J Inf Process Syst, Vol.13, No.6, pp.1467~1486, December 2017
https://doi.org/10.3745/JIPS.01.0019

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Mixed Co-clustering Algorithm Based on Information Bottleneck

Yongli Liu*, Tianyi Duan*, Xing Wan*, and Hao Chao*

Abstract

Fuzzy co-clustering is sensitive to noise data. To overcome this noise sensitivity defect, possibilistic clustering relaxes the constraints in FCM-type fuzzy (co-)clustering. In this paper, we introduce a new possibilistic fuzzy co-clustering algorithm based on information bottleneck (ibPFCC). This algorithm combines fuzzy co-clustering and possibilistic clustering, and formulates an objective function which includes a distance function that employs information bottleneck theory to measure the distance between feature data point and feature cluster centroid. Many experiments were conducted on three datasets and one artificial dataset. Experimental results show that ibPFCC is better than such prominent fuzzy (co-)clustering algorithms as FCM, FCCM, RFCC and FCCI, in terms of accuracy and robustness.

Keywords

Co-clustering, F-Measure, Fuzzy Clustering, Information Bottleneck, Objective Function

1. Introduction

In order to keep up with the tremendous growth of information, many data mining techniques, aiming at revealing and visualizing structure of data, were studied and used to help end users find the information they required. One of these techniques is clustering, which tries to identify clusters that exhibit high intra-cluster similarity and low inter-cluster similarity [1,2].

Any clustering technique relies on two crucial components: a clustering algorithm and a similarity measure [1]. To introduce a clustering algorithm in detail, let us consider a set of *N* objects, denoted by $X = \{x_1, x_2, \cdots, x_N\} \subset \mathbb{R}^K$, where each object is a numerical feature vector that describes the objects' attributes, and *K* is the dimension of the feature space. Clustering tries to partition *X* into *C* (1<*C*<*N*) subgroups such that each subgroup represents "natural" substructure in *X* [3]. There are three main types of clustering algorithms according to different division standards: crisp clustering, fuzzy clustering and possibilistic clustering [4]. Crisp clustering is, in actuality, hard clustering, that puts each object into a single cluster. Let u_{ci} be the membership of object x_i in cluster *c*, and the partition element will equal 1 if x_i belongs to *c* and equal 0 otherwise. In other words, the value of u_{ci} can only be 0 or 1.

Different from crisp clustering, fuzzy clustering allows an object to belong to more than one cluster

Manuscript received January 6, 2017; accepted March 4, 2017.

Corresponding Author: Yongli Liu (yongli.buaa@gmail.com)
* School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo, Henan, China (yongli.buaa@gmail.com, dtyhpu@sina.com, wxxiaofanke@163.com, chaohao@hpu.edu.cn)

[5-7]. Here the u_{ci} is usually interpreted as a probability $p(c|x_i)$ that x_i is in the i-th class. Therefore, the range of u_{ci} is extended from {0, 1} in crisp clustering to [0, 1]. On u_{ci}, there is an important constraint, $\sum_c u_{ci} = 1$, which ensures that every object is involved in each cluster. Because the indisputable fact is that any data object may involve multiple topics, fuzzy clustering is more realistic than crisp clustering. Fuzzy C-means (FCM) [8], known as the fuzzy version of the K-means algorithm, is a representative fuzzy clustering algorithm. In FCM, the affiliations of objects to clusters are represented by memberships. Fuzzy co-clustering extends fuzzy clustering by assigning both objects and features membership functions. It filters out relevant features during the computation of object membership function and thus solves the problem of sparseness of data by reducing the dimensionality [9]. The co-clustering algorithm is thus suited to Web applications with high dimensions. At present popular fuzzy co-clustering algorithms include fuzzy clustering for categorical multivariate data (FCCM) [10], robust fuzzy co-clustering (RFCC) [11] and fuzzy co-clustering algorithm for images (FCCI) [9], etc. However, fuzzy co-clustering algorithms also apply the constraint that the memberships of each object across groups sum to 1 [12]. Due to this constraint, fuzzy co-clustering has considerable trouble in handling outliers in a data set.

To address this problem, possibilistic clustering relaxes the constraint, and the sum of each column satisfies the looser constraint, $0 < \sum_c u_{ci} \le C$. That is, the value of u_{ci} can be any number between 0 and 1. Krishnapuram and Keller [13] proposed the well-known possibilistic c-means clustering algorithm (PCM), and suggested that the u_{ci} should be interpreted as the typicality of x_i relative to cluster c rather than membership. Because of the looser constraint, PCM is not as sensitive to outlier as FCM. However, PCM is very sensitive to data initializations and sometimes generates coincident clusters [14]. Pal et al. [3] proposed a fuzzy possibilistic c-means clustering algorithm (FPCM), which possess the good features of both FCM and PCM, while eliminating some of their bad features.

Besides a clustering algorithm mentioned above, a similarity measure is also an important component of a clustering technique. There exist many similarity measures, such as the Cosine measure, the Dice measure, the Jaccard measure, the overlap measure and the information-theoretic measure [15]. We all know that selecting a similarity measure to evaluate similarities of objects has a significant impact on clustering and final results [16]. However, the choice of similarity measures has no clear uniform standards, and too often this is an arbitrary choice, which may significantly affect the clustering accuracy. Slonim and Tishby [17] introduced information bottleneck into clustering. Different from common similarity measures, the information bottleneck based clustering algorithms group objects by calculating mutual information loss when merging objects into a cluster. The results [17,18] showed that their algorithms perform much better.

Currently, the Web is the largest information repository. The huge size motivates us to present a new clustering algorithm that could easily deal with large-scale and high-dimensional data and provide high clustering quality. Fuzzy co-clustering can group high-dimensional sparse data by reducing data dimensionality. And its noise sensitivity defect could be overcome by possibilistic clustering which relaxes the constraint in FCM-type fuzzy clustering. Further, we believe that information bottleneck based similarity measure could help to improve clustering accuracy.

Therefore, in this paper, we propose a new possibilistic fuzzy co-clustering algorithm based on information bottleneck (ibPFCC). This algorithm integrates possibilistic clustering, fuzzy clustering and co-clustering, and strives to keep their benefits. Furthermore, it introduces information bottleneck based similarity measure into its objective function, which could help to reduce the subjective error caused by arbitrary choice on similarity measures and improve clustering quality.

The remainder of this paper is organized as follows. In Section 2, we provide a literature review of fuzzy co-clustering, possibilistic clustering and information bottleneck theory. Section 3 introduces in detail the proposed algorithm, ibPFCC. In Section 4, some experiments are performed and experimental results are discussed. Finally, we conclude our work.

2. Related Work

In this section, we briefly review some related clustering algorithms, including fuzzy co-clustering, possibilistic clustering, and information bottleneck theory. This review section can help to understand our algorithm introduced in the next section. The explanations on the mathematical notations used in this paper are listed in Table 1.

Table 1. List of mathematical notations

Notation	Description
C,N,K	Numbers of clusters, documents, and words
u_{ci}	Document partitioning membership
v_{cj}	Word ranking membership
t_{ci}	Document typicality membership
T_u, T_v, T_t	User-defined membership parameters
$Dist(\ldots,\ldots)$	Distance function
p_{cj}	Feature cluster centroid
D_{cij}	Distance between feature point x_{ij} and feature cluster centroid p_{cj}
x_i	Data point
p_c	Cluster centroid
τ	Number of iterations
τ_{max}	Maximum number of iterations
ε	Convergence indicator parameter
λ_i, γ_c, β_c	Lagrange multipliers

2.1 Fuzzy Co-clustering

Most of classic clustering algorithms belong to crisp clustering, where the memberships are all equal to 0 or 1. Fuzzy clustering is more flexible than crisp clustering because it allows each object belong to more than one cluster [4].

FCM is the most influential fuzzy clustering algorithm, which adds fuzzy theory into *K*-means clustering. Different from the crisp *K*-means, in FCM, the objective function is extended from crisp partition to fuzzy partition, and the distance between the objects and the cluster centers is weighted by the membership degree. However, FCM, after all, is a one-dimensional clustering algorithm, which only considers the correlation between objects and neglects the correlation between features. In 2001, Oh et al. [10] extended FCM from one-dimensional to two-dimensional, and proposed a fuzzy co-clustering algorithm, FCCM. In FCCM, both objects and features are assigned membership functions, and documents and words are grouped simultaneously. Besides FCCM, The FCCI [9] is also a fuzzy co-clustering algorithm, whose objective function includes a multi-dimensional distance function as the dissimilarity measure and the entropy as the regularization term. The objective function of FCCI is given as Eq. (1), which is required to be minimized subject to the membership constraints in Eqs. (2) and (3).

$$\begin{aligned} J_{FCCI}(U,V,P) = \sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K} u_{ci}v_{cj}D_{cij} + T_u\sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci} \\ + T_v\sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj} \end{aligned} \tag{1}$$

$$\sum_{c=1}^{C} u_{ci} = 1, u_{ci} \in [0,1], \forall i = 1,\ldots N \tag{2}$$

$$\sum_{j=1}^{K} v_{cj} = 1, v_{cj} \in [0,1], \forall c = 1,\ldots C \tag{3}$$

The first term in Eq. (1) denotes the effective squared distance. The u_{ci} denotes the object membership of the *i*-th data object to cluster *c*, and the v_{cj} is the feature membership defined as the membership of the *j*-th feature to the *c*-th cluster. The D_{cij} is the square of the Euclidean distance between feature data point x_{ij} and the feature cluster centroid p_{cj}. In order to minimize this term, it is necessary to assign higher membership values to the objects nearer to cluster centers, and higher weights to the features that are more relevant. The second and third terms try to maximize the fuzzy entropies $-\sum_{c=1}^{C}\sum_{i=1}^{N} u_{ci}\log u_{ci}$ and $-\sum_{c=1}^{C}\sum_{j=1}^{K} v_{cj}\log v_{cj}$. The T_u and T_v control the degrees of partition fuzziness respectively, and the larger the values the fuzzier the partition.

Eq. (1) can be minimized by alternatively updating the following membership equations until convergence is achieved:

$$u_{ci} = \frac{exp\{-\sum_{j=1}^{K} v_{cj}D_{cij} / Tu\}}{\sum_{c=1}^{C} exp\{-\sum_{j=1}^{K} v_{cj}D_{cij} / Tu\}} \tag{4}$$

$$v_{cj} = \frac{\exp\{-\sum_{i=1}^{N} u_{ci}D_{cij} / T_v\}}{\sum_{j=1}^{K} \exp\{-\sum_{i=1}^{N} u_{ci}D_{cij} / T_v\}} \tag{5}$$

$$p_{cj} = \frac{\sum_{i=1}^{N} u_{ci}x_{ij}}{\sum_{i=1}^{N} u_{ci}} \tag{6}$$

2.2 Possibilistic Clustering

In FCM-type clustering, for each object, the sum of the membership degrees in the clusters must be equal to 1 [10]. Because of this constraint, fuzzy clustering may cause some meaningless clustering results, especially when noise is present. Fig. 1 illustrates the limitation.

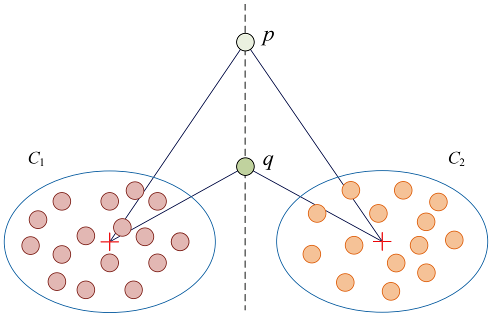


Fig. 1. Limitation of FCM-type algorithms.

Suppose there are two clusters c_1 and c_2, and two objects p and q that are both equidistant from cluster centers. The membership values of p and q in each cluster will be all equal to 0.5, which contradicts our intuition. We intuitively think that the object q should have higher membership value than the object p, because q is physically much closer to the two clusters than p, although these two objects are right in the middle of the two clusters. It should be evident that the constraints in FCM-type clustering only attach importance to the values of membership, and disregard the absolute distance value of each object from two centroids. Therefore, if there are noisy objects, fuzzy clustering will struggle to provide high quality clustering results.

In 1993, Krishnapuram and Keller [13] proposed the PCM algorithm. After that, much research work on PCM was conducted, because PCM could address the drawbacks associated with the constrained memberships used in FCM and be especially effective when there are outliers. Recently, there is more and more work that combines fuzzy clustering and possibilistic clustering. Pal et al. [3] proposed a new model called fuzzy-possibilistic c-means (FPCM), which simultaneously produces both memberships and possibilities, along with the usual point prototypes or cluster centers for each cluster. The FPCM solves the noise sensitivity defect of FCM, and also overcomes the coincident clusters problem of PCM. Abidi and Yahia [19] introduced a fuzzy possibilistic clustering algorithm, called PFKCN, based on neural network. This algorithm introduces both membership and typicality values, simultaneously, into the Kohonen Network clustering. Duraisamy and Haridass [20] developed a modified fuzzy possibilistic clustering algorithm based on FPCM to obtain better quality clustering results.

In this section, FPCM, the classic fuzzy possibilistic clustering algorithm, will be introduced. This approach tries to minimize the following objective function:

$$J_{FPCM}=\sum_{c=1}^{C}\sum_{i=1}^{N}\left(u_{ci}^{m}+t_{ci}^{\eta}\right)Dist(x_i,p_c) \tag{7}$$

In order to explain the typical degree of partitioning memberships, FPCM relaxes the restriction of partitioning memberships by introducing object typicality membership t_{ci}. This membership t_{ci} measures the typicality between the i-th object and the cluster c, relative to the similarities between all the other objects and the cluster *c*. The two parameters, *m* and *η*, are used to control the degree of fuzziness of partition respectively. When we minimize the objective function of the FPCM, two constraints, listed as the Eq. (2) and Eq. (8), are required to be satisfied.

$$\sum_{i=1}^{N} t_{ci} = 1, t_{ci} \in [0,1], \forall c = 1,\ldots C \tag{8}$$

If $Dist(x_i, p_c)$ is the squared Euclidean norm, the minimization of Eq. (7) can be solved by alternatively updating Eqs. (9) to (11) until the convergence is achieved:

$$u_{ci} = \left(\sum_{f=1}^{C} \left(\frac{Dist(x_i, p_c)}{Dist(x_i, p_f)} \right)^{\frac{2}{m-1}} \right)^{-1} \tag{9}$$

$$t_{ci} = \left(\sum_{f=1}^{N} \left(\frac{Dist(x_i, p_c)}{Dist(x_f, p_c)} \right)^{\frac{2}{\eta-1}} \right)^{-1} \tag{10}$$

$$p_{cj} = \frac{\sum_{i=1}^{N} \left(u_{ci}^{m} + t_{ci}^{\eta} \right) x_{ij}}{\sum_{i=1}^{N} \left(u_{ci}^{m} + t_{ci}^{\eta} \right)} \tag{11}$$

2.3 Information Bottleneck Theory

Information bottleneck theory [11] originated from Shannon's information theory. Given joint probability distribution *P*(*X, Y*), we try to search for a compact representation of *X*, under the condition of preserving the maximum information of *Y*. Thus, the information that *X* contains about *Y* is squeezed through a compact "bottleneck". In the past few years, information bottleneck theory has been frequently used in clustering. The desired clustering is the one that minimizes the loss of mutual information between the objects and the features extracted from them [21]. At the beginning of clustering, each object is regarded as a cluster. In the subsequent steps, some adjacent clusters need to be merged. And the merge process will produce mutual information loss. In order to minimize the mutual information loss in the whole clustering process, a greedy agglomeration manner is usually adopted, which merges two clusters that cause the minimal mutual information loss in each step.

The loss of mutual information of two clusters, c_x and c_y, is denoted as $D(c_x, c_y)$ and calculated based on information theory as,

$$\begin{aligned} D(c_x,c_y) &= I(C_{before},Y) - I(C_{after},Y) \\ &= \sum_y p(c_x,y)\log\frac{p(c_x,y)}{p(c_x)p(y)} + \sum_y p(c_y,y)\log\frac{p(c_y,y)}{p(c_y)p(y)} \\ &- \sum_y p(c_x \cup c_y,y)\log\frac{p(c_x \cup c_y,y)}{p(c_x \cup c_y)p(y)} \\ &= \sum_y p(c_x,y)\log\frac{p(y|c_x)}{\mathrm{p}(y|\mathrm{c}_x \cup \mathrm{c}_y)} + \sum_y p(c_y,y)\log\frac{p(y|c_y)}{\mathrm{p}(y|\mathrm{c}_x \cup \mathrm{c}_y)} \end{aligned} \tag{12}$$

where $I(C_{before},Y)$ and $I(C_{after},Y)$ are the mutual information between the cluster and the feature space before and after the cluster merging two clusters c_x and c_y, respectively.

The loss of mutual information between the feature data point x_{ij} and the feature cluster centroid p_{cj} is given by.

$$D_{cij} = \frac{|x_i|}{N} x_{ij} \log \frac{x_{ij}}{t} + \frac{|c_c|}{N} p_{cj} \log \frac{p_{cj}}{t} \tag{13}$$

where $t = |x_i| * x_{ij} / |x_i \cup c_c| + |c_c| * p_{cj} / |x_i \cup c_c|$, N represents the total number of objects, $|\cdot|$ is the number of objects in the cluster, and $x_i \cup c_c$ is the cluster merging the data x_i and the cluster c_c, whose centroids are denoted as x_i and p_c respectively.

Information bottleneck based clustering shows much better clustering quality than conventional clustering algorithms [17,18,21,22]. Many clustering experiments show that information bottleneck based similarity measure, which use joint probability model and mutual information, can better represent the correlation between objects and features. Slonim and Tishby [17] proposed an agglomerative hierarchical clustering algorithm based on information bottleneck theory, AIB. Experimental results show that the AIB algorithm based on information bottleneck is very effective. The average performance over all datasets attained 0.55 accuracy, while the second best result was only 0.47 accuracy.

3. The ibPFCC Algorithm

Co-clustering can offer several benefits [23] including (1) dimensionality reduction, (2) interpretable document cluster, and (3) improvement in accuracy due to local model on clustering. Fuzzy co-clustering extends co-clustering by adding fuzzy sets theory, and could generate co-clusters that are more realistic [2]. Fuzzy co-clustering is prone to achieve better performance than standard clustering. However, this technique usually suffers from the presence of outliers. The root of the problem lies in the membership constraint. Possibilistic clustering overcomes this problem by relaxing this constraint. Besides, since the similarity measure is very crucial to clustering, it is inappropriate to select a similarity measure arbitrarily. Information bottleneck theory keeps as much as information in clustering, and proves to be able to achieve higher clustering quality. Therefore, in this paper, we propose an information bottleneck based Possibilistic Fuzzy Co-Clustering algorithm, called ibPFCC. This algorithm will have the following advantages:

- It is a hybrid of fuzzy clustering and possibilistic clustering.
- It could minimize the impact of outliers to improve the accuracy of co-clustering.
- Its objective function contains the distance function based on the information bottleneck.
- This algorithm should keep such benefits derived from fuzzy co-clustering, as dimensionality reduction, interpretable document cluster.

3.1 The Objective Function

The ibPFCC employs the distance function based on information bottleneck to measure the degree of correlation between objects. Its clustering process is carried out towards the direction where minimum mutual information loss is generated. Therefore, the objective function of ibPFCC is designed as Eq. (14), which will be minimized subject to the document partitioning membership constraint as Eq. (2), the document typicality membership constraint as Eq. (8), and the word ranking membership constraint as Eq. (3).

$$\begin{aligned} J_{ibPFCC} = & \sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K}(u_{ci}+t_{ci})*v_{cj}D_{cij} \\ & +T_u\sum_{c=1}^{C}\sum_{i=1}^{N}u_{ci}\ln u_{ci}+T_t\sum_{c=1}^{C}\sum_{i=1}^{N}t_{ci}\ln t_{ci} \\ & +T_v\sum_{c=1}^{C}\sum_{j=1}^{K}v_{cj}\ln v_{cj} \end{aligned} \tag{14}$$

The objective function has three types of membership: the document possibilistic membership u_{ci}, the document typicality membership t_{ci}, and the word ranking membership v_{cj}. The first term of Eq. (14) is degree of aggregation which originates, but is different from FPCM. It is evident that this term contains both membership and typicality values for each object across all the clusters. D_{cij} is the amount of mutual information loss calculated by Eq. (13). The three remaining terms are nonlinear regularization terms which are downward convex functions, making u_{ci}, t_{ci} and v_{cj} to be fuzzy. The T_u， T_v and T_t adjust the degree of fuzziness in clustering, and the larger these parameters, the fuzzier the partition.

The constraint in Eq. (8) is different from the mathematical representations of fuzzy clustering. This constraint does not require each column to sum to 1, $\sum_{c=1}^{C}t_{ci}=1$, but each row to sum to 1, $\sum_{i=1}^{N}t_{ci}=1, t_{ci}\in[0,1]$, which means $\sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K}t_{ci}*v_{cj}D_{cij}$ in the objective function is a possibilistic term. This term can distribute the possibility values with respect to all N objects, but not to all C clusters.

3.2 The Update Equations

The constrained optimization of ibPFCC can be solved by applying the Lagrange multipliers λ, γ and β to constraints in Eqs. (2), (3) and (8) respectively.

$$\begin{aligned} J = & \sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K}(u_{ci}+t_{ci})*v_{cj}D_{cij}+T_u\sum_{c=1}^{C}\sum_{i=1}^{N}u_{ci}\ln u_{ci} \\ & +T_t\sum_{c=1}^{C}\sum_{i=1}^{N}t_{ci}\ln t_{ci}+T_v\sum_{c=1}^{C}\sum_{j=1}^{K}v_{cj}\ln v_{cj}+\sum_{i=1}^{N}\lambda_i(\sum_{c=1}^{C}u_{ci}-1) \\ & +\sum_{c=1}^{C}\beta_c(\sum_{i=1}^{N}t_{ci}-1)+\sum_{c=1}^{C}\gamma_c(\sum_{j=1}^{K}v_{cj}-1) \end{aligned} \tag{15}$$

where λ_i, γ_c and β_c are Lagrange multipliers for constraints in Eqs. (2), (3) and (8) respectively, the necessary conditions for the optimal solution of the Lagrange multiplier method. Taking the partial derivative of objective function in Eq. (15) with respect to u_{ci}, t_{ci} and v_{cj} and setting the gradient to zero, and then we have,

$$\frac{\partial J}{\partial u_{ci}} = \sum_{j=1}^{K} v_{cj} D_{cij} + T_u \left(\ln u_{ci} + 1 \right) + \lambda_i = 0 \tag{16}$$

$$\frac{\partial J}{\partial t_{ci}} = \sum_{j=1}^{K} v_{cj} D_{cij} + T_t \left(\ln t_{ci} + 1 \right) + \beta_c = 0 \tag{17}$$

$$\frac{\partial J}{\partial v_{cj}} = \sum_{i=1}^{N} \left(u_{ci} + t_{ci} \right) D_{cij} + T_v \left(\ln v_{cj} + 1 \right) + \gamma_c = 0 \tag{18}$$

Subjecting u_{ci} derived from Eq. (16) to the constraint in Eq. (2) the formula for computing u_{ci} reduces to,

$$u_{ci} = \frac{e^{(-\sum_{j=1}^{K} \frac{v_{cj} D_{cij}}{T_u})}}{\sum_{f=1}^{C} e^{(-\sum_{j=1}^{K} \frac{v_{fj} D_{fij}}{T_u})}} \tag{19}$$

Applying the constraint in Eq. (8) to t_{ci} derived from Eq. (17), we obtain the formula for t_{ci} as.

$$t_{ci} = \frac{e^{(-\sum_{j=1}^{K} \frac{v_{cj} D_{cij}}{T_u})}}{\sum_{f=1}^{N} e^{(-\sum_{j=1}^{K} \frac{v_{cj} D_{cfj}}{T_u})}} \tag{20}$$

In a similar manner, applying the constraint in Eq. (3) to v_{cj} derived from Eq. (18), we obtain the formula for v_{cj} as.

$$v_{cj} = \frac{e^{(-\sum_{i=1}^{N} \frac{(u_{ci} + t_{ci}) D_{cij}}{T_v})}}{\sum_{f=1}^{K} e^{(-\sum_{i=1}^{N} \frac{(u_{ci} + t_{ci}) D_{cif}}{T_v})}} \tag{21}$$

Note that when considering distance of information bottleneck as the distance function in Eq. (14), the cluster center is difficult to be expressed directly by the formula. In this paper, we present a new algorithm that directly calculates cluster centroids by employing a weighted averaging method based on u_{ci}, t_{ci} and x_{ij} , since p_{cj} means feature cluster centroid. The algorithm is given in Table 2.

The objective function in Eq. (14) can be minimized by alternatively updating the above membership equations until convergence is achieved. The ibPFCC can be written as Table 3.

Because each iteration needs to update all memberships, the time complexity of ibPFCC is O($CNK\tau$), which is equivalent to such fuzzy co-clustering algorithms as FCCM and FCCI, where τ denotes the number of iterations. We can prove that the ibPFCC algorithm could converge to a local minimum of the optimization, and the detailed proof procedure can be found in Appendix 1.

Table 2. Calculating cluster centers

```
Algorithm: Calculating the values of $p_{cj}$

Input: $C$, $N$, $K$
Output: the cluster center
Method:
    for $c$ = 1,2,……$C$ do
      for $j$ = 1,2……$K$ do
        fractions=0;
        numerator=0;
        for $i$=1,2……$N$ do
          fractions = fractions + $(u_{ci}+t_{ci})*x_{ij}$;
          numerator = numerator + $(u_{ci}+t_{ci})$;
        end
        $p_{cj}$ = fractions/numerator;
      end
    end
```

Table 3. Pseudo-code of ibPFCC algorithm

```
Algorithm: ibPFCC

Input: $C$, $N$, $K$, $\varepsilon$, $\tau_{max}$
Output: fuzzy object partitioning membership
Method:
    Set parameter $T_u$, $T_t$, $T_v$;
    Set iteration number $\tau$=0;
    Randomly initialize $u_{ci}$ and $t_{ci}$, such that $0 \le u_{ci} \le 1$, $0 \le t_{ci} \le 1$;
    REPEAT
      Calculate $p_{cj}$ using the algorithm as Table 2;
      Calculate $D_{cij}$ using Eq. (13);
      Calculate $v_{cj}$ using Eq. (21);
      Calculate $u_{ci}$ using Eq. (19);
      Calculate $t_{ci}$ using Eq. (20);
      $\tau=\tau+1$;
    UNTIL max$|u_{ci}(\tau)-u_{ci}(\tau-1)| \le \varepsilon$ or $\tau=\tau_{max}$
```

4. Experiments

In order to test the effectiveness of ibPFCC, we carried out a set of experiments on several document data sets. Experimental results are compared with four well received approaches in the literature, FCM, FCCI, RFCC and PFCC (Possibilistic fuzzy co-clustering of large document collections). We choose four standard clustering datasets to evaluate the performance of ibPFCC, 20NewsGroups, Ohsumed, UW-CAN, and Reuters-21578. The details of these datasets in our experiments are given as Table 4.

Table 4. List of datasets

Database	# docs	# words	Categories
NG (20NewsGroups)	500	500	4
OH (Ohsumed)	1,000	500	5
UC (UW-CAN)	314	2,000	10
RT (Reuters-21578)	2,315	2,000	10

4.1 Experimental Setup

The 20NewsGroups dataset is a collection of approximately 20,000 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups, and has become a popular data set for experiments in text applications of machine learning techniques, such as text classification and text clustering.

The Ohsumed dataset is a subset of the MEDLINE database, which includes medical abstracts from the MeSH categories of the year 1991. The specific task was to categorize the 23 diseases categories identified from C01 to C23. The 3 subsets come from the first 20,000 abstracts that are about cardiovascular diseases.

The UW-CAN dataset contains 314 web pages that have been taken from the University of Waterloo and various Canadian web sites. The pages are pre-classified into 10 different categories/classes. We use this existing classification as our baseline on how the dataset should be clustered.

The Reuters-21578 dataset is a collection of documents that appeared on Reuter newswire in 1987. The documents were assembled and indexed with categories.

In our experiments, the maximum number of iterations $\tau_{max} = 50$ and the maximum error limit $\varepsilon = 0.0001$. Each algorithm is implemented on each dataset for 10 times, and the average accuracy is recorded as the final experimental result. Because of the different data sets, the optimal parameter values of each algorithm are also different. In FCM, the parameter m equals 1.2, 1.5, 1.4 and 1.5 on the datasets, *NG*, *OH*, *UC* and *RT*, respectively. There are two parameters in FCCI, T_u and T_v. The T_u is 0.1, 5.0, 1.0E-6 and 1.0E-6, and the T_v is 0.1, 1.0E+7, 1.0E-2 and 1.0E-2, on the four datasets, respectively. The RFCC and PFCC both have three parameters. In RFCC, the first parameter is T_u, which equals 1.0E-4, 1.0, 0.1 and 0.1 respectively; the second parameter is T_v, which equals 1.0E+7, 1.0E+4, 1.0E+6 and 1.0E+6 respectively; and T_x is the third parameter, which equals 1.0E+3, 1.0E-3, 1.0E+4 and 1.0E-3 respectively. In PFCC, the three parameters are T_u, T_v and T_w. On the four datasets, the values of the T_u are 5.0E-8, 5.0E-8, 1.0E+5 and 1.0E+7, the values of the T_v are 1.0E+4, 1.0E-4, 1.0E-3 and 0.01, and the values of T_w are 1.0E+5, 1.0E-4, 1.0E+4 and 1.0E+4, respectively. Our algorithm, ibPFCC, has also three parameters, T_u, T_t and T_v. In our experiments, these three parameters remains constant on the four datasets, whose values are 1.0E-8, 1.0 and 1.0, respectively.

4.2 Evaluation Measures

Currently there are many clustering evaluation functions, including Entropy, F-Measure, purity, Similarity Overall and so on. In this paper, to evaluate the clustering quality of ibPFCC, we choose two evaluation criteria, F-Measure and Entropy, which are frequently used in clustering.

F-Measure is the harmonic average of precision and recall, which is always to evaluating cluster quality. The larger the value, the better the clustering performance, and vice versa. The F-Measure of a cluster *c* and a standard class *i* is given by,

$$F(c,i) = \frac{2P(c,i)R(c,i)}{P(c,i)+R(c,i)} \tag{22}$$

where *P*(*c*,*i*) and *R*(*c*,*i*) are the precision and recall between the cluster *c* and the standard class *i* respectively,

$$P(c,i) = \frac{N_{ci}}{N_c} \tag{23}$$

$$R(c,i) = \frac{N_{ci}}{N_i} \tag{24}$$

where N_{ci} is the number of members of the class *i* in the cluster *c*, N_c is the number of members of the cluster *c*, and N_i is the number of members of the class *i*.

The overall F-Measure for the clustering results is the weighted average of the F-Measure of each class *i*,

$$F = \frac{\sum_i N_i \times F(c,i)}{\sum_i N_i} \tag{25}$$

The Entropy is contrary to F-Measure, and the lower the value, the better the clustering quality. The formula of Entropy of the cluster *c* is calculated as,

$$e_c = -\sum_{i=1}^{C} P(c,i)\log P(c,i) \tag{26}$$

The total Entropy of the clustering result is calculated as the sum of Entropies of each cluster weighted by the size of that cluster,

$$e = \sum_{c}^{C} \frac{N_c}{N} e_c \tag{27}$$

where *N* denotes the total number of documents.

4.3 Experimental Results

The ibPFCC aims at minimizing the objective function in Eq. (14), and thus it is a prerequisite to converge to a local minimum. Although in Appendix 1 of this paper, rigorous theoretical proof of the convergence for ibPFCC is provided, we inspect the real value of objective function in our experiments before discussing the performance comparisons. Fig. 2 illustrates the objective function value changes of ibPFCC on the four datasets, *NG*, *OH*, *UC* and *RT*. It is evident that if we choose different data sets, the value will be in the different interval. However anyway, with the iteration times increasing, the value of objective function decreases gradually. When the iteration time exceeding 10 (even 5), the value has reached a plateau. It shows that ibPFCC has a rapid constringency speed of approaching local minimum, which can significantly improve the efficiency of clustering.

Clustering accuracy is one of the most important indicators to measure clustering algorithms. The results illustrated in Fig. 3(a), (b), (c) and (d) show comparisons on the clustering quality on the four datasets, *NG*, *OH*, *UC* and *RT*, respectively. In Fig. 3(a) on the first data set, the percentage of improvement ranges from 7.9 (against PFCC) to 44.7 (against FCM) percent increase in the F-measure quality, and 2.6 (against PFCC) to 20.8 (against RFCC) percent drop in Entropy (lower is better for Entropy). Fig. 3(b) shows the clustering results on the *OH* dataset. Our ibPFCC achieves the highest F-Measure, 0.35, and the lowest Entropy, 0.64. Although the improvement is slight, clustering accuracy is still the highest. Experimental results on the *UC* dataset are illustrated as Fig. 3(c). On this dataset, the ibPFCC and FCCI get much higher F-Measure and lower Entropy than the counterparts, which shows that clustering accuracies of these two algorithms are much higher on the dataset. Further, an improvement is achieved by our ibPFCC, reaching 2.5 percent in terms of F-Measure and 31 percent in terms of Entropy, over FCCI. For the last data set (Fig. 3(d)), the improvement is very significant. In terms of F-Measure, the improvement reaches 33.3%, 25.9%, 11.5% and 3.0% over FCM, FCCI, RFCC and PFCC, respectively. And the improvement in terms of Entropy also reaches 51.5%, 50.0%, 52.2% and 52.2% over FCM, FCCI, RFCC and PFCC, respectively.

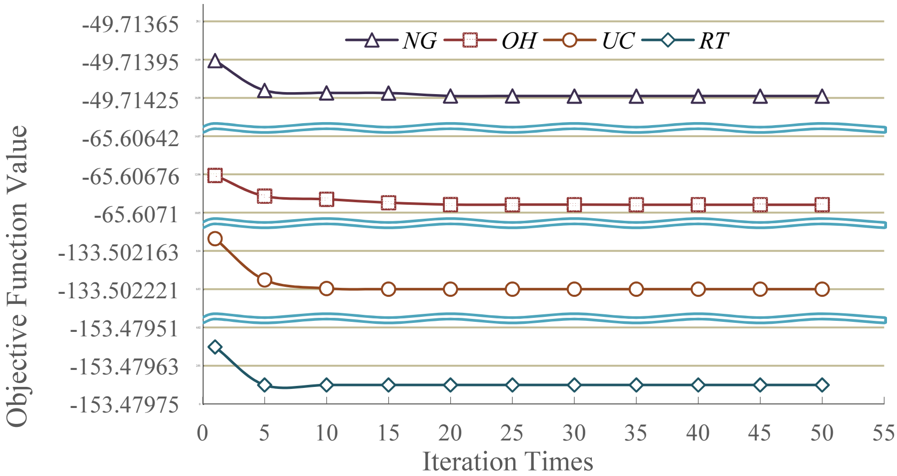
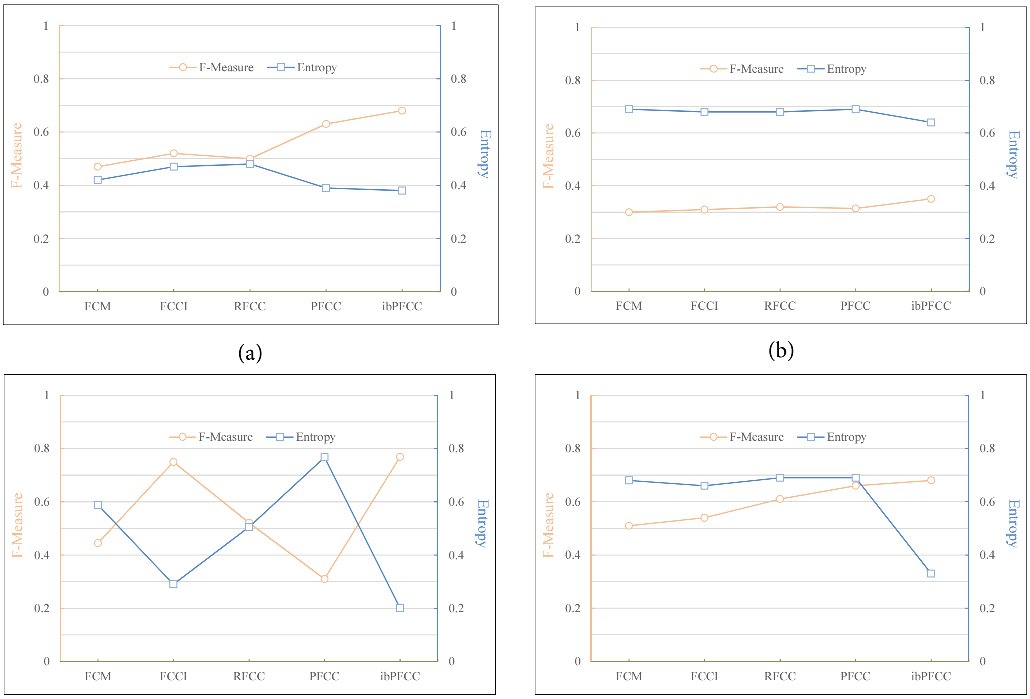


Fig. 2. Objective function values during optimization.



(a) (b)

(c) (d)

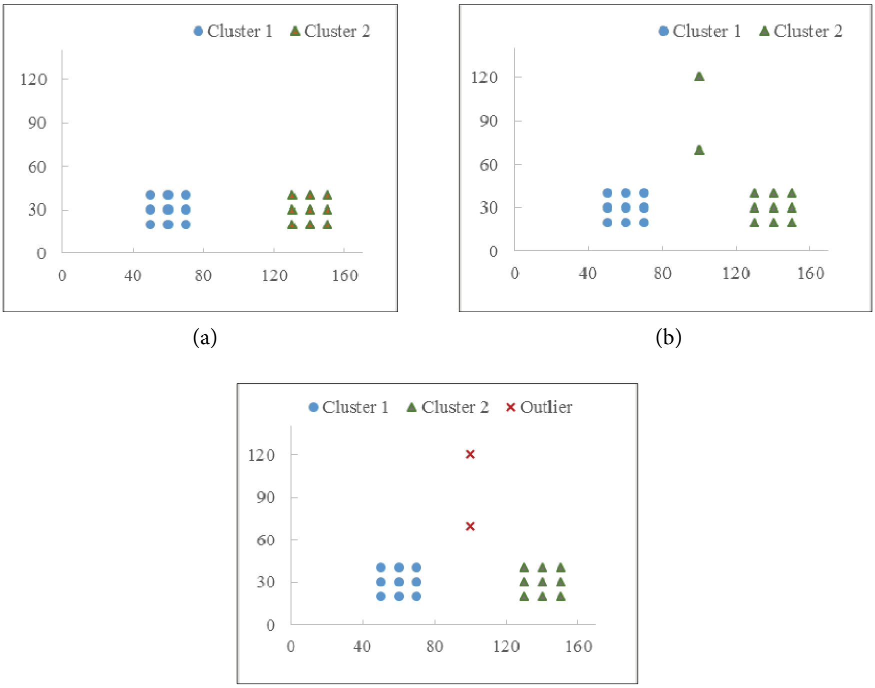
Fig. 3. Quality of clustering comparison in terms of F-Measure and Entropy. (a) NG, (b) OH, (c) UC, and (d) RT.

Fig. 3 shows the achieved improvement in comparison with the other approaches. This could be attributed to the combination of fuzzy clustering and possibilistic clustering, and the similarity measure based on information bottleneck.

Because ibPFCC absorbs advantages of possibilistic clustering, it should achieve strong capability of overcoming noise sensitivity defect. To confirm this and further enrich the discussions, we conduct more experiments where a new artificial dataset is constructed. The dataset involves two well-separated clusters of nine points each. As shown in Fig. 4(a), the three algorithms, FCM, FCCI and ibPFCC, generate the same final crisp partition, and essentially have the same cluster centers as in Table 5. Note that the RFCC and PFCC have no updating formulae of cluster centers, and thus does not participate in this group of experiments, because cluster center will be an important indicator to measure the robustness. Fig. 4(b) and (c) show the final crisp partitions obtained from the FCM and the FCCI and ibPFCC algorithms, respectively, after two noise objects are added into the dataset. And the membership values of these three algorithms are shown in Table 6. The FCM algorithm gives approximately equal membership of 0.5 in both clusters for the noise points (as the first two rows in Table 6). And the clustering results of FCM is illustrated as Fig. 4(b), where the two outliers are put into one cluster. Because two outliers infiltrate into clusters, the clustering quality will be significantly affected. The most remarkable effect is that the cluster centers move from (60.0, 30.0) and (140.0, 30.0) to (62.3, 33.7) and (137.7, 33.7) respectively, as can be seen in Table 5.

In FCCI, the membership case of the two outliers is similar to the FCM case, and the outliers are also both grouped into one cluster (as Fig. 4(b) and Table 6). Fig. 4(b) illustrates the clustering results of

FCCI. Also, because the two outliers are included in clustering results, the cluster centers move from (60.0, 30.0) and (140.0, 30.0) to (65.5, 36.5) and (134.5, 36.5) respectively (as Table 5), which will significantly lower the clustering accuracy.



(a)

(b)

(c)

Fig. 4. Results on a simple dataset: (a) the crisp partition without noise resulting from the FCM, FCCI and ibPFCC algorithms; (b) the crisp partition with noise resulting from the FCM and FCCI algorithms; (c) the crisp partition with noise resulting from the ibPFCC algorithm.

Table 5. Cluster centers resulting from the FCM, FCCI and ibPFCC algorithms before and after the noise is added

	FCM	FCCI	ibPFCC
No noise	(60.0, 30.0) (140.0, 30.0)	(60.0, 30.0) (140.0, 30.0)	(60.0, 30.0) (140.0, 30.0)
With noise	(62.3, 33.7) (137.7, 33.7)	(65.5, 36.5) (134.5, 36.5)	(75.2, 30.1) (124.9, 30.1)

In ibPFCC, the membership values are also approximately equal to 0.5 in both clusters for the noise points. However, different from FCM and FCCI, the ibPFCC does not classify objects based on only the membership values but also the typicality values. Note that the farther away the feature vector is to the typical member (i.e., the prototype), the smaller the typicality [5]. In Table 6, the membership values of one outlier are 0.505 and 0.495 (as the first row in Table 6), and thus this noise object should be put into the first cluster. However, the typicality values of this outlier are both 0.003, which shows that the aberrant outlier is clearly the least typical point in both clusters and far away from either cluster. The memberships of the other outlier are 0.488 and 0.512, and it should belong to the second cluster in pure fuzzy clustering. The typicality values of this noise object are both 0.041, which shows : (1) like the former one, the typicality of this outlier to both clusters is equal; (2) the typicality value 0.041 is greater than 0.003, the typicality value of the former outlier, which indicates that this outlier is much closer to the clusters; (3) nevertheless, the value 0.041 is much less than the typicality values of other objects in the second cluster, therefore, in contrast, this object is still aberrant. In ibPFCC, the original cluster centers are moved from (60.0, 30.0) and (140.0, 30.0) to (75.2, 30.1) and (124.9, 30.1) respectively (as Table 5), because we add the two noise objects into the final clusters. In fact, the ibPFCC could identify noise objects, and in that case, the cluster centers are virtually unchanged. Above discussion shows that the ibPFCC achieves stronger robust performance than such fuzzy (co-) clustering algorithms as FCM and FCCI.

Table 6. Memberships resulting from the FCM, FCCI and ibPFCC algorithms after the noise is added

	FCM		**FCCI**		**ibPFCC(u_{ci})**		**ibPFCC(t_{ci})**	
	Cluster1	**Cluster2**	**Cluster1**	**Cluster2**	**Cluster1**	**Cluster2**	**Cluster1**	**Cluster2**
1	0.500	0.500	0.500	0.500	0.505	0.495	0.003	0.003
2	0.500	0.500	0.500	0.500	0.488	0.512	0.041	0.041
3	0.994	0.006	0.959	0.041	0.866	0.134	0.055	0.014
4	0.982	0.018	0.970	0.030	0.814	0.186	0.067	0.023
5	0.953	0.047	0.951	0.049	0.747	0.253	0.074	0.033
6	0.994	0.006	0.979	0.021	0.866	0.134	0.068	0.018
7	0.982	0.018	0.997	0.003	0.813	0.187	0.083	0.029
8	0.953	0.047	0.984	0.016	0.746	0.254	0.092	0.042
9	0.994	0.006	0.976	0.024	0.865	0.135	0.070	0.018
10	0.982	0.018	0.993	0.007	0.812	0.188	0.085	0.029
11	0.953	0.047	0.979	0.021	0.745	0.255	0.094	0.043
12	0.047	0.953	0.049	0.951	0.244	0.756	0.034	0.072
13	0.018	0.982	0.030	0.970	0.185	0.815	0.025	0.068
14	0.006	0.994	0.041	0.959	0.139	0.861	0.018	0.062
15	0.047	0.953	0.016	0.984	0.243	0.757	0.042	0.089
16	0.018	0.982	0.003	0.997	0.184	0.816	0.031	0.084
17	0.006	0.994	0.021	0.979	0.138	0.862	0.022	0.077
18	0.047	0.953	0.021	0.979	0.242	0.758	0.043	0.091
19	0.018	0.982	0.007	0.993	0.184	0.816	0.031	0.086
20	0.006	0.994	0.024	0.976	0.137	0.863	0.022	0.078

5. Conclusion

Fuzzy co-clustering could simultaneously group objects and features, and therefore has many advantages such as dimensionality reduction and interpretable document cluster that are kept from co-clustering and fuzzy clustering. However, like FCM, fuzzy co-clustering also usually suffers from the inherent noise sensitivity defect, which lies in the membership constraint in FCM-type clustering. In this paper, we overcome this problem by combining possibilistic clustering, which relaxes the constraint, with fuzzy co-clustering, and propose a mixed clustering algorithm, named ibPFCC. In ibPFCC, we formulate an objective function including a distance function based on information bottleneck as the dissimilarity measure and entropy as the regularization term. To test the effectiveness of ibPFCC, we implemented experiments on four standard datasets, and the experimental results show that the proposed algorithm outperforms such fuzzy (co-)clustering algorithms as FCM, FCCI, RFCC and PFCC, in terms of accuracy and robustness.

Acknowledgement

The authors would like to thank the support of the Natural Science Foundation of China (Grant No. 61202286) and Foundation for University Key Teacher by Henan Province (Grant No. 2015GGJS-068). The authors would also like to thank members of the IR&DM Research Group from Henan Polytechnic University for their invaluable advice that makes this paper successfully completed.

Appendix 1

The proof of convergence of the ibPFCC algorithm is shown below:

Based on the bounded monotonic principle, we know that a monotone bounded function is convergent. Therefore, in order to prove the convergence of ibPFCC, we need to prove that the value of J_{ibPFCC} never increases when we update Eqs. (19-21) and J_{ibPFCC} is a bounded function.

Theorem 1. In every iteration, the updated value of u_{ci} given by Eq. (19) never increases the value of the objective function J_{ibPFCC} in Eq. (14).

Proof. We consider the objective function J_{ibPFCC} as a function of a single variable u_{ci}, denoted by $J(U)$:

$$J(U)=\sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K}\left(u_{ci}+t_{ci}\right)*v_{cj}D_{cij} + T_u\sum_{c=1}^{C}\sum_{i=1}^{N}u_{ci}\ln u_{ci}+constant \tag{28}$$

where $constant=T_t\sum_{c=1}^{C}\sum_{i=1}^{N}t_{ci}\ln t_{ci}+T_v\sum_{c=1}^{C}\sum_{j=1}^{K}v_{cj}\ln v_{cj}$.

Similarly, the variables v_{cj}, D_{cij} and t_{ci} may be considered as three constants. And then theorem 1 can be proven by showing that the u^* (i.e., the updated value of u_{ci} given by Eq. (19)) is the local minima of the objective function $J(U)$ by Lagrange multiplier method. For this we need to prove that the Hessian matrix $\triangle^2 J(u^*)$ is positive definite.

$$\Delta^2 J(u)=\begin{bmatrix}\frac{\partial^2 J(u)}{\partial u_{11}\partial u_{11}} & \cdots & \frac{\partial^2 J(u)}{\partial u_{11}\partial u_{CN}}\\ \vdots & \ddots & \vdots\\ \frac{\partial^2 J(u)}{\partial u_{CN}\partial u_{11}} & \cdots & \frac{\partial^2 J(u)}{\partial u_{CN}\partial u_{CN}}\end{bmatrix}=\begin{bmatrix}\frac{Tu}{u_{11}} & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & \frac{Tu}{u_{CN}}\end{bmatrix} \tag{29}$$

At u^*, $u_{ci}\geq 0$ and T_u is always assigned with a positive value. Therefore the Hessian matrix $\Delta^2 J(u^*)$ is positive definite. In summary, u^* is the objective function of stationary point $((\partial J\ (u_{ci})/\partial u_{ci})=0)$ and Hessian matrix $\Delta^2 J(u^*)$ is positive definite. By sufficient and necessary condition for the existence of extreme value of multivariate function knows that the updated u_{ci} is indeed a local minima of $J\ (U)$ and it never increases the objective function value.

Theorem 2. The updated values of t_{ci} given by Eq. (20) never increase the objective function J_{ibPFCC} in Eq. (14) in every iteration.

Proof. Theorem 2 can be proven in a similar fashion as Theorem 1.

Theorem 3. At every iteration, the updated values of v_{cj} given by Eq. (21) never increase the objective function J_{ibPFCC} in Eq. (14).

Proof. Theorem 3 can be proven in a similar fashion as Theorem 1.

Theorem 4. The objective function of J_{ibPFCC} in Eq. (14) is bounded. In other words, there is a constant M, which makes the J_{ibPFCC} more than M all the way (i.e., $J_{ibPFCC}\geq M$).

Proof. Since the minimum value of u_{ci}, t_{ci} and v_{cj} is 0, and $D_{cij}\geq 0$, we know that the first term of J_{ibPFCC} is greater than or equal to 0, that is,

$$\sum_{c=1}^{C}\sum_{i=1}^{N}\sum_{j=1}^{K}\left(u_{ci}+t_{ci}\right)*v_{cj}D_{cij}\geq 0 \tag{30}$$

The second, third and fourth terms of J_{ibPFCC} in Eq. (14) are all entropy regularization terms, and when $u_{ci}=1/C$, $t_{ci}=1/N$ and $v_{cj}=1/K$, the minimum value of the function will be achieved.

$$J_{ibPFCC}\geq T_u*N*\log\frac{1}{C}+T_t*C*\log\frac{1}{N}+T_v*C*\log\frac{1}{K} \tag{31}$$

Because T_u, N, C, T_t, T_v and K are all constants, we can get that $J_{ibPFCC}\geq M$, when $M=T_u*N*\log(1/C)+T_t*C*\log(1/N)+T_v*C*\log(1/K)$. In summary, the objective function J_{ibPFCC} is bounded.

Corollary 1. The ibPFCC algorithm converges to a local minimum of the optimization, with the update formulae given in Eqs. (19-21).

Proof. This corollary is a direct consequence of the above four theorems. Theorems 1-3 indicate that the procedure of membership updating never increases the value of the J_{ibPFCC}. Theorem 4 states that there is a limit to how much this objective function can be decreased. So eventually the procedure should stop somewhere before or when it reaches this limit.

References

[1] K. M. Hammouda and M. S. Kamel, “Efficient phrase-based document indexing for web document clustering,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 10, pp. 1279-1296, 2004.

[2] Y. Liu, T. Yang, and L. Fu, "A partitioning based algorithm to fuzzy tricluster," *Mathematical Problems in Engineering*, vol. 2015, article ID. 235790, 2015.

[3] N. R. Pal, K. Pal, and J. C. Bezdek, "A mixed c-means clustering model," in *Proceedings of 6th International Fuzzy Systems Conference*, Barcelona, Spain, 1997, pp. 11-21.

[4] T. C. Havens, R. Chitta, A. K. Jain, and R. Jin, "Speedup of fuzzy and possibilistic kernel c-means for large-scale clustering," in *Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, Taipei, Taiwan, 2011, pp. 463-470.

[5] W. C. Tjhi and L. Chen, "Possibilistic fuzzy co-clustering of large document collections," *Pattern Recognition*, vol. 40, no. 12, pp. 3452-3466, 2007.

[6] J. P. Mei, Y. Wang, L. Chen, and C. Miao, "Incremental fuzzy clustering for document categorization," in *Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Beijing, China, 2014, pp. 1518-1525.

[7] J. Liu, X. Wu, and X. Luo, "Fuzzy clustering research based on intelligent computing," in *Proceedings of International Conference on Intelligent Transportation, Big Data and Smart City*, Halong Bay, Vietnam, 2015, pp. 429-432.

[8] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York, NY: Plenum Press, 1981, pp. 203-239.

[9] M. Hanmandlu, O. P. Verma, S. Susan, and V. K. Madasu. "Color segmentation by fuzzy co-clustering of chrominance color features," *Neurocomputing*, vol. 120, pp. 235-249, 2013.

[10] C. H. Oh, K. Honda, and H. Ichihashi, "Fuzzy clustering for categorical multivariate data," in *Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference (Cat. No. 01TH8569)*, Vancouver, Canada, 2001, pp. 2154-2159.

[11] W. C. Tjhi and L. Chen, "Robust fuzzy co-clustering algorithm," in *Proceedings of 6th International Conference on Information, Communications & Signal Processing*, Singapore, Singapore, 2007, pp. 1-5.

[12] J. Leski, "Robust possibilistic clustering," *Archives of Control Sciences*, vol. 10, no. 3/4, pp. 141-155, 2000.

[13] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 98-110, 1993.

[14] M. Barni, V. Cappellini, and A. Mecocci, "Comments on 'A possibilistic approach to clustering'," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 3, pp. 393-396, 1996.

[15] X. Wan, "A novel document similarity measure based on earth mover's distance," *Information Sciences*, vol. 177, no. 18, pp. 3718-3730, 2007.

[16] H. Izakian, W. Pedrycz, and I. Jamal, "Fuzzy clustering of time series data using dynamic time warping distance," *Engineering Applications of Artificial Intelligence*, vol. 39, pp. 235-244, 2015.

[17] N. Slonim and N. Tishby, "Document clustering using word clusters via the information bottleneck method," in *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Athens, Greece, 2000, pp. 208-215.

[18] N. Slonim, N. Friedman, and N. Tishby. "Unsupervised document classification using sequential information maximization," in *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, 2002, pp. 129-136.

[19] B. Abidi and S. B. Yahia, "Multi-PFKCN: a fuzzy possibilistic clustering algorithm based on neural network," in *Proceedings of 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Hyderabad, India, 2013, pp. 1-8.

[20] K. Duraisamy and K. Haridass. "Modified fuzzy possibilistic C-means," *Fuzzy Systems*, vol. 6, no. 3, pp. 78-83, 2014.

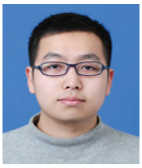
[21] J. Goldberger, H. Greenspan, and S. Gordon, "Unsupervised image clustering using the information bottleneck method," in *Proceedings of 24th DAGM Symposium*, Zurich, Switzerland, 2002, pp. 158-165.

[22] Y. Liu, Y. Ouyang, and Z. Xiong. "Incremental clustering using information bottleneck theory," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 5, pp. 695-712, 2011.

[23] W. C. Tjhi and L. Chen, "A partitioning based algorithm to fuzzy co-cluster documents and words," *Pattern Recognition Letters*, vol. 27, no. 3, pp. 151-159, 2006.

Yongli Liu https://orcid.org/0000-0002-0540-865X

He received his Ph.D. degree in computer science and engineering from Beihang University in 2010. He is currently an associate professor in Henan Polytechnic University. His current research interests include data mining and information retrieval.

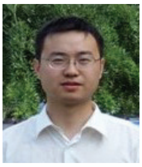


Tianyi Duan

He is currently a master student in Henan Polytechnic University. His current research interests include data mining and information retrieval.

Xing Wan

He is currently a master student in Henan Polytechnic University. His current research interests include data mining and information retrieval.



Hao Chao

He received his Ph.D. degree in pattern recognition and intelligent system from Institute of Automation, Chinese Academy of Sciences in 2012. He is currently a lecturer in Henan Polytechnic University. His current research interests include data mining and speech recognition.