\lambda_h|S) \leq -\log \frac{1}{k_h}$. The more objects in S have the same labels in the base clustering, the lower the

$H(\lambda_h|S)$ value is. Thus, we can use $H(\lambda_h|S)$ to evaluate the consensus of cluster S on the clustering feature λ_h.

Furthermore, we consider Λ is a set of T random variables and each λ_h is independent of each other. The weighted information entropy of Λ given S is defined as

$$H_w(\Lambda|S) = \sum_{h=1}^{T} w_h H(\lambda_h|S), \tag{3}$$

where w_h is a weight of λ_h. $H_w(\Lambda|S)$ can be seen as the weighted sum of the consensus of S on all the base clustering features. If each w_h is equal to 1, $H_w(\Lambda|S)$ becomes the classical information entropy. In this paper, we assume different base clusterings should play different roles in evaluating the consensus of S. A computing method for the weight will be discussed in Section 3.4.

Let us consider an example in Table 2 to demonstrate the performance of $H_w(\Lambda|S)$. There are seven objects with three base clusterings. We use the information entropy to evaluate the consensus of clusters $S_1 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ and $S_2 = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_4\}$, respectively. We first assume each weight is equal to 1. $H_w(\Lambda|S_1)$ and $H_w(\Lambda|S_2)$ are computed as follows. $H_w(\Lambda|S_1) = 0$ and $H_w(\Lambda|S_2) = 0 - \frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3} - \frac{1}{3}\log\frac{1}{3} \approx 1.2730$. We have $H_w(\Lambda|S_1) < H_w(\Lambda|S_2)$. According to Table 2, we know that the consensus of S_1 on Λ is obviously better than S_2. Thus, the information entropy measure can be used to evaluate the consensus of a cluster on the clustering feature set.

TABLE 2
Example about seven objects with two clusters.

	λ_1	λ_2	λ_3
$\mathbf{x}_1$	1	1	1
$\mathbf{x}_2$	1	1	1
$\mathbf{x}_3$	1	1	1
$\mathbf{x}_4$	1	2	2
$\mathbf{x}_5$	2	2	2
$\mathbf{x}_6$	2	2	3
$\mathbf{x}_7$	1	2	3

Next, we discuss how to apply information entropy to measure the consensus of a clustering on Λ. Let $S \subseteq X$ be an object set and $\mathbb{S} = \{S_l\}_{l=1}^{p}$ be a clustering of S, including p clusters, where S_l is the lth cluster, for $1 \leq l \leq p$. According to the definition of conditional entropy, the weighted information entropy of Λ given $\mathbb{S}$ is described as

$$H_w(\Lambda|\mathbb{S}) = \sum_{l=1}^{p} P(S_l|\mathbb{S}) H_w(\Lambda|S_l), \tag{4}$$

where $P(S_l|\mathbb{S})$ is the probability of cluster S_l in $\mathbb{S}$, which can be estimated by

$$P(S_l|\mathbb{S}) = \frac{|S_l|}{\sum_{l=1}^{p}|S_l|}. \tag{5}$$

$H_w(\Lambda|\mathbb{S})$ can be seen as the weighted sum of the consensus of all the clusters of $\mathbb{S}$. If each of its clusters has high consensus on Λ, the $H_w(\Lambda|\mathbb{S})$ value is low.

Based on Eqs. (3) and (4), we define the weighted consensus measure of $\mathbb{S}$ on Λ as follows.

$$I_w(\Lambda|\mathbb{S}) = H_w(\Lambda|S) - H_w(\Lambda|\mathbb{S}). \tag{6}$$

The consensus measure $I_w(\Lambda|\mathbb{S})$ compares the consensus of the object set S with the overall consensus of all the clusters of $\mathbb{S}$. It reflects the consensus change of the object set S on Λ brought by the clustering $\mathbb{S}$. We think that if $\mathbb{S}$ is a good clustering of S, it can enhance the consensus of S on Λ. Therefore, the larger $I_w(\Lambda|\mathbb{S})$ is, the higher the consensus of the clustering on Λ is.

The consensus measure $I_w(\Lambda|\mathbb{S})$ can be used to evaluate the quality of the clustering and the similarity between clusters. For example, if $\mathbb{S} = \mathbb{C}$ is a clustering of X, according to Eq.(6), we have

$$I_w(\Lambda|\mathbb{C}) = H_w(\Lambda|X) - H_w(\Lambda|\mathbb{C}).$$

Given X, $H_w(\Lambda|X)$ is constant. The more objects in each cluster of $\mathbb{C}$ have the same label values on Λ, the lower the $H_w(\Lambda|\mathbb{C})$ value is. Therefore, a clustering with high consensus on Λ is thought to have good performance. Besides, if $\mathbb{S} = \{S_i, S_j\}$ is a clustering including two clusters S_i and S_j, according to Eq.(6), we have

$$I_w(\Lambda|\{S_i, S_j\}) = H_w(\Lambda|S_i \cup S_j) - H_w(\Lambda|\{S_i, S_j\}).$$

Given S_i and S_j, $H_w(\Lambda|\{S_i, S_j\})$ is constant. If most of objects in $S_i \cup S_j$ have the same label values on Λ, $I_w(\Lambda|\{S_i, S_j\})$ is very low. In this case, the two clusters are thought to be similar on Λ.

However, according to the above analysis, we see that the measure $I_w(\Lambda|\mathbb{S})$ seemly only considers the consensus of a clustering on Λ. In order to make the measure $I_w(\Lambda|\mathbb{S})$ simultaneously evaluate the consensus on both Λ and A, in the following, we will propose an importance measure of a base clustering to evaluate its consensus on A, which will be used to compute its weight in the measure $I_w(\Lambda|\mathbb{S})$.

3.4 The importance measure of a base clustering

In the consensus measure $I_w(\Lambda|\mathbb{S})$, we need to provide a weight for each base clustering which may play different role in the process of cluster ensemble. Therefore, it is a key issue how to evaluate the importance of a base clustering. In this paper, the consensus of a base clustering on the original feature set A is used to reflect its importance. We continue using the information entropy to evaluate the consensus. For a cluster S, $H(A|S)$ is the information entropy of A given S. The lower the $H(A|S)$ value is, the more objects in S have the same or similar feature values on A. Therefore, the consensus of base clustering $\mathbb{C}_h$ on A is defined as

$$I(A|\mathbb{C}_h) = H(A|X) - H(A|\mathbb{C}_h),$$

where

$$H(A|\mathbb{C}_h) = \sum_{l=1}^{k_h} \frac{|C_{hl}|}{N} H(A|C_{hl})$$

is the conditional entropy of A given $\mathbb{C}_h$. Similarly to the consensus measure on Λ, $I(A|\mathbb{C}_h)$ reflects the consensus change of X on A brought by base clustering $\mathbb{C}_h$. If the $I(A|\mathbb{C}_h)$ value of base clustering $\mathbb{C}_h$ is large, it is thought to have high consensus on A.

However, since each feature variable in A maybe not discrete, the computing method for $H(A|S)$ is different from $H_w(\Lambda|S)$. In order to compute $H(A|S)$, we use continuous entropy, instead of Shannon entropy. In this case, we need to determine a probability density function of the original feature set A. We suggest to using a gaussian density function, which assumes that each cluster subjects to a gaussian distribution in a given feature space. Given cluster S, the gaussian density function of the feature set A is defined as

$$f(A|S) = \frac{1}{\sigma_S\sqrt{2\pi}} e^{-\frac{||\mathbf{x}(A)-\mu_S||^2}{2\sigma_S^2}} \tag{7}$$

where μ_S and σ_S^2 are the mean and variance of cluster S on the feature set A, respectively. However, not all the clusters on the feature set A satisfy the assumption of the gaussian distribution. Thus, we use the gaussian kernel κ to map the original feature set A into a new feature set with the gaussian distribution, i.e., ϕ: $A \rightarrow \phi(A)$. $\phi(A)$ is expressed by the gaussian kernel function which is described as

$$\begin{aligned}\kappa(\mathbf{x}_i, \mathbf{x}_j) &= < \mathbf{x}_i(\phi(A)), \mathbf{x}_i(\phi(A)) > \\ &= e^{-\frac{1}{\beta}||\mathbf{x}_i(A)-\mathbf{x}_j(A)||^2},\end{aligned} \tag{8}$$

where the parameter β is set to the standard deviation of $||\mathbf{x}_i(A) - \mathbf{x}_j(A)||^2$, $1 \leq i, j \leq N$.

After the data set is mapped, we use $H(\phi(A)|S)$, instead of $H(A|S)$, to evaluate the importance of a base clustering. According to the literature [49], $H(\phi(A)|S)$ can be computed by

$$H(\phi(A)|S) = \frac{1}{2}\log(2\pi e\sigma_S^2). \tag{9}$$

Eq.(9) tells us that we only need to compute σ_S^2, in order to get $H(\phi(A)|S)$. According to the definition of the kernel function, σ_S^2 can be computed by

$$\sigma_S^2 = \frac{1}{|S|}\sum_{\mathbf{x}_i \in S} ||\mathbf{x}_i(\phi(A)) - \mu_S||^2$$

and

$$||\mathbf{x}_i(\phi(A)) - \mu_S||^2 = 1 - \frac{2\sum\limits_{\mathbf{x}_j \in S}\kappa(\mathbf{x}_i, \mathbf{x}_j)}{|S|} + \frac{\sum\limits_{\mathbf{x}_p, \mathbf{x}_q \in S}\kappa(\mathbf{x}_p, \mathbf{x}_q)}{|S|^2}.$$

Based on Eq.(9), the consensus of base clustering $\mathbb{C}_h$ on $\phi(A)$ is defined as

$$\begin{aligned}I(\phi(A)|\mathbb{C}_h) &= H(\phi(A)|X) - H(\phi(A)|\mathbb{C}_h) \\ &= \frac{1}{2}\log(2\pi e\sigma_X^2) - \sum_{l=1}^{k_h}\frac{|C_{hl}|}{|X|}\frac{1}{2}\log(2\pi e\sigma_{C_{hl}}^2).\end{aligned} \tag{10}$$

In this paper, we assume that the importance of base clustering $\mathbb{C}_h$ is proportional to $I(\phi(A)|\mathbb{C}_h)$. Therefore, we define the importance measure of a base clustering as

$$w_h = \frac{e^{\frac{1}{\alpha}I(\phi(A)|\mathbb{C}_h)}}{\sum\limits_{j=1}^{T} e^{\frac{1}{\alpha}I(\phi(A)|\mathbb{C}_j)}}, \tag{11}$$

where the parameter α is set to the standard deviation of $I(\phi(A)|\mathbb{C}_j)$, $1 \leq j \leq T$, which is used to control the sparsity of these weights. We use the importance measure to compute the weight of each base clustering for the consensus measure $I_w(\Lambda|\mathbb{S})$. The weighted consensus measure can effectively evaluate the consensus of a clustering on both the original feature set and the base clustering feature set.

4 The Ensemble Algorithms Based on Information Entropy

In this section, we propose an information-theoretical framework for cluster ensemble. The framework employs the proposed consensus measure to evaluate the quality of a clustering, the similarity between clusters and the similarity between objects. Based on the proposed measure, we develop three cluster ensemble algorithms which use different ensemble strategies to find a final clustering with high consensus on both the original feature set and the base clustering feature set. Fig.2 shows the relation of these algorithms. The first algorithm is a feature-based approach, called the Weighted Feature Consensus (WFC) algorithm. It views a cluster ensemble task as clustering on the base clustering matrix and uses the weighted information entropy as the objective function to find a good final clustering. The second algorithm is a relabeling-based approach, called the Weighted Relabeling Consensus (WRC) algorithm. It employs the proposed consensus measure to compute the similarity between clusters and determines which clusters represent the same final clusters based on the cluster-similarity matrix. The third algorithm is a pairwise-similarity approach, called the Weighted Pairwise-similarity Consensus (WPC) algorithm. It computes the similarity between objects based on the cluster-similarity matrix obtained by the WRC algorithm and partitions objects into the final clusters based on the object-similarity matrix. In the following, we will introduce these algorithms in detail.

4.1 The weighted feature consensus algorithm

In the subsection, we propose a feature-based approach of cluster ensemble based on information entropy. In the approach, we assume that a good final clustering should have high consensus on both the original feature set and the base clustering feature set. Therefore, we employ the weighted information entropy $H_w(\Lambda|\mathbb{C}_*)$ as the objective function of this approach and design an optimization model to directly perform the cluster ensemble task on the base clustering set.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2018.2865954, IEEE Transactions on Knowledge and Data Engineering

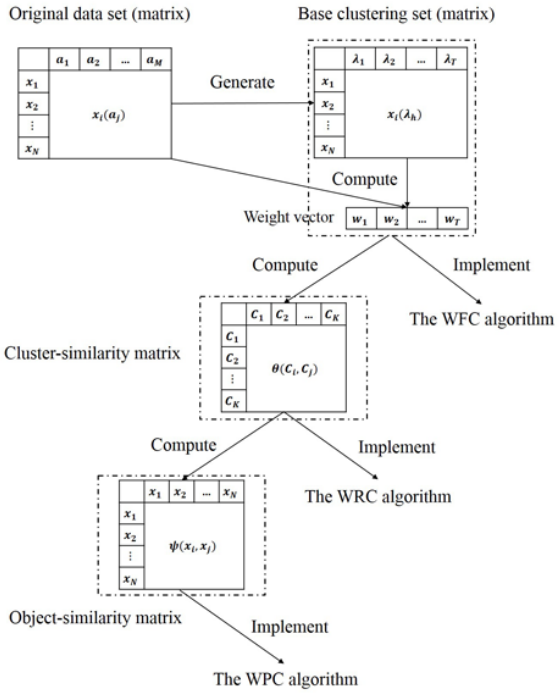

Fig. 2. Cluster ensemble framework

The optimization model of the cluster ensemble is formally described as

$$\min_{\mathbb{C}_*} H_w(\Lambda|\mathbb{C}_*). \tag{12}$$

We know that given X,

$$\max_{\mathbb{C}_*} I_w(\Lambda|\mathbb{C}_*) = H_w(\Lambda|X) - \min_{\mathbb{C}_*} H_w(\Lambda|\mathbb{C}_*). \tag{13}$$

This tells us that minimizing the information entropy $H_w(\Lambda|\mathbb{C}_*)$ is equivalent to maximizing the weighted consensus of the final clustering.

In order to solve the optimization problem, we use a hierarchical clustering strategy to find out the final clustering. Let $\mathbb{C}^{(t)}$ be the partition of X produced at the tth merge step. We first produce an initial partition $\mathbb{C}^{(0)}$ where each object is as a cluster. In this case, $H_w(\Lambda|\mathbb{C}^{(0)}) = 0$. Next, we continuously combine two clusters to obtain a new partition $\mathbb{C}^{(t+1)}$. According to the property of the conditional entropy, we have

$$H_w(\Lambda|\mathbb{C}^{(t+1)}) \geq H_w(\Lambda|\mathbb{C}^{(t)}).$$

This tells us that as the number of clusters decreases, the consensus of the clustering result decreases. We wish that the consensus of the clustering result decreases as little as possible at each merge step. Therefore, given $\mathbb{C}^{(t)}$, the optimization problem (12) becomes

$$\begin{aligned} &\min_{\mathbb{C}^{(t+1)}} \left[H_w(\Lambda|\mathbb{C}^{(t)}) - H_w(\Lambda|\mathbb{C}^{(t+1)}) \right] \\ &= \min_{C_i^{(t)}, C_j^{(t)} \in \mathbb{C}^{(t)}} I_w(\Lambda|\{C_i^{(t)}, C_j^{(t)}\}). \end{aligned} \tag{14}$$

According to Eq.(14), in each step, we merge the two most consistent clusters $C_i^{(t)}$ and $C_j^{(t)} \in \mathbb{C}^{(t)}$. The merge criterion is described as

$$\min_{C_i^{(t)}, C_j^{(t)} \in \mathbb{C}^{(t)}} I_w(\Lambda|\{C_i^{(t)}, C_j^{(t)}\}). \tag{15}$$

When the number of clusters in $\mathbb{C}^{(t)}$ is equal to k, the merge step is terminated and $\mathbb{C}_* = \mathbb{C}^{(t)}$.

The feature-based approach is formally described in Algorithm 1. The basic operation of the algorithm is $I(.,.)$ whose computing cost is $O(K)$, where $K = \sum_{h=1}^{T} k_h$. The computing cost of the hierarchical clustering strategy of the algorithm is $O(N^2 \log N)$. Therefore, the time complexity of the algorithm is $O(N^2 \log NK)$.

Algorithm 1: The WFC algorithm

Input: $X(A)$, $X(\Lambda)$, k
Output: $X(\lambda_*)$
Set $t = 0$ and produce an initial partition $\mathbb{C}^{(t)} = \{C_i^{(t)}\}_{i=1}^{N}$ where $C_i^{(t)} = \{\mathbf{x}_i\}$,$1 \leq i \leq N$;
while $|\mathbb{C}^{(t)}| > k$ **do**
  Select two clusters $C_i^{(t)}$ and $C_j^{(t)}$ which satisfy Eq.(15);
  $C_i^{(t)} = C_i^{(t)} \cup C_j^{(t)}$;
  $\mathbb{C}^{(t+1)} = \mathbb{C}^{(t)} - C_j^{(t)}$;
  $t = t + 1$;
Get $X(\lambda_*)$ corresponding to $\mathbb{C}^{(t)}$;
Return $X(\lambda_*)$;

4.2 The weighted relabeling consensus algorithm

In the subsection, we propose a relabeling-based approach of cluster ensemble based on information entropy. It is an important step for the relabeling-based approaches to compute the similarity between clusters from the same or different base clusterings. The similarity is used to determine whether two clusters from base clusterings represent the same final cluster. In many relabeling-based approaches [9], [16], the number of common objects between clusters is widely used to reflect their similarity. The more the number of their common objects is, the more similar they are thought to be. However, if there are few objects which belong to both them, the measure can not fully reflect their similarity. Therefore, we employ the proposed consensus measure to evaluate the similarity between clusters as follows.

Let $\Omega = \{C_{hl}\}$, $1 \leq h \leq T, 1 \leq l \leq k_h$, be a set of all the clusters from $X(\Lambda)$. For any two clusters $C_{ip}, C_{jq} \in \Omega$, their similarity is described as

$$\theta(C_{ip}, C_{jq}) = e^{-\frac{1}{\gamma} I_w(\Lambda|\{C_{ip}, C_{jq}\})}, \tag{16}$$

where the parameter γ is set to the standard deviation of $I_w(\Lambda|\{C_{ip}, C_{jq}\})$ for all the clusters from Ω. In the measure, the set of two clusters is seen as a clustering. Their similarity is inversely proportional to the consensus of

the clustering. The high consensus of the clustering indicates that the two clusters tend to representing different final clusters.

According to the definition of the similarity, we can obtain a $K \times K$ cluster-similarity matrix. Based on the similarity matrix, the relabeling problem can be transferred to a graph mini-cuts problem which is described as follows [7].

$$\min_{\mathbb{G}} \left[Q(\mathbb{G}) = \sum_{l=1}^{k} \sum_{C_{ip} \in G_l, C_{jq} \in \Omega - G_l} \theta(C_{ip}, C_{jq}) \right], \quad (17)$$

where $\mathbb{G} = \{G_l\}_{l=1}^{k}$ is a partition of Ω and G_l is the lth subset of Ω. We wish to obtain such a partition by minimizing the objective function Q that the clusters in the same subsets have very high similarity but are very dissimilar with clusters in other subsets. In order to solve the optimization problem, we apply the spectral clustering (SC) algorithm [8] with average-linkage to obtain a final partition of Ω. The clusters in the same subsets are used to represent the same final cluster. Therefore, we use the label values of the final clustering variable λ_* as the labels of the subsets of $\mathbb{G}$. Based on $\mathbb{G}$, we relabel $X(\Lambda)$ as follows.

$$\mathbf{x}_i(\lambda_h) = \lambda_{*l}, \ if \ C_{\mathbf{x}_i(\lambda_h)} \in G_l, \quad (18)$$

where $C_{\mathbf{x}_i(\lambda_h)}$ is the cluster which $\mathbf{x}_i$ belongs to in λ_h, for $1 \leq i \leq N$ and $1 \leq h \leq T$.

After the base clustering set $X(\Lambda)$ is relabeled, the optimization problem of cluster ensemble is written as follows.

$$\max_{X(\lambda_*)} \left[E(X(\lambda_*)) = \sum_{i=1}^{N} \sum_{h=1}^{T} w_h \delta(\mathbf{x}_i(\lambda_h), \mathbf{x}_i(\lambda_*)) \right], \quad (19)$$

where

$$\delta(x, y) = \begin{cases} 1, & x = y, \\ 0, & x \neq y. \end{cases}$$

Maximizing $E(X(\lambda_*))$ aims to finding the most consensus label of each object. We can maximize the objective function E by the following equation

$$\mathbf{x}_i(\lambda_*) = \arg\max_{l=1}^{k} \sum_{h=1}^{T} w_h \delta(\mathbf{x}_i(\lambda_h), \lambda_{*l}), \quad (20)$$

for $1 \leq i \leq N$.

The relabeling-based approach is formally described in Algorithm 2. The computing cost of obtaining a cluster-similarity matrix is $O(K^3)$. Relabeling all the clusters needs $O(K^2 \log K)$ costs. Maximizing the objective function E needs $O(NK)$ costs. Therefore, the time complexity of the algorithm is $O(K^3 + K^2 \log K + NK)$.

4.3 The weighted pairwise-similarity consensus algorithm

In the subsection, we propose a pairwise-similarity approach of cluster ensemble based on information entropy. Many pairwise-similarity methods use the simple matching method to measure the similarity between objects, which is described as follows.

Algorithm 2: The WRC algorithm

Input: $X(A)$, $X(\Lambda)$, k
Output: $X(\lambda_*)$
for $\forall C_{ip}, C_{jq} \in \Omega$ **do**
  Compute $\theta(C_{ip}, C_{jq})$ by Eq.(16);
Get $\mathbb{G} = \arg\min Q(\mathbb{G})$ by the SC algorithm [8] with average-linkage;
Relabel the base clustering set $X(\Lambda)$ by Eq.(18);
Get $X(\lambda_*) = \arg\max E(X(\lambda_*))$ by Eq.(20);
Return $X(\lambda_*)$;

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = \sum_{h=1}^{T} \delta(\mathbf{x}_i(\lambda_h), \mathbf{x}_j(\lambda_h)). \quad (21)$$

This measure evaluates the similarity between two objects by using the number of common clusters to which they belong. However, it does not fully consider the similarity between different clusters to which the objects belong. Let us continue using the example in Table 2 to show this issue. We compare the similarity between $\mathbf{x}_3$ and $\mathbf{x}_4$ with that between $\mathbf{x}_4$ and $\mathbf{x}_6$, according to the base clustering feature λ_3. Based on Eq.(21), we compute $\delta(\mathbf{x}_3(\lambda_3), \mathbf{x}_4(\lambda_3)) = \delta(\mathbf{x}_4(\lambda_3), \mathbf{x}_6(\lambda_3)) = 0$. We see that the similarity can not be distinguished, since the clusters to which objects belong in λ_3 are different. However, according to the proposed consensus measure, we have $I_w(\Lambda|\{C_{31}, C_{32}\}) > I_w(\Lambda|\{C_{32}, C_{33}\})$. This means that C_{32} is more similar with C_{33} than C_{31}. Thus, we conclude that $\mathbf{x}_4$ is more similar with $\mathbf{x}_6$ than $\mathbf{x}_3$ from the view of λ_3.

Therefore, we provide a similarity measure ψ between objects to obtain an object-similarity matrix, which is described as

$$\psi(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{1}{\tau} d(\mathbf{x}_i, \mathbf{x}_j)}, \quad (22)$$

where

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{h=1}^{T} w_h I_w(\Lambda|\{C_{\mathbf{x}_i(\lambda_h)}, C_{\mathbf{x}_j(\lambda_h)}\}). \quad (23)$$

The parameter τ is set to the standard deviation of $d(\mathbf{x}_i, \mathbf{x}_j)$ for $1 \leq i, j \leq N$. According to the definition $d(.,.)$, we see that object $\mathbf{x}_i$ is viewed as the cluster set $\{C_{\mathbf{x}_i(\lambda_h)}\}_{h=1}^{T}$. We evaluate the similarity between two objects by measuring the similarity between the clusters which they belong to in each base clustering.

After the object-similarity matrix is obtained, we transfer the cluster ensemble problem into a graph mini-cuts problem which is described as follows [7].

$$\min_{X(\lambda_*)} \left[F(X(\lambda_*)) = \sum_{l=1}^{k} \sum_{\mathbf{x}_i \in C_{*l}, \mathbf{x}_j \in X - C_{*l}} \psi(\mathbf{x}_i, \mathbf{x}_j) \right]. \quad (24)$$

We use the spectral clustering (SC) algorithm [8] with average-linkage to derive the final solution.

The pairwise-similarity approach is formally described in Algorithm 3. The computing cost of obtaining an object-similarity matrix is $O(N^2K)$. Minimizing the objective function F needs $O(N^2 \log N)$ costs. Therefore, the time complexity of the algorithm is $O(N^2K + N^2 \log N)$.

Algorithm 3: The WPC algorithm

Input: $X(A)$, $X(\Lambda)$, k
Output: $X(\lambda_*)$
for $1 \leq h \leq T$ **do**
  for $1 \leq p, q \leq k_h$ **do**
    Compute $I_w(\Lambda|\{C_{hp}, C_{hq}\})$ by Eq.(6);
for $1 \leq i, j \leq N$ **do**
  Compute $\psi(\mathbf{x}_i, \mathbf{x}_j)$ by Eqs.(22) and (23);
Get $X(\lambda_*) = \arg\min F(X(\lambda_*))$ by the SC algorithm [8] with average-linkage;
Return $X(\lambda_*)$;

5 Experimental Analysis

In this section, we carry out the proposed algorithms on nine real data sets and evaluate their performance by two validity criteria.

5.1 Data sets

The experimental evaluation is conducted on nine real data sets, including Iris, Wine, Dermatology, Breast cancers, Handwritten Digits, Landsat Satellite, Spambase, MNIST and KDDCUP99 from www.ics.uci.edu/mlearn /MLRepository.html and yann.lecun.com/exdb/mnist/. For KDDCUP99 data set, we selected the first 10,000 records which includes 8030 normal and 1970 abnormal connections to test the performance of different algorithms. Table 3 shows the details of these data sets.

TABLE 3
The description of data sets: Number of Data Objects (N), Number of Attributes (M), Number of Clusters (k).

Data set	N	M	k
Iris	150	4	3
Wine	178	13	3
Dermatology	336	20	6
Breast cancers	699	9	2
Handwritten Digits	5,620	63	10
Landsat Satellite	6,435	36	7
Spambase	4,601	20	2
MNIST	10,000	784	10
KDDCUP99	10,000	41	2

5.2 Evaluation criteria

We employ the two widely-used external criteria to measure the similarity between the clustering result and the true partition on a data set. Given a data set X and two partitions of these objects, namely $C = \{C_1, C_2, \cdots, C_k\}$ (the clustering result) and $P = \{P_1, P_2, \cdots, P_{k'}\}$ (the true partition), the overlappings between C and P can be summarized in a contingency table (Table 4) where n_{ij} denotes the number of common objects of groups C_i and P_j: $n_{ij} = |C_i \cap P_j|$. The adjusted rand index [57] is defined

TABLE 4
Notation for the contingency table for comparing two partitions.

$C \backslash P$	P_1	P_2	$\cdots$	$P_{k'}$	$Sums$
C_1	n_{11}	n_{12}	$\cdots$	$n_{1k'}$	b_1
C_2	n_{21}	n_{22}	$\cdots$	$n_{2k'}$	b_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
C_k	n_{k1}	n_{k2}	$\cdots$	$n_{kk'}$	b_k
$Sums$	d_1	d_2	$\cdots$	$d_{k'}$	

as

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{b_i}{2}\sum_j \binom{d_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{b_i}{2} + \sum_j \binom{d_j}{2}] - [\sum_i \binom{b_i}{2}\sum_j \binom{d_j}{2}]/\binom{n}{2}}$$

where n_{ij}, b_i, d_j are values from Table 4. The normalized mutual information (NMI) [58] is defined as

$$NMI = \frac{2\sum_i \sum_j n_{ij} \log \frac{n_{ij} n}{b_i d_j}}{-\sum_i b_i \log \frac{b_i}{n} - \sum_j d_j \log \frac{d_j}{n}}.$$

If the partition result is close to the true partition, then their values are high.

5.3 Compared algorithms

The WFC, WRC and WPC algorithms are different types of cluster ensemble. Therefore, in order to properly examine the performance of these algorithms, we compare them with different algorithms, respectively.

- The WFC algorithm is a kind of *feature-based algorithms*. We compare it with other three feature-based algorithms including the expectation maximization (EM) algorithm [14], the iterative voting consensus (IVC) algorithm [45] and the weighted k-means-based consensus clustering (WKCC) algorithm [47]. Besides, we also compare the WFC algorithm with the average results of the generated base clusterings.
- The WRC algorithm is a kind of *cluster-similarity algorithms*. We compare it with other four algorithms with different cluster-similarity matrices, including the MCLA algorithm [9], the selectively voting (SV) algorithm [16], the selectively weighted voting (SWV) algorithm [16], and the probability trajectory based graph partitioning (PTGP) algorithm [36].
- The WPC algorithm is a kind of *object-similarity algorithms*. We compare it with other four algorithms with different object-similarity matrices, including the CSPA algorithm [9], the HGPA algorithm [9], the WCT algorithm with the average-link (AL) [32], and the weighed cluster ensemble (WCE) algorithm [17].

The matlab codes of these compared algorithms, except the EM and IVC algorithms, were provided by their authors. We produced the matlab codes of the EM and IVC algorithms, according to their descriptions in the papers [14], [45]. The environment of experimental comparisons is Matlab R2016b and a PC with an Intel i7-4710MQ and 16 GB RAM.

5.4 Experimental Settings

To ensure that the comparisons are in an uniform environmental condition, we set that the number of clusters k_h in each base clustering is equal to the number of known classes on each of the given data sets.

Furthermore, we consider two schemes for the generation of base clusterings as follows.

- In the first scheme, we select four classical algorithms including the k-means [4], kernel k-means [5], non-negative matrix factorization [6] with k-means, spectral clustering [8] with k-means to produce base clusterings on a data set. For a data set, a base clustering set includes 20 clusterings, every five of which are produced by one of the selected algorithms with 5 different initial sets of cluster centers. In order to test the performance of clustering ensemble algorithms, we compare their the average results of cluster ensemble on 50 base clustering sets of each data set.
- In the second scheme, k-means is used as a generator of base clusterings. On each data set, we run k-means 100 times and select three groups, each of which includes 20 base clusterings. In order to test the robustness of algorithms on a data set, we compute the average ARI and NMI values of 100 clustering results and set that 50, 40 and 30 percents of the selected base clusterings have the higher ARI and NMI values than the average values. The scheme is used to test the robustness of different ensemble algorithms on the qualities of base clusterings. We compare their average results of cluster ensemble on 50 base clustering sets of each data set.

5.5 Experimental Results

Based on the validity criteria ARI and NMI, we evaluate the performance of different cluster ensemble algorithms on each of real data sets.

We first compare the performance of different algorithms on the first scheme. Table 5 shows the comparison of the WFC algorithm with the average results of base clusterings, the EM, IVC and WKCC algorithms. According to the table, we see that the ARI and NMI values of the WFC algorihm are obviously higher than other algorithms and the average results of base clusterings on the given data sets. The experimental results tell us that the proposed algorithm can improve the performance of the base clusterings, compared to other algorithms. Table 6 shows the comparison of the WRC algorithm with the MCLA, SV, SWV and PTGP algorithms. We see that the WRC algorithm has the same performance as the MCLA algorithm on Spamebase and KDDCUP99 data sets. On other data sets, the ensemble accuracies of the WRC algorithm are superior to other algorithms. Table 7 shows the comparison of the WPC algorithm with the CSPA, HGPA, WCT and WCE algorithms. We see that the ensemble results of the WPC algorithm can obtain the highest ARI and NMI values on the tested data sets. Therefore, according to these tables, we conclude that the proposed entropy-based framework can further enhance the accuracy of cluster ensemble.

Next, we test different algorithms on the second scheme. Table 8 shows the comparison of the WFC algorithm with the average results of the base clusterings produced by k-means, the EM, IVC and WKCC algorithms. The comparison results illustrate that the performance of the WFC algorithm is obviously better than other algorithms and the average results of base clusterings. Table 9 shows the comparison of the WRC algorithm with the MCLA, SV, SWV and PTGP algorithms. The WRC algorithm has the best performance on these given data sets, except Statlog. Table 10 illustrates the comparison of the WPC algorithm with the CSPA, HGPA, WCT and WCE algorithms. The ensemble results of the WPC algorithm for ARI and NMI are superior to other algorithms on these data sets, except Spambase. Besides, these tables show the mean and standard deviation of each algorithm for ARI and NMI on all the data sets. The proposed algorithms have high means in the tables, compared to other algorithms. The standard deviations of the proposed algorithms for the ARI and NMI measures are not the lowest values of the compared algorithms. However, we can see that their standard deviations are lower than 0.21, which indicates that the proposed algorithms are very stable on these given data sets. Furthermore, we can see the effect of the quality of base clusterings on the performance of the proposed algorithms on each data set, according to Tables 8, 9 and 10. These tables show that as the number of base clusterings with higher ARI and NMI values decreases, the performance of the proposed algorithms is little changed. This tells us that the proposed algorithms have good robustness for different qualities of base clusterings.

According to the above experiment analysis, we conclude that keeping consensus of a final clustering on both the original feature set and the base clustering feature set can improve the quality and robustness of the clustering result.

6 Conclusions

We developed an information-theoretic framework for cluster ensemble. The information entropy is used to define a weighted consensus measure which can reflect the consensus of a clustering on both the original feature set and the clustering feature set. The framework

TABLE 5
Comparisons of different feature-based approaches on mixed clusterings

Data Set	Base clusterings		EM		IVC		WKCC		WFC	
	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	0.6798	0.7140	0.6821	0.7156	0.6067	0.7033	0.5843	0.6864	**0.7163**	**0.7419**
Wine	0.8570	0.8380	0.7955	0.8098	0.6701	0.7328	0.6159	0.6656	**0.9149**	**0.8926**
Dermatology	0.6722	0.8236	0.7035	0.8380	0.6669	0.8018	0.6375	0.7914	**0.9190**	**0.9308**
Breast cancers	0.6575	0.5614	0.7003	0.5950	0.5213	0.4397	0.4308	0.3754	**0.7667**	**0.6466**
Digits	0.6134	0.7185	0.6449	0.7438	0.5877	0.7119	0.5708	0.7112	**0.7028**	**0.7761**
Statlog	0.5224	0.6049	0.5205	0.6079	0.5163	0.6022	0.5071	0.5932	**0.5344**	**0.6202**
Spambase	0.2298	0.1738	0.2983	0.2188	0.2346	0.1737	0.1534	0.1118	**0.3569**	**0.2601**
MNIST	0.3555	0.4917	0.3839	0.5024	0.3906	0.5055	0.3854	0.5078	**0.4090**	**0.5246**
KDDCUP99	0.4115	0.3613	0.4517	0.4074	0.3112	0.2806	0.2409	0.2173	**0.5019**	**0.4527**

TABLE 6
Comparisons of approaches based on different cluster-similarity matrices on mixed clusterings

Data Set	MCLA		SV		SWV		PTGP		WRC	
	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	0.7028	0.7277	0.4692	0.5217	0.5645	0.6822	0.6761	0.6945	**0.7163**	**0.7419**
Wine	0.8837	0.8650	0.7401	0.7246	0.7549	0.7352	0.8488	0.8342	**0.9325**	**0.9120**
Dermatology	0.7114	0.8761	0.1956	0.4872	0.3878	0.6365	0.5476	0.7363	**0.8706**	**0.9348**
Breast cancers	0.7667	0.6466	0.0128	0.0078	0.0128	0.0078	0.7696	0.6563	**0.7872**	**0.6721**
Digits	0.7107	0.7735	0.2461	0.2958	0.3793	0.4928	0.7214	0.7786	**0.7377**	**0.7932**
Statlog	0.5129	0.5742	0.2172	0.1895	0.4597	0.4592	0.5403	0.5948	**0.5324**	**0.6095**
Spambase	**0.3569**	**0.2601**	-0.0048	0.0101	-0.0048	0.0101	0.0053	0.0270	**0.3569**	**0.2601**
MNIST	0.4015	0.5113	0.1930	0.2425	0.1705	0.2611	0.4474	0.5383	**0.4552**	**0.5632**
KDDCUP99	**0.5019**	**0.4527**	0.0001	0.0002	0.0001	0.0002	0.2949	0.2439	**0.5019**	**0.4527**

TABLE 7
Comparisons of approaches based on different object-similarity matrices on mixed clusterings

Data Set	CSPA		HGPA		WCT-AL		WCE		WPC	
	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	0.7415	0.7437	0.2532	0.2779	0.7163	0.7419	0.6898	0.7151	**0.7707**	**0.7657**
Wine	0.7943	0.7859	0.3934	0.4980	0.8837	0.8650	0.9149	0.8926	**0.9149**	**0.8926**
Dermatology	0.6477	0.7569	0.5781	0.7271	0.8654	0.9302	0.8610	0.9249	**0.9293**	**0.9323**
Breast cancers	0.4155	0.4070	-0.0011	0.0001	0.7563	0.6341	0.7718	0.6528	**0.7872**	**0.6721**
Digits	0.6814	0.7399	0.2831	0.3976	0.6302	0.7456	0.6314	0.7446	**0.7153**	**0.7922**
Statlog	0.4517	0.5716	0.2630	0.3861	0.5277	0.6140	0.5253	0.6095	**0.5546**	**0.6449**
Spambase	**0.3710**	**0.3136**	-0.0002	0.0000	-0.0048	0.0101	0.3569	0.2601	**0.3710**	**0.3136**
MNIST	0.4050	0.4986	0.0001	0.0021	0.4144	0.5403	0.3850	0.5061	**0.4248**	**0.5454**
KDDCUP99	0.1520	0.2592	-0.0001	0.0000	0.5019	0.4527	0.4521	0.3956	**0.5019**	**0.4527**

includes the weighted feature consensus WFC, weighted relabeling consensus WRC and weighted pairwise-similarity consensus WPC algorithms. These algorithms use different strategies based on the proposed consensus measure to discover a high-quality final clustering. In the experimental analysis, we compared them with other cluster ensemble algorithms on nine real data sets. The comparison results have shown that the proposed algorithms are very effective and stable, compared to other algorithms.

ACKNOWLEDGEMENT

We are very grateful to the editors and reviewers for their valuable comments and suggestions. We also wish to thank the authors of the compared algorithms for sharing their codes.

REFERENCES

[1] J. Han, M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2001.

[2] A.K. Jain, R.C. Dubes, *Algorithms for Clustering Data*. Prentice Hall, 1988.

[3] A.K. Jain, Data clustering: 50 years beyond K-means, *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651-666, 2010.

[4] J.B. MacQueen, "Some methods for classification and analysis of multivariate observations". *Proc. of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, vol. 1, pp. 281-297, 1967.

[5] B. Scholkopf, A. J. Smola, and K.-R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," Neural Comput., vol. 10, no. 5, pp. 1299C1319, Jul. 1998.

[6] D.D. Lee, H.S. Seung, "Algorithms for non-negative matrix factorization," *International Conference on Neural Information Processing Systems*, MIT Press, pp. 535-541, 2000.

[7] J. Shi, J. Malik, "Normalized cuts and image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888C905, 2000.

[8] A.Y. Ng, M.I. Jordan, Y. Weiss, *On Spectral Clustering: Analysis and an Algorithm*, in: T.G. Dietterich, S. Becker, Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems, vol. 14, MIT Press, Cambridge, MA, 2002.

[9] A. Strehl, J. Ghosh, "Cluster ensembles: a knowledge reuse framework for combining multiple partitions", *Journal on Machine Learning Research*, vol. 3, pp. 583-617, 2002.

[10] A. Gionis, H. Mannila, P. Tsaparas, "Clustering aggregation, ACM Transactions on Knowledge Discovery from Data", vol. 1, no. 1, pp. 1–30, 2007.

TABLE 8
Comparisons of different feature-based approaches on k-means clusterings

Data Set	Rate	Base clusterings		EM		IVC		WKCC		WFC	
		ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	50%	0.5726	0.6646	0.5121	0.6002	0.5662	0.6568	0.4963	0.6017	**0.7163**	**0.7419**
	40%	0.5439	0.6492	0.4743	0.5875	0.4627	0.5589	0.5146	0.6074	**0.7163**	**0.7419**
	30%	0.5152	0.6337	0.4951	0.5972	0.5642	0.6261	0.4731	0.5429	**0.7163**	**0.7419**
Wine	50%	0.8594	0.8451	0.7864	0.7976	0.6568	0.7222	0.6347	0.7059	**0.8685**	**0.8529**
	40%	0.8572	0.8432	0.7538	0.7786	0.6194	0.6860	0.5144	0.5779	**0.8992**	**0.8782**
	30%	0.8535	0.8401	0.7134	0.7535	0.6620	0.7162	0.5485	0.6052	**0.8992**	**0.8782**
Dermatology	50%	0.6561	0.8181	0.6882	0.8324	0.7067	0.8296	0.5848	0.7666	**0.9293**	**0.9323**
	40%	0.6098	0.7968	0.6023	0.7994	0.5392	0.7497	0.6230	0.7896	**0.9293**	**0.9323**
	30%	0.5969	0.7905	0.5648	0.7677	0.7726	0.8576	0.5581	0.7557	**0.9293**	**0.9323**
Breast cancers	50%	0.3562	0.3373	0.7577	0.6508	0.4139	0.3914	0.5500	0.4723	**0.7815**	**0.6650**
	40%	0.2881	0.2882	0.5381	0.4772	0.3795	0.3445	0.4844	0.4449	**0.7815**	**0.6650**
	30%	0.2205	0.2345	0.2781	0.2553	0.3116	0.2974	0.4973	0.4655	**0.7815**	**0.6650**
Digits	50%	0.6496	0.7482	0.6437	0.7449	0.6037	0.7206	0.5248	0.6853	**0.7826**	**0.8128**
	40%	0.6396	0.7445	0.6255	0.7318	0.4906	0.6728	0.4950	0.6745	**0.7631**	**0.7831**
	30%	0.6277	0.7392	0.6094	0.7248	0.5508	0.6988	0.4945	0.6668	**0.7533**	**0.8037**
Statlog	50%	0.5452	0.6220	0.4683	0.5517	0.4538	0.5404	0.4579	0.5489	**0.5700**	**0.6318**
	40%	0.5404	0.6200	0.4967	0.5675	0.3975	0.5228	0.4032	0.5258	**0.5700**	**0.6318**
	30%	0.5372	0.6188	0.4910	0.5709	0.3990	0.5102	0.4046	0.5299	**0.5700**	**0.6318**
Spambase	50%	0.1760	0.1368	0.3205	0.2336	0.2143	0.1576	0.1570	0.1144	**0.3580**	**0.2610**
	40%	0.1398	0.1117	0.1771	0.1346	0.1774	0.1340	0.1570	0.1144	**0.3580**	**0.2610**
	30%	0.1035	0.0866	0.2446	0.1808	0.2864	0.2088	0.1781	0.1309	**0.3580**	**0.2610**
MNIST	50%	0.3751	0.4983	0.3606	0.4857	0.3767	0.4931	0.3593	0.4864	**0.4003**	**0.5212**
	40%	0.3708	0.4975	0.3629	0.4891	0.3683	0.4846	0.3435	0.4811	**0.3982**	**0.5199**
	30%	0.3696	0.4970	0.3662	0.4896	0.3729	0.4863	0.3482	0.4847	**0.3974**	**0.5159**
KDDCUP99	50%	0.2996	0.2761	0.4373	0.3959	0.2509	0.2263	0.2511	0.2263	**0.5019**	**0.4527**
	40%	0.2717	0.2517	0.3919	0.3518	0.3312	0.2988	0.2514	0.2264	**0.5019**	**0.4527**
	30%	0.2349	0.2196	0.3899	0.3512	0.3413	0.3078	0.2113	0.1901	**0.5019**	**0.4527**
Mean		0.4744	0.5337	0.5018	0.5519	0.4544	0.5148	0.4265	0.4971	**0.6568**	**0.6526**
Std		0.2173	0.2535	0.1635	0.2036	0.1553	0.2107	**0.1439**	**0.2043**	0.2001	0.2058

TABLE 9
Comparisons of approaches based on different cluster-similarity matrices on k-means clusterings

Data Set	Rate	MCLA		SV		SWV		PTGP		WRC	
		ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	50%	**0.7163**	**0.7419**	0.4290	0.5874	0.4290	0.5874	0.4499	0.5794	**0.7163**	**0.7419**
	40%	**0.7163**	**0.7419**	0.4290	0.5874	0.4290	0.5874	0.5504	0.5931	**0.7163**	**0.7419**
	30%	**0.7163**	**0.7419**	0.7163	0.7419	0.7163	0.7419	0.5272	0.5820	**0.7163**	**0.7419**
Wine	50%	**0.8685**	**0.8529**	**0.8685**	**0.8529**	**0.8685**	**0.8529**	0.7995	0.8164	**0.8685**	**0.8529**
	40%	0.8471	0.8347	0.8685	0.8529	0.8685	0.8529	0.8210	0.8416	**0.8992**	**0.8782**
	30%	0.8471	0.8347	0.8685	0.8529	0.8685	0.8529	0.7255	0.7372	**0.8992**	**0.8782**
Dermatology	50%	0.7132	0.8765	0.2669	0.5350	0.4449	0.6795	0.4717	0.6727	**0.9293**	**0.9323**
	40%	0.7112	0.8761	0.1038	0.1188	0.3207	0.5797	0.3788	0.6208	**0.8706**	**0.9348**
	30%	0.7157	0.8771	0.0396	0.3292	0.2360	0.5244	0.4016	0.6333	**0.8706**	**0.9348**
Breast cancers	50%	0.7553	0.6464	-0.0212	0.0075	-0.0341	0.0220	0.7220	0.6012	**0.7863**	**0.6767**
	40%	0.5793	0.4718	-0.0173	0.0034	-0.0247	0.0332	0.5999	0.4984	**0.7863**	**0.6767**
	30%	0.3550	0.3142	-0.0383	0.0549	-0.0151	0.0129	0.5016	0.4463	**0.7866**	**0.6716**
Digits	50%	0.6957	0.7620	0.1323	0.4043	0.5321	0.6516	0.6558	0.7391	**0.7471**	**0.7974**
	40%	0.6916	0.7585	0.0311	0.1462	0.4825	0.6322	0.6330	0.7250	**0.6915**	**0.7601**
	30%	0.6869	0.7523	0.2663	0.5448	0.4583	0.6649	0.6209	0.7144	**0.6520**	**0.7584**
Statlog	50%	0.5700	**0.6318**	**0.5951**	0.6139	0.4442	0.5592	0.4852	0.5582	0.5700	**0.6318**
	40%	0.5208	0.6093	0.4300	0.5202	0.5125	0.6226	0.4704	0.5540	**0.5700**	**0.6318**
	30%	0.5222	0.6111	0.5184	0.5928	0.4823	0.5714	0.4539	0.5445	**0.5700**	**0.6318**
Spambase	50%	**0.3580**	**0.2610**	-0.0048	0.0101	-0.0048	0.0101	0.0989	0.0759	**0.3580**	**0.2610**
	40%	0.0320	0.0804	-0.0048	0.0101	-0.0048	0.0101	0.1669	0.2265	**0.3580**	**0.2610**
	30%	-0.0048	0.0101	-0.0159	0.0346	-0.0159	0.0346	0.1340	0.1752	**0.3580**	**0.2610**
MNIST	50%	0.3874	0.5002	0.1343	0.1912	0.2566	0.3803	0.4124	0.5159	**0.4218**	**0.5293**
	40%	0.3877	0.5045	0.1062	0.1424	0.1938	0.2944	0.4000	0.5108	**0.4212**	**0.5282**
	30%	0.3823	0.5003	0.1331	0.1665	0.2159	0.3409	0.3982	0.5113	**0.4128**	**0.5200**
KDDCUP99	50%	**0.5019**	**0.4527**	-0.0191	0.0075	-0.0157	0.0087	0.4766	0.3832	**0.5019**	**0.4527**
	40%	**0.5019**	**0.4527**	-0.0191	0.0075	-0.0157	0.0087	0.4226	0.3440	**0.5019**	**0.4527**
	30%	-0.0195	0.0077	-0.0191	0.0075	-0.0157	0.0087	0.4260	0.3435	**0.5019**	**0.4527**
Mean		0.5252	0.5817	0.2510	0.3305	0.3190	0.4121	0.4890	0.5387	**0.6475**	**0.6515**
Std		0.2579	0.2622	0.3085	0.3071	0.2990	0.3135	**0.1793**	**0.1876**	0.1890	0.2036

TABLE 10
Comparisons of approaches based on different object-similarity matrices on k-means clusterings

Data Set	Rate	CSPA		HGPA		WCT-AL		WCE		WPC	
		ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	50%	0.6756	0.6963	0.4489	0.5217	0.5609	0.7098	0.5525	0.6537	**0.7163**	**0.7419**
	40%	0.6005	0.6461	0.4379	0.5114	0.5609	0.7098	0.5525	0.6537	**0.7163**	**0.7419**
	30%	0.6005	0.6461	0.4755	0.5483	0.4290	0.5874	0.4290	0.5874	**0.7163**	**0.7419**
Wine	50%	0.7808	0.7867	0.3918	0.4251	0.4667	0.6118	0.8471	0.8347	**0.9149**	**0.8926**
	40%	0.7808	0.7867	0.1024	0.1800	0.8471	0.8347	0.8471	0.8347	**0.9149**	**0.8926**
	30%	0.7808	0.7771	0.0938	0.1518	0.8471	0.8347	0.8471	0.8347	**0.9149**	**0.8926**
Dermatology	50%	0.6557	0.7753	0.5337	0.6783	0.8654	0.9302	0.5806	0.7583	**0.9293**	**0.9323**
	40%	0.6294	0.7484	0.5439	0.6371	0.8654	0.9302	0.5806	0.7583	**0.8654**	**0.9302**
	30%	0.6557	0.7753	0.3919	0.6055	0.8654	0.9302	0.5806	0.7583	**0.8654**	**0.9302**
Breast cancers	50%	0.3516	0.3308	-0.0011	0.0001	0.6698	0.5750	0.6698	0.5750	**0.7863**	**0.6767**
	40%	0.3516	0.3308	-0.0011	0.0001	0.6698	0.5750	0.7815	0.6650	**0.7863**	**0.6767**
	30%	0.3315	0.3086	-0.0011	0.0001	0.7813	0.6679	0.7765	0.6538	**0.7863**	**0.6767**
Digits	50%	0.6552	0.7181	0.3592	0.4800	0.6142	0.7325	0.6452	0.7496	**0.6917**	**0.7733**
	40%	0.6809	0.7364	0.2159	0.3220	0.6101	0.7271	0.6112	0.7281	**0.6880**	**0.7721**
	30%	0.6841	0.7396	0.4422	0.5721	0.6106	0.7280	0.6092	0.7258	**0.6938**	**0.7730**
Statlog	50%	0.4107	0.5155	0.5016	0.5606	0.5252	0.6144	0.5179	0.6012	**0.5700**	**0.6318**
	40%	0.4065	0.5076	0.3924	0.4604	0.5230	0.6110	0.5249	0.6120	**0.5700**	**0.6318**
	30%	0.4001	0.4931	0.3942	0.4607	0.5298	0.6161	0.5269	0.6140	**0.5700**	**0.6318**
Spambase	50%	**0.3710**	**0.3136**	-0.0002	0.0000	-0.0048	0.0101	0.3470	0.2556	0.3580	0.2610
	40%	**0.3795**	**0.3218**	-0.0002	0.0000	-0.0048	0.0101	-0.0048	0.0101	0.3580	0.2610
	30%	**0.3942**	**0.4607**	-0.0002	0.0000	-0.0048	0.0101	-0.0048	0.0101	0.3580	0.2610
MNIST	50%	0.3595	0.4635	0.0001	0.0021	0.4050	0.5236	0.3850	0.5061	**0.4042**	**0.5147**
	40%	0.3261	0.4527	0.0001	0.0021	0.3851	0.5042	0.3582	0.4898	**0.4039**	**0.5137**
	30%	0.3508	0.4638	0.0001	0.0021	0.3887	0.5115	0.3454	0.4704	**0.3938**	**0.5123**
KDDCUP99	50%	0.0000	0.0000	-0.0001	0.0000	-0.0187	0.0073	0.3167	0.2641	**0.5019**	**0.4527**
	40%	0.0000	0.0000	-0.0001	0.0000	-0.0187	0.0073	0.2935	0.2574	**0.5019**	**0.4527**
	30%	0.0000	0.0000	-0.0001	0.0000	-0.0044	0.0001	-0.0187	0.0073	**0.5019**	**0.4527**
Mean		0.4672	0.5109	0.2119	0.2638	0.4802	0.5374	0.4999	0.5507	**0.6473**	**0.6527**
Std		0.2302	0.2489	0.2189	0.2649	0.3046	0.3127	0.2448	0.2540	**0.1944**	**0.2069**

[11] Z. Zhou. *Ensemble Methods: Foundations and Algorithms*, CRC Press, 2012.

[12] E. Gonzlez, J. Turmo, "Unsupervised ensemble minority clustering", *Machine Learning*, 98: 217–268, 2015.

[13] P. Zhou, L. Du, L. Shi, H. Wang et al., "Learning a robust consensus matrix for clustering ensemble via kullback-leibler divergence minimization", *Proc. the 25th International Joint Conference on Artificial Intelligence*, 2015.

[14] A. Topchy, A. Jain, W. Punch, "Clustering ensembles: Models of consensus and weak partitions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, 1866-1881, 2005.

[15] N. Iam-On, T. Boongoen, "Comparative study of matrix refinement approaches for ensemble clustering", *Machine Learning*, 98: 269-300, 2015.

[16] Z. Zhou, W. Tang, "Clusterer ensemble", *Knowledge-Based Systems*, vol. 19, no. 1, pp. 77–83, 2006.

[17] Y. Yang, K. Chen, "Temporal data clustering via weighted clustering ensemble with different representations", *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 2, pp. 307-320, 2011.

[18] M. C. Naldi, A. Carvalho, R. Campello, "Cluster ensemble selection based on relative validity indexes", *Data Mining and Knowledge Discovery*, vol. 27, pp. 259C289, 2013.

[19] A. Fred, A. Jain, "Combining multiple clusterings using evidence accumulation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 6, pp.835-850, 2005.

[20] L. Kuncheva, D. Vetrov, "Evaluation of stability of k-means cluster ensembles with respect to random initialization", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 11, pp. 1798-1808, 2006.

[21] X. Zhang, L. Jiao, F. Liu, L. Bo, M. Gong. "Spectral clustering ensemble applied to SAR image segmentation", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 7, pp. 2126-2136, 2008.

[22] M. Law, A. Topchy, A. Jain, "Multiobjective data clustering", *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2004.

[23] Z. Yu, H. Chen, J. You, et al, "Hybrid fuzzy cluster ensemble framework for tumor clustering from bio-molecular data", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 3, pp. 657–670, 2013.

[24] B. Fischer, J. Buhmann, "Bagging for path-based clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 11, pp. 1411-1415, 2003.

[25] A. Topchy, B. Minaei-Bidgoli, A. Jain, "Adaptive clustering ensembles", *Proc. the 17th International Conference on Pattern Recognition*, 2004.

[26] Y. Hong, S. Kwong, H. Wang, Q. Ren, "Resampling-based selective clustering ensembles", *Pattern Recognition Letters*, 2009, 41(9):2742C2756.

[27] X. Fern, C. Brodley, "Random projection for high dimensional data clustering: A cluster ensemble approach", *Proc. International Conference on Machine Learning*, 2003.

[28] Z. Yu, L. Li, J. Liu et al., "Adaptive noise immune cluster ensemble using affinity propagation", *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 19, pp. 3176-3189, 2015.

[29] F. Gullo, C. Domeniconi, "Metacluster-based projective clustering ensembles", *Machine Learning*, vol. 98, no. 1-2, pp. 1-36, 2013.

[30] Y. Yang, J. Jiang, "Hybrid Sampling-Based Clustering Ensemble With Global and Local Constitutions", *Ieee Transactions on Neural Networks and Learning Systems*, vol. 27, no. 5, pp. 952-965, 2016.

[31] A. Fred, A. K. Jain, "Data clustering using evidence accumulation", *Proc. the 16th International Conference on Pattern Recognition*, pp. 276-280, 2002.

[32] N. Iam-On, T. Boongoen, S. Garrett, C. Price, "A link-based approach to the cluster ensemble problem", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2396-2409, 2011.

[33] N. Iam-On, T. Boongoen, S. Garrett, C. Price, "A link-based cluster ensemble approach for categorical data clustering", *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 3, pp. 413-425, 2012.

[34] Z. Yu, H. Chen, J. You, J. Liu, H. Wong, G. Han, L. Li, "Adaptive fuzzy consensus clustering framework for clustering analysis of cancer data", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 12, no. 4, pp. 887–901, 2015.

[35] X. Fern, C. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning", *Proc. of the 21st International Conference on Machine Learning*, 2004.

[36] D. Huang, J. Lai, C. D. Wang, "Robust ensemble clustering using

probability trajectories", *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, pp. 1312-1326, 2016.
[37] M. Selim, E. Ertunc, "Combining multiple clusterings using similarity graph", *Pattern Recognition*, vol. 44, no. 3, 694-703, 2011.
[38] C. Boulis, M. Ostendorf, "Combining multiple clustering systems", *Proc. European Conf. Principles and Practice of Knowledge Discovery in Databases*, 2004.
[39] P. Hore, L. O. Hall , B. Goldgo, "A scalable framework for cluster ensembles", *Pattern Recognition*, vol. 42, no. 5, 676-688, 2009.
[40] B. Long, Z. Zhang, P. S. Yu, "Combining multiple clusterings by soft correspondence", *Proc. the 4th IEEE International Conference on Data Mining*, 2005.
[41] P. Rathore, J.C. Bezdek, S.M. Erfani, S. Rajasegarar; M. Palaniswami, "Ensemble Fuzzy Clustering using Cumulative Aggregation on Random Projections", *IEEE Transactions on Fuzzy Systems*, DOI: 10.1109/TFUZZ.2017.2729501, 2017.
[42] D. Cristofor, D. Simovici, "Finding median partitions using information theoretical based genetic algorithms", *J. Universal Computer Science*, vol. 8, no. 2, pp. 153–172, 2002.
[43] H. Wang, H. Shan, A. Banerjee, "Bayesian cluster ensembles", *Statistical Analysis and Data Mining*, vol. 4, no. 1, pp. 54-70, 2011.
[44] Z. He, X. Xu, S. Deng, "A cluster ensemble method for clustering categorical data", *Information Fusion*, vol. 6, no. 2, pp. 143C151, 2005.
[45] N. Nguyen, R. Caruana, "Consensus Clusterings", *Proc. IEEE Intl Conf. Data Mining*, pp. 607–612, 2007.
[46] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values", *Data Mining and Knowledge Discovery*, vol. 2, no. 3, pp. 283-304, 1998.
[47] H. Liu, J. Wu, T. Liu, D. Tao, Y. Fu, "Spectral ensemble clustering via weighted k-means: theoretical and practical evidence", *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1129-1143, 2017.
[48] J. Wu, H. Liu, H. Xiong, J. Cao, "A theoretic framework of k-means-based consensus clustering", *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, pp. 799-1805, 2013.
[49] T.M. Cover, J.A. Thomas, *Elements of Information Theory*, Wiley, 2006.
[50] E. Gokcay, J.C. Principe, "Information theoretic clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 158-171, 2002.
[51] N.B. Karayiannis, "MECA: Maximum entropy clustering algorithm", *In: Proc IEEE Int Conf Fuzzy Syst*, Orlando, pp. 630C635, 1994.
[52] D. Barbara, Y. Li, and J. Couto, "Coolcat: An entropy-based algorithm for categorical clustering," *in Proceedings of the Eleventh International Conference on Information and Knowledge Management*, 2002, pp. 582C589.
[53] K. Chen, L. Liu, "The best k for entropy-based categorical clustering", *In: Proceedings of international conference on scientific and statistical database management (SSDBM)*, pp. 253C262.
[54] L. Bai, J. Liang, "Cluster validity functions for categorical data: a solution-space perspective," *Data Mining and Knowledge Discovery*, vol. 29, pp. 1560-1597, 2015.
[55] M. Rosvall, C. T. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks," *The National Academy of Sciences of the USA*, vol. 104, no. 18, 7327C7331, 2007.
[56] L. Bai, X. Cheng, J. Liang, Y. Guo, "Fast graph clustering with a new description model for community detection,", *Information Sciences*, vol. 388C389, pp. 37C47, 2017.
[57] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846C-850, 1971.
[58] T. S. A. V. W. T. Press, W. H. and B. P. Flannery, *Conditional Entropy and Mutual Information*. Numerical Recipes: The Art of Scientific Computing (3rd ed.) . New York: Cambridge University Press., 2007.

Liang Bai received his Ph.D degree in Computer Science from Shanxi University in 2012. He worked for the postdoctoral research in the institute of Computing Technology, Chinese Academy of Sciences. He is currently an Associate Professor with the School of Computer and Information Technology, Shanxi University. His research interest is cluster analysis. He has published several journal papers in his research fields, including IEEE TPAMI, IEEE TKDE, IEEE TFS, Data Mining and Knowledge Discovery.

Jiye Liang received the PhD degree in Applied Mathematics from Xi'an Jiaotong University, in 2001. He is a professor in the School of Computer and Information Technology, Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University. His research interests include artificial intelligence, granular computing, data mining, and machine learning. He has published more than 100 articles in international journals.

Hangyuan Du received the B.S. degree from Heilongjiang Institute of Technology, in 2008, and the Ph.D. degree in Control Science and Engineering from Harbin Engineering University, in 2012. He is currently a member of School of Computer and Information Technology at Shanxi University, as a lecturer. His research interests lie in the areas of cluster analysis.

Yike Guo received the PhD degree in logic and declarative programming from Imperial College, University of London, in 1993. He is now a professor in computing science in the Department of Computing, Imperial College, University of London. His research is in large scale scientific data analysis, data mining algorithms and applications, parallel algorithms, and cloud computing.