Int. J. Mach. Learn. & Cyber. (2018) 9:1085–1100
DOI 10.1007/s13042-016-0628-6

ORIGINAL ARTICLE

Combined constraint-based with metric-based in semi-supervised clustering ensemble

Siting Wei[1,2] · Zhixin Li[1,2,3] · Canlong Zhang[1,2,3]

Received: 30 July 2015 / Accepted: 27 December 2016 / Published online: 17 February 2017

Abstract Recently, both semi-supervised clustering and cluster ensemble have received tremendous attention due to their accurate and reliable performance. There are mainly two kinds of existing semi-supervised clustering algorithms called constraint-based and metric-based. In this paper, we present a semi-supervised clustering ensemble approach which takes both pairwise constraints and metric measure into account. Firstly, under the assistance of supervised information included pairwise constraints and labeled data, the approach generates different base clustering partitions respectively using constraint-based semi-supervised clustering and metric-based semi-supervised clustering, in which the latter develops a new metric function. Given the spatial particularity of image pixels, the metric considers spatial distribution of surrounding pixels besides inherent features of pixels in the process of image feature extraction. And then the target clustering is obtained by integrating those base clustering partitions into an ensemble function. Finally, we conduct experimental verification on general data sets and image data sets, and compare clustering performance of our approach with those of other approaches. Both theoretical analysis and experimental results demonstrate that the proposed method produces considerable improvement in clustering accuracy and yields superior clustering results over a number of representative clustering methods.

Keywords Emi-supervised clustering · Consensus function · Pairwise constraints · Metric measure · Image data clustering

1 Introduction

Clustering is acknowledged as one of the most important unsupervised learning methods in machine learning field. As a pre-processing technique, clustering has been frequently researched and widely applied to all kinds of practical scenes. It aims to categorize unlabeled samples into multiple classes based on the similarity between samples, and these multiple classes are often called clusters. In contrast to unsupervised learning, supervised learning needs a mass of labeled samples to assist cluster process. Nevertheless, labeled samples are relatively less in the real world because they are fairly difficult to get, so that it leads to undesirable clustering without the aid of labeled samples. Hence, the semi-supervised clustering [1] method emerges at the proper time. With a small amount of prior knowledge provided in advance, it divides plentiful unlabeled examples into several groups with higher clustering performance, and has become a hot spot in current research.

Despite a variety of clustering methods have been proposed, none is applicable to all data sets and achieves satisfactory clustering goals for all data sets. That is to say each clustering algorithm has own merits and demerits. Due to different data sets and different algorithms, as well as different parameters in the same algorithm, all these will lead to different results. By this token, how to select the proper algorithm is a significant and hard task for the users. The

✉ Zhixin Li
lizx@gxnu.edu.cn

[1] Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin 541004, China

[2] College of Computer Science and Information Technology, Guangxi Normal University, Guilin 541004, China

[3] Guangxi Experiment Center of Information Science, Guilin 541004, China

emerging cluster ensemble approach presented by Strehl et al. [2] can just ameliorate this puzzle. Du et al. [3] propose a novel self-supervised learning framework for clustering ensemble. Hao et al. [4] propose an improved clustering ensemble method based link analysis. In general, the basic idea of cluster ensemble is to integrate a variety of clustering partitions by using a specific consensus function to generate the final decision that outperforms individual clustering.

Semi-supervised clustering ensemble applies these two strategies simultaneously, namely semi-supervised clustering and cluster ensemble. Similarly, it combines different clustering results of various semi-supervised clustering algorithms by using ensemble function to create a single target clustering with more optimal performance than those of individual semi-supervised clustering. What's more, it has strengths of low sensitivity to noise, outliers and variables. Yu et al. [5] propose a feature selection method based semi-supervised cluster ensemble framework for tumor clustering from bio-molecular data. Yu et al. [6] propose an incremental semi-supervised clustering ensemble framework for high dimensional data clustering.

At present, two typical approaches of semi-supervised clustering called constraint-based and metric-based are researched a lot. The former revises objective function of algorithm to guide the process of clustering by using supervised information provided in advance. Xiong et al. [7] study the active learning problem of selecting pairwise constraints for semi-supervised clustering. Wang et al. [8] propose a semi-supervised nonnegative matrix factorization method with pairwise constraints. The latter exploits a specific distance/similarity metric for clustering to satisfy the given pairwise constraints. Yan et al. [9] propose a semi-supervised clustering method with multi-viewpoint based similarity measure. Yin et al. [10] develop a semi-supervised fuzzy clustering algorithm with metric learning and entropy regularization simultaneously (SMUC). Although the two kinds of methods have their own singular focus respectively, they aren't not only separated completely, but also exists symbiotic relationship between them. Inspired by the work of Bilenko et al. [11], many scholars have begun to turn their attention to the field of exploitation of hybrid approaches, which aims to combine the advantages of constraint-based with that of metric-based. In order to sufficiently solve the violation problem of pairwise constraints and to mitigate the problem of manually tuning the kernel parameters owing to the fact that no sufficient supervision, an adaptive semi-supervised clustering kernel method based on metric learning (SCKMM) is proposed by Yin et al. [12]. Arzeno and Vikalo [15] present an extension of soft-constraint semi-supervised affinity propagation (E-SCSSAP) which incorporates metric learning in the optimization objective and acquires desirable clusters. Reviewing previous related literature, we found that most researchers just take single objective function of the two factors into account, but relatively few researchers synthesize the two different kinds of algorithms adopted the mechanism of ensemble. Different from these conventional methods, this paper presents a semi-supervised clustering ensemble approach in conjunction with the both.

Additionally, most of image data metric measures used in previous literature are merely based on intrinsic properties of pixels. As we all known, one pixel and its surrounding pixels are tightly linked, so that it is necessary and reasonable to incorporate spatial characteristic in objective function. However, the most common way used in existing methods is selecting various means or statistical operators in a designated area around pixel as its spatial information, whose results still exist more or less deviation with the actual features. To mitigate the deviation, the paper concerns the intrinsic feature of pixel as well as spatial characteristic of its surroundings in metric measure simultaneously. Comparison experiments and analysis are made to validate the superiority of our method. The main contributions of this paper can be summarized as three aspects.

Firstly, this paper improves the performance of semi-supervised clustering by introducing ensemble mechanism, which unites different results respectively produced from constraint-based algorithm and metric-based algorithm. They have certain preoccupations in their fields that one concentrates on the adjustment of objective function according to pairwise constraints while the other is concerned with the introduction of metric function to measure the distance/similarity between samples more precisely. The combination method obtains benefits beyond what a single algorithm achieves.

Secondly, this paper proposes a metric-based semi-supervised clustering algorithm, in which the metric measure is formulated by two styles. One is used for measuring the distance between general data samples, and the other is for image pixel samples. Out of consideration for pixels' spatial properties, we conclude the point that metric between pixels should be collectively based on the inherent feature of pixels and its neighboring spatial information. The metric evaluation function is expressed as the ratio of the feature similarity based on spatial information to the distance based on inherent feature. This new perspective breaks through traditional single idea for pixels metric, and significantly improve the accuracy.

Thirdly, we conduct two group experiments respectively on general data sets and image data sets for performance evaluation of clustering. On the basis of the proposed approach, the former simply employs ensemble mechanism integrated a constraint-based method and a metric-based method with a general metric measure. Besides the stages of the former, the latter adds a space-based pixel similarity

to general metric measure that measures the similarity based on the inherent feature of pixels. It follows that our method not only applies to clustering of general data sets, but also to clustering of image data sets. Both of the two experiments complete targeted clustering on different data sets with high precision.

We organize the remainder of this paper into four sections. Section 2 surveys related works about what we have done. In Sect. 3, the new semi-supervised clustering ensemble approach is illustrated and the process of clustering with our proposed approach is described in detail. Section 4 provides experiment results and analysis. Finally we come to conclusions and look forward to several issues for future works in Sect. 5.

2 Related works

Semi-supervised clustering and clustering ensemble have been studied extensively, and the research topics in the fields of related are mainly distributed into three parts.

The first one is semi-supervised clustering learning approach. In real application, various useful prior knowledge, such as class labels or pairwise constraints, are taken into account in clustering process. This strategy of technique is called semi-supervised clustering [16]. The popular semi-supervised clustering approach is mainly composed of two categories called constraint-based method [16–20] and metric-based method [10, 21–23]. For the former, the linear-time constrained vector quantization error algorithm (LCVQE) [17] concerns on the problem of clustering with soft instance level constraints, in which a more intuitive objective function is introduced with lower computational complexity. Pairwise constrained maximum margin clustering approach (PMMC) [19] develops a set of effective loss functions for discouraging the violation of given pairwise constraints. For the later, the information theoretic metric method (ITML) [21] learns a Mahalanobis distance function by minimizing the differential relative entropy between two multivariate Gaussian under constraints on the distance function. Large margin nearest neighbor classification (LMNN) [22] focuses on how to learn a Mahalanobis distance metric for large margin nearest neighbor from labeled example so that the k-nearest neighbors always belong to the same class while examples from different classes are separated by a large margin. But beyond that, several hybrid approaches have been proposed gradually. For instance, Bilenko et al. [11] integrate the constraint-based method and distance-function learning method to form a hybrid method, which is a metric pairwise constrained k-means algorithm incorporated both metric learning and the use of pairwise constraints in a uniform and principled manner (MPCK). Yin et al. [12] propose an adaptive semi-supervised clustering kernel method based on metric learning (SCKMM), where the parameter of Gaussian kernel can be estimated through the objective function from pairwise constraints, and the pairwise constraint-based K-means is used to solve the violation of constraints. In addition, Lin et al. [13] proposes a semi-supervised grid clustering algorithm based on rough reduction (RSGrid). Zhang and Lu [14] provide a semi-supervised clustering approach using the kernel-based method based on KFCM (SSKFCM).

The second one is cluster ensemble learning approach. From the generation perspective, a set of diverse ensemble members is generated and known as base clustering in various forms. Base clustering can result from different views, different initialization parameter, different methods, and so on. But then it is difficult to learn a suitable consensus function to summarize the base clusterings and search for an optimal unified clustering decision. In light of that theoretical analysis, a great amount of well-known ensemble methods emerge. Three graph-based ensemble methods are introduced in study [2], all of which partition clusters based on a constructed similarity graph. The cluster-based similarity partition algorithm (CSPA) uses METIS to partition the induced similarity graph. The hyper-graph partition algorithm (HGPA) uses HMETIS to partition the hyper-graph. The meta-clustering algorithm (MCLA) collapses related hyper-edges and assigns each object to the collapsed hyper-edge in which it participates most strongly. An iterative voting consensus (IVC) [24] is a feature-based approach, in which each base clustering provides a cluster label as a new feature describing each data point that is utilized to formulate the final solution. By exploiting the significance of attribute defined in rough set theory, Wang et al. [25] apply the proposed two feature selection algorithms to a cluster ensemble selection problem.

The third one is semi-supervised clustering ensemble learning approach. Due to the lack of prior knowledge about cluster labels, cluster ensemble is still a challenging problem. By leveraging limited supervision information in cluster ensemble, semi-supervised clustering ensemble offers an effective solution to overcome this limitation and obtains accurate, robust and stable results. Wang et al. [26] construct semi-supervised cluster ensemble based on binary similarity matrix (BSMSCE), which takes the strengths of known information to improve the quality of clustering. Chen et al. [27] analyze convergence of semi-supervised clustering ensemble and proposed a new relabeling approach for semi-supervised clustering ensemble by majority voting (we called it MVSCE for short). Yu et al. [5] view the expert's knowledge as constraints in the process of clustering and propose a framework called FS-SSCE, which not only applies the feature selection technique to perform gene selection on the gene dimension, but

also selects an optimal subset of representative clustering solutions in the ensemble and improve the performance of tumor clustering using the normalized cut algorithm.

3 The proposed approach

3.1 An overview of the proposed approach

With the high stability and robustness, it is verified that cluster ensemble is an ideal alternative for a single clustering algorithm. For the purpose of improving performance of individual semi-supervised clustering algorithm, the paper makes great efforts to study about semi-supervised clustering ensemble, and introduces ensemble technique combining constraint-based semi-supervised clustering method and metric-based semi-supervised clustering method. We refer to the proposed hybrid semi-supervised clustering ensemble algorithm as HSCE for short. The procedure of our proposed approach is showed in Fig. 1.

Figure 1 shows the clustering process based on the proposed semi-supervised clustering ensemble approach. Based on the semi-supervised clustering ensemble model, the clustering process of HSCE can be described concisely as follows. (a) Supervised information (also called prior knowledge, such as class labels, pairwise constraints and the number of clusters) are specified in advance for further clustering. (b) It is the second step—clustering and partition generation—that seems to be the most essential step. From the generation perspective, with assistance of supervised information, data points are divided into different clustering groups by using constraint-based semi-supervised clustering approach and metric-based semi-supervised clustering approach respectively. Thus, several clustering partitions are generated and known as base clustering decisions. (c) Decisions integration. The consensus function employs a specific form of meta-level information matrix that stacks up all the previous base partition results, and it is available for deriving the final results with appealing properties superior to any individual one.

3.2 The constraint-based semi-supervised clustering Approach

3.2.1 Pairwise constraints

Generally, we hope to get more constraint information to aid clustering, because it is more easily obtained than class label information in practice. The pairwise constraints reflect prior knowledge about whether a pair of samples should be grouped together or not. It contains two types, namely must-link and cannot-link. Must-link indicates the two data points must be grouped in the same cluster marked as $M = \{(x_i, x_j)\}$ while cannot-link indicates the two data points must be grouped in different clusters marked as $C = \{(x_i, x_j)\}$. Pairwise constraints have transitivity and symmetry properties. Assume $x_i, x_j, x_k \in X$, the properties are showed as follows.

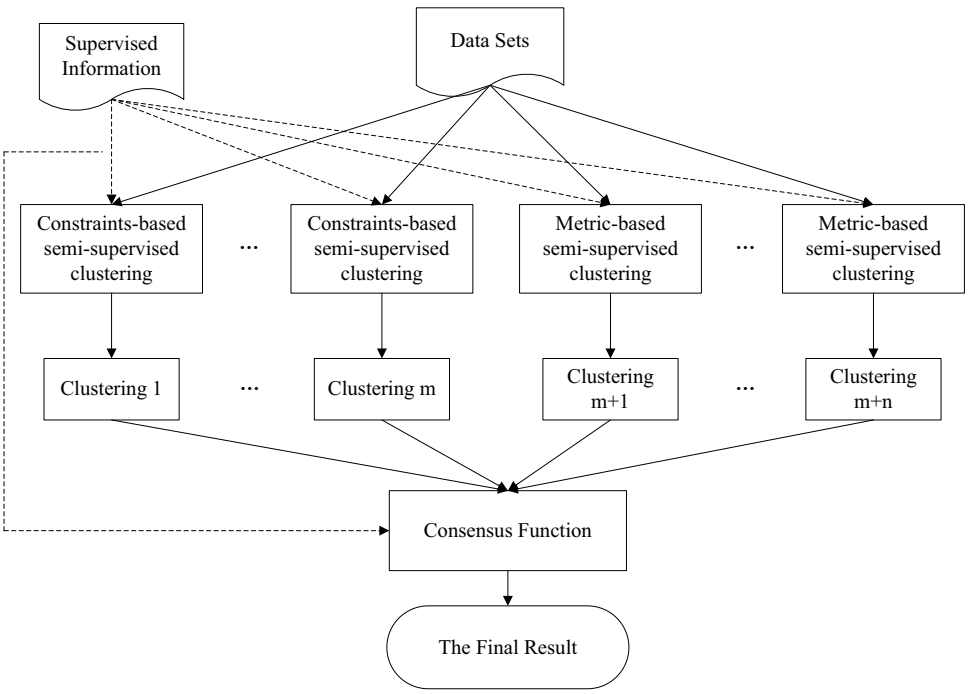

Fig. 1 An overview of the proposed approach

(1) Transitivity: $(x_i, x_k) \in M \ \varsigma \ (x_k, x_j) \in M \Rightarrow (x_i, x_j) \in M$,

$(x_i, x_k) \in M \ \varsigma \ (x_k, x_j) \in C \Rightarrow (x_i, x_j) \in C$

(2) Symmetry: $(x_i, x_j) \in M \Rightarrow (x_j, x_i) \in M$, $(x_i, x_j) \in C \Rightarrow (x_j, x_i) \in C$.

3.2.2 *The semi-supervised spectral clustering algorithm based on pairwise constraints*

In the section, the semi-supervised spectral clustering algorithm based on pairwise constraints (SSCA) [20] is used as the constraint-based semi-supervised clustering approach in our approach. The process of SSCA is briefly described as below. Firstly, it revises distance matrix of samples according to pairwise constraints information. If $(x_i, x_j) \in M$ then $D_{ij} = 0$; if $(x_i, x_j) \in C$, then $D_{ij} = \infty$. Secondly, it constructs the similarity matrix of samples according to this equation: $\overline{w}_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma_i \sigma_j}\right)$, where $\sigma_i = \frac{1}{k}\sum_{n=1}^{k} \|x_i - x_n\|$ is a corresponding parameter for each data point. And then spectral clustering algorithm is used to solve the eigenvalues and eigenvectors of Laplacian matrix, to which is transformed by the similarity matrix. Finally, it divides the sample set X into k clusters combining with kernel fuzzy c-means (KFCM) clustering [28].

3.3 The metric-based semi-supervised clustering approach

3.3.1 *The large margin nearest cluster distance metric (LMNC)*

From the study provided by Huang et al. [23], we learn that LMNC metric is a generalized inspired by Mahalanobis metric. One common goal for metric learning is min–max principle: minimize the distances between same-cluster samples meanwhile maximize the distances between different-cluster samples. A set of n labeled data is denoted as $\{(x_i, y_i)\}_{i=1}^{n}$, where $x_i \in R^d$, $y_i \in \{1, 2, ..., K\}$ represents discrete class labels. Let $\mathbf{M}$ be the symmetric matrix of size $d \times d$, for any two given data points x_i and x_j on a vector space R^d, the squared distance measure between them can be expressed by Eq. (1).

$$D(x_i, x_j) = (x_i - x_j)^T M (x_i - x_j) \tag{1}$$

In general, $\mathbf{M}$ is a positive semi-definite matrix, i.e. $\mathbf{M} \geq 0$. To learn matrix $\mathbf{M} \in R^{d \times d}$, a cost function over this distance metric can be constructed by Eq. (2).

$$\varepsilon(L) = \sum_{ij} a_{ij}(x_i - z_j)^T M (x_i - z_j) + c \sum_{ijl} a_{ij}(1 - a_{ij}) \left[1 + (x_i - z_j)^T M (x_i - z_j) - (x_i - z_l)^T M (x_i - z_l)\right]_+ \tag{2}$$

where the weight matrix $a_{ij} \in \{0, 1\}$ indicates that $a_{ij} = 1$ if the class labels y_i and y_j match, otherwise $a_{ij} = 0$. z_j represents the center of the j^{th} cluster, $c > 0$ is a positive constant, $[f]_+ = \max(f, 0)$ denotes the loss function. To reach min–max goal, we transform the loss metric problem into an optimization problem (3).

$$\text{Min} \sum_{ij} a_{ij}(x_i - z_j)^T M (x_i - z_j) + c \sum_{ijl} a_{ij}(1 - a_{ij}) \xi_{ijl} \tag{3}$$

$$\text{s.t.} \quad (1) \quad \xi_{ijl} \geq 0, \quad (2) \quad M \geq 0, \quad (3) \quad (x_i - z_l)^T M (x_i - z_l) - (x_i - z_j)^T M (x_i - z_j) \geq 1 - \xi_{ijl}.$$

This expression induces a slack variables ξ_{ijl} to represent the loss function. This optimization problem can be solved by means of the gradient projection algorithm [29].

3.3.2 *The similarity metric of image pixels based on spatial information (SMIP)*

In contrast to ordinary approaches, the innovation of the pixel affinity presented by Na and Yu [30], is the affinity based on patch [31] and the edge information on the lines of two pixels.

The first step is to solve patch-based similarity. For an image *I(x, y)*, the similarity of arbitrary two pixels i and j is denoted as $S_{ij} \in [0, 1]$. Normally, the more i similar to j is, the higher the value of $S_{ij}$is. Pixel's neighbor information should be considered to obtain stable feature due to the greater difference of the visual feature of single pixel. So the affinity based on patch can reflect the similarity between pixels veritably and satisfactorily. Using the average L*a*b* color space, the weighted average features of pixels in certain patch can be calculated by $\hat{I}(x, y) = I(x, y) * G_\sigma(x, y) = I(x, y) * G_\sigma(x) * G_\sigma(y)$, where * represents convolution operation, *(x, y)* describes one pixel's coordinate, $G_\sigma(x, y)$denotes a two-dimensional Gaussian kernel function used σ^2 as a variance, σ is designed for controlling the size. σ_I is the scale parameter for color feature. The patch-based similarity can be described as Eq. (4).

$$S_{ij}^{(1)} = \exp\left(\frac{-\left\|\hat{I}(i) - \hat{I}(j)\right\|_2^2}{\sigma_I^2}\right) \tag{4}$$

The second step is to solve edge-based similarity. In the light of the model conveyed by Cour et al. [32], we assume $E(x, y)$ is edge information of image $I(x, y)$, then the edge-based similarity between two pixels i and j is formulated by Eq. (5).

$$S_{ij}^{(2)} = \exp\left(\frac{-\max_{p\in line(i,j)}\left\|E\left(p_x, p_y\right)\right\|^2}{\sigma_E^2}\right) \tag{5}$$

where P indicates one of the pixels in the line from pixel i to j, $\left(p_x, p_y\right)$means its corresponding coordinate, σ_E^2 represents the scale parameter of edge-based similarity.

According to the study of Martin et al. [33], the patch that centers on one pixel p can be cut into two parts by diameter from one angle. The feature difference between two parts can be denoted as $E(x, y, \theta, r) = \left|E_+(x, y) - E_-(x, y)\right|$, where θ is a cutting angle, E_+ and E_- respectively represent the features of two sides, r means radius. By adjusting cutting angle, the direction of cutting may be as near to the actual one as possible until the biggest difference generates. At this point, the edge information can be denoted as $E(x, y) = \max_{\theta\in[0,180)} E(x, y, \theta, r)$. Applying the convolution theorem, the edge information at specific angle θ can be calculated by equation $E_\theta(x, y) = \left|I(x, y) * \nabla_\theta G(x, y)\right|$. For higher calculating speed, the edge feature is approximately represented by that at four angles 0°, 45°, 90°, 135°, transformed to $E(x, y) \approx E_0(x, y) + E_{45}(x, y) + E_{90}(x, y) + E_{135}(x, y)$.

On account of convolution property: $G_\theta(x, y) = G(x, y) * \nabla_\theta$, further transformation, $E(x, y) \approx \left|I_2(x, y) * \nabla_0\right| + \left|I_2(x, y) * \nabla_{45}\right| + \left|I_2(x, y) * \nabla_{90}\right| + \left|I_2(x, y) * \nabla_{135}\right|$, where $\nabla_0, \nabla_{45}, \nabla_{90}, \nabla_{135}$ denote corresponding partial derivatives filters ($v = 0.5\sigma\sqrt{2\pi}$, called normalizing factor).

$$\nabla_{\mathbf{0}} = \begin{bmatrix} -\mathbf{v} & \mathbf{0} & \mathbf{v} \end{bmatrix}, \nabla_{\mathbf{45}} = \begin{bmatrix} -\mathbf{v} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{v} \end{bmatrix}, \nabla_{\mathbf{90}} = \begin{bmatrix} -\mathbf{v} \\ \mathbf{0} \\ \mathbf{v} \end{bmatrix}, \nabla_{\mathbf{135}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{v} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{v} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

The third step is to solve final similarity. Integrating with the aforementioned approaches, we convey the final similarity as Eq. (6).

$$S_{ij} = \sqrt{S_{ij}^{(1)} \times S_{ij}^{(2)}} \tag{6}$$

3.3.3 *The proposed metric-based semi-supervised clustering approach*

In this part, we combine the two methods aforementioned, namely LMNC and SMIP, to form a new metric function firstly. And then the generated combination function is applied to a semi-supervised clustering algorithm [34] by substituting its objective function to form a new metric-based semi-supervised clustering algorithm. This algorithm proposed is called LSSC for short, which is used as the metric-based semi-supervised clustering approach in our proposed approach. Specifically, we formulate the similarity between x_i and x_j by using a new metric function. The metric function is denoted as Eq. (7).

$$\hat{D}_{ij} = \frac{S_{ij}}{D\left(x_i, x_j\right)} \tag{7}$$

It is applied to cases where samples are pixels of one image, $\hat{D}_{ij} = \frac{S_{ij}}{D\left(x_i, x_j\right)}$ is the similarity for image pixel samples, where S_{ij} is the space-based similarity calculated by Eq. (6). Otherwise $S_{ij} = 1$, in the case of general data similarity, and then $\hat{D}_{ij} = \frac{1}{D\left(x_i, x_j\right)}$ is the similarity for general data samples, where $D\left(x_i, x_j\right)$ is the LMNC distance metric. The larger $\hat{D}_{ij}$ is, the more similar the pair samples are. The LSSC algorithm is described in Algorithm 1.

Algorithm 1. The LSSC Algorithm

Input data set $X=\{x_1,x_2,...,x_n\}$; labeled data set $\hat{X}=\left\{\hat{x}_1,\hat{x}_2,...,\hat{x}_m\right\}$, $m\le n$; the number of cluster k; the max-number of iterations t

Output cluster partition $C=(c_1,c_2,...,c_k)$

Process

1. Determine cluster centers.

a. Extend $\hat{X}$ to adjacent data, which is assumed to belong to the same clustering, and generate labeled matrix $\tilde{X}$. Partition $\tilde{X}$ into s different labeled subsets $\left\{\overset{*}{X}_i\right\}$

b. If $s=k$, k cluster centers $v_i=\frac{1}{n_i}\sum_{x_l\in\overset{*}{X}_i}x_l$, $i\in\{1,2,...,k\}$, where n_i denotes the data total of the i-th cluster;

c. If $s<k$, s cluster centers $v_i=\frac{1}{n_i}\sum_{x_l\in\overset{*}{X}_i}x_l$, $i\in\{1,2,...,s\}$, and the rest k-s cluster centers need further to be determined, k cluster centers are obtained by means of k-means, then remove s cluster centers nearest to v_i and obtain the rest k-s ones;

d. If $s>k$, prompt error and reset labeled data.

2. Partition clustering

a. Divide X into different clusters c_j, $j\in\{1,2,...,k\}$. Every data point is categorized into the nearest cluster center by calculating Eq. (7).

3. Rebuild cluster centers

a. Recalculate cluster centers of c_j: $v_j^{\,b}=\frac{1}{\sum_{x_l\in c_j}\eta_{x_l}}\sum_{x_l\in c_j}\eta_{x_l}x_l$, where b is the number of iteration, η_{x_l} denotes the weight of x_l in the collection data points of c_j.

4. Until $b\ge t$ the algorithm stops; otherwise turn back to step 2.

3.4 Consensus function

The CSPA algorithm [2] aims to obtain the final clustering π^* from the set of M base clustering results marked as $\prod=\{\pi_1,\pi_2,...,\pi_M\}$. In details, this algorithm constructs a $N\times N$ similarity matrix for each base clustering π_m, and it is denoted as S_m, $m\in\{1,...,M\}$. For two data samples x_i and x_j in π_m, if x_i and x_j have the same class labels $C(x_i)=C(x_j)$, $S_m(x_i,x_j)=1$; if not $S_m(x_i,x_j)=0$. Then the M similarity matrix are merged to form a co-association (**CO**) matrix, which represented as $CO(x_i,x_j)=\frac{1}{M}\sum_{m=1}^{M}S_m(x_i,x_j)$. This algorithm creates a similarity graph, where vertexes represent data points and edges' weight represent similarity scores obtained from the **CO** matrix. Following that, the graph partitioning algorithm METIS is used to partition the similarity graph into k clusters and yield the final partition decision.

3.5 Complexity analysis

We also analyze the computational complexity of the HSCE algorithm. We refer to the corresponding time complexity of HSCE as T_{HSCE}, which is estimated as follows:

$T_{\mathrm{HSCE}}=T_{\mathrm{SSCA}}+T_{\mathrm{LSSC}}+T_{\mathrm{CSPA}}$.

where T_{SSCA} is the computational cost of the constraint-based SSCA algorithm and T_{LSSC} is the computational cost of the metric-based LSSC algorithm. T_{SSCA} and T_{LSSC} serves as the computational complexity of the original ensemble member generation algorithm. T_{CSPA} is the

computational complexity of the final ensemble decision generation algorithm.

Concretely, T_{SSCA} is affected by the number of instances n as follows: $T_{SSCA} = O(n\log n)$ [20]. For LSSC, step 1 requires to determine k cluster centers, and thus has a complexity of $O(k)$. Step 2 divide n instances into k clusters, therefore requires $O(nk)$. Step 3 recalculates the cluster centers and its complexity is $O(n)$. Step 2 and step 3 compose an iteration process and the max-number of iterations is t. So T_{LSSC} is estimated as follows: $T_{LSSC} = O(k) + O((O(nk) + O(n))t)$. The computational complexity of CSPA are as follows: $T_{CSPA} = O(n^2km)$ [2], where n is the number of instances, k denotes the number of clusters in the final result, and m denotes the number of the original ensemble members (base clusterings). Overall, the computational complexity of HSCE is approximately $O(n^2km)$.

The space complexity of HSCE mainly depends on the initial size of the data sets, and thus the space complexity is $O(mn)$, where m denotes the number of instances, n denotes the number of dimensions for general data sets, and **m*n** denotes the matrix of an image for image data sets.

4 Experiments

In this section, we apply the proposed approach to implement two groups of experiments. The first group makes some contrast tests on the proposed algorithm with the other related algorithms to verify clustering performance on general test data sets. The second group conducts comparison experiments for image data clustering comparing with several representative algorithms to evaluate the clustering result.

We use Matlab as the programing language in our experiment. The running environment consists of software and hardware. The configurations are described as follows. (a) Software: windows 64-bit operating system, MATLAB R2013a. (b) Hardware: CPU/Intel(R) Xeon(R), Graphics card/NVIDIA Quadro 5000, RAM/DDRA3 8 GB.

4.1 Comparison experiments on general data sets

4.1.1 Data set and experimental setting

In order to validate the effectiveness of our proposed clustering algorithm, eight data sets are selected from UCI Machine Learning Repository [35] for following considerations. Firstly, these data sets are widely used as benchmark data sets for machine learning and data mining research. In addition, these data sets enjoy different properties, of which three are binary-class and five are multiple-class.

Table 1 The characteristics of these data sets

Data set	Number of instances	Number of dimensions	Number of clusters
Segment	2310	19	7
Vowel	360	10	4
Ecoli	336	7	8
Grass	214	9	6
Iris	150	4	3
Ionosphere	351	34	2
Mushroom	8124	22	2
Statlog(heart)	270	13	2

Furthermore, the dimensional representations are with distributions ranging from low to high. The detailed description of these data sets are showed in Table 1.

In this section, the SMIP need to be ignored. So, the metric function of LSSC algorithm uses the general form, i.e., $\hat{D}_{ij} = \frac{1}{D(x_i,x_j)}$, where $D(x_i, x_j)$ is the LMNC distance metric between two instances. Some related parameters are set as reported in literature [20, 23]. In LSSC algorithm, the maximum of iterations is set as 150. The number of clusters k is set as the same as the ground-true class number. Next, we conduct three comparison experiments on UCI data sets. For each dataset, we repeat experiments for 20 trials.

On one hand, the number of supervised information (included pairwise constraints and labels) has effect on the result of semi-supervised clustering ensemble algorithm. In order to test the effectiveness of supervised information, we investigate the impact of supervised information on the performance of HSCE. In our experiments, we select the percent of supervised information as 0, 5, 10, 15, 20, 25% to analyze the effect of supervised information, where 0% means without any supervised information.

On the other hand, HSCE is compared with the following eight representation semi-supervised clustering algorithms. Two constraint-based clustering algorithms are LCVQE [17] and PMMC [19]. Two metric-based clustering algorithms are ITML [21] and LMNN [22]. Three hybrid clustering algorithms which combined constraint-based and metric-based are MPCKM [11], SCKMM [12] and ESCSSAP [15]. One semi-supervised clustering ensemble algorithm is MVSCE [27]. For the sake of fair comparison with other algorithms, we employ 20% of all the samples of each class as labeled samples, then those labeled samples are used to produce pairwise constraints in metric-based method. Meanwhile, we select

20% must-link constraints of all instances' must-link constraints and 20% cannot-link constraints of all instances' cannot-link constraints for each class in constraint-based method.

Furthermore, to obtain a better understanding of our work, we compare the proposed method with other semi-supervised clustering algorithms based on the average running times, included LCVQE [17], ITML [21], SMUC [10] and RSGrid [13]. In this part of experiment, we test the time performance of HSCE on the whole dataset and employ 20% of supervised information for each dataset.

4.1.2 Evaluation criterion

In our experiments, we adopt two popular evaluation criteria on the clustering performance.

a. Normalized mutual information (NMI for short). It reflects the coherence between the inferred clustering and the ground truth aggregation [36]. Let C be the random variable denoting the cluster assignments of data points, and K be the random variable denoting the underlying class labels, and then the NMI measure is defined as

$$NMI = \frac{2I(C;K)}{H(C) + H(K)}$$

where $I(X;Y) = H(X) - H(X|Y)$ is the mutual information between the random variables X and Y, $H(X)$ is the Shannon entropy of X, and $H(X|Y)$ is the conditional entropy of X given Y. The normalization by the average entropy of C and K makes the value of NMI stay between 0 and 1.

b. F-Measure. It evaluates the clustering result based on the underlying classes and considers the same-cluster pairs. Pairwise F-measure relies on the traditional information retrieval measures. It consists of precision index and recall index. The value of F-measure stays between 0 and 1. F-Measure is defined as

$$Precision = \frac{\#\text{Samples Correctly Predicted In Same Cluster}}{\#\text{Total Samples Predicted In Same Cluster}}$$

$$Recall = \frac{\#\text{Samples Correctly Predicted In Same Cluster}}{\#\text{Total Samples In Same Cluster}}$$

$$F - Measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

For both two evaluation indexes, the larger the value it is, the more similar the groupings by clustering and those by the ground true class labels.

4.1.3 Experimental results and analysis

To obtain a comprehensive understanding of our work, we add the number of supervised information in the experimental process for each data set gradually. For each case, the average NMI and F-measure values of HSCE are shown in Fig. 2. From these results, we get some interesting points. NMI and F-measure values of HSCE are considerably improved as the percentage of supervised information increases. The more supervised information make the better performance of semi-supervised clustering ensemble. This is due to the fact that more abundant labeled instances are, more reliable the metric measure is. At the same time, the more pairwise constraints are, the more completely the violation issue of pairwise constraints is solved. As a result, HSCE achieve better performance through adding prior knowledge.

Tables 2 and 3 separately summarize the NMI and F-Measure performance of the proposed semi-supervised clustering algorithm and the other eight algorithms based on the same proportion of constraints over different data sets. In the both tables, the top highest NMI and F-measure scores are highlighted in bold font. Table 4 reports the average running times of all the algorithms. From these comparison results, these semi-supervised clustering algorithms demonstrate the significantly different degree on clustering performance. The detailed analysis of this group experiment is summarized as follows.

Firstly, as a whole, we observe that SCKMM, E-SCSSAP, MVSCE and HSCE can consistently achieve comprehensive better-performance with higher value on both two indexes, and outperform other algorithms obviously. This is due to the fact that a combination method of both pairwise constraints and metric measure can obtain performance superior to other traditional single clustering methods (either constraint-based or metric-based). MVSCE and HSCE work well than other algorithms distinctly, we can verify that the cluster ensemble approaches are able to integrate multiple clustering solutions and provide a more accurate, robust and stable final result when compared with standard clustering algorithms.

Secondly, as can be seen from results, the two metric-based methods achieve a little higher performance than the two constraint-based methods on most data sets. This observation can be explained that sufficient pairwise constraints and labeled instances are good for measuring the distance between instances with an accurate metric function, which can easily enforce samples from the same cluster closely and samples from different clusters far apart.

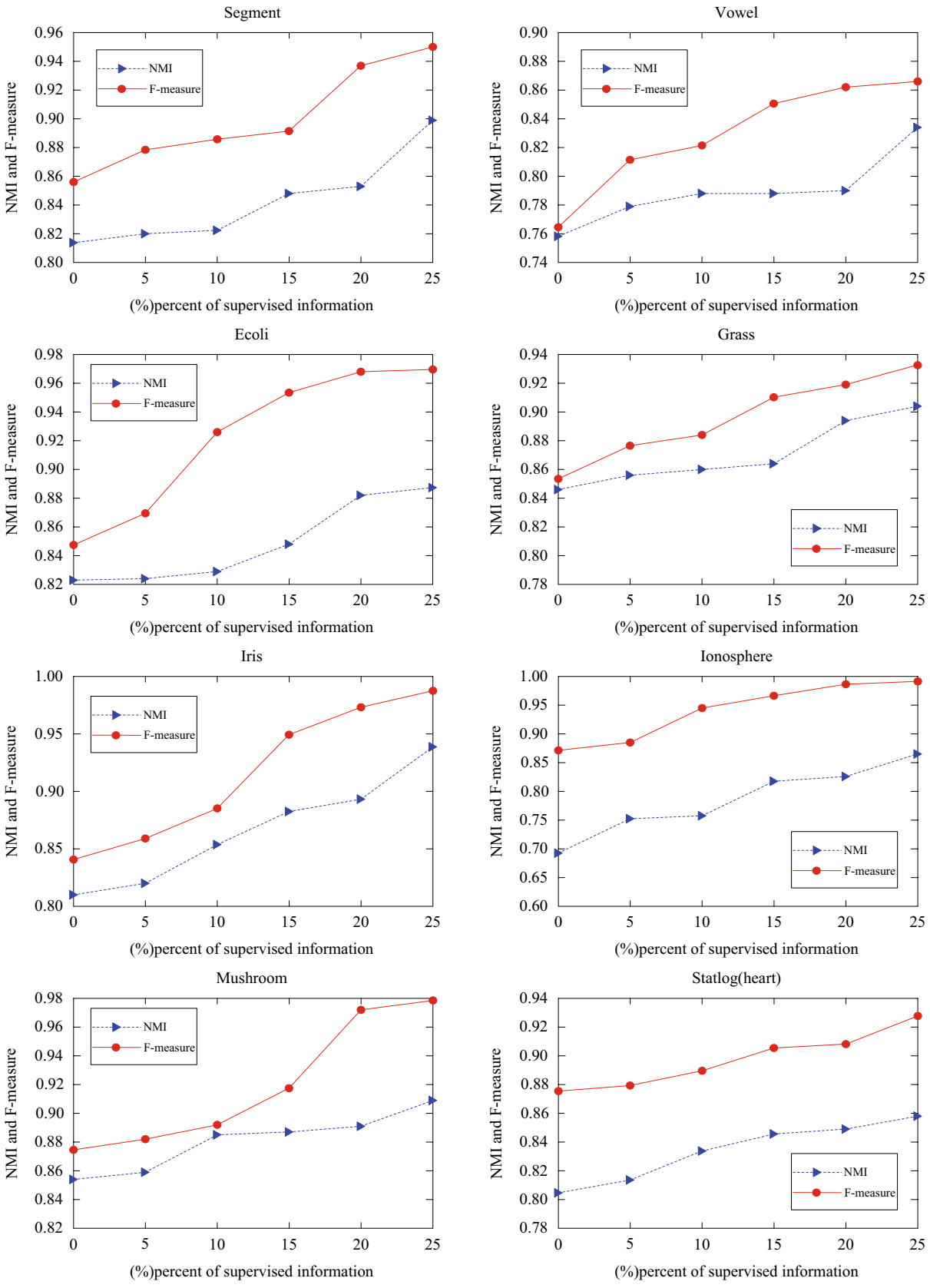

Fig. 2 NMI and F-measure values with supervised information increasing

Table 2 Evaluation of clustering performance on NMI for nine algorithms

Data set	LCVQE	PMMC	ITML	LMNN	MPCKM	SCKMM	E-SCSSAP	MVSCE	HSCE
Segment	0.696	0.734	0.685	0.758	0.534	0.837	0.847	0.848	**0.853**
Vowel	0.592	0.669	0.577	0.603	0.557	0.795	**0.816**	0.784	0.790
Ecoli	0.716	0.802	0.666	0.620	0.642	0.831	0.879	0.796	**0.882**
Grass	0.579	0.668	0.646	0.719	0.508	0.863	0.865	0.855	**0.894**
Iris	0.699	0.854	0.902	0.854	0.683	0.894	0.888	**0.903**	0.893
Ionosphere	0.737	0.729	0.779	0.761	0.737	0.786	0.811	0.779	**0.826**
Mushroom	0.815	0.854	0.860	0.853	0.726	**0.904**	0.872	0.883	0.891
Statlog (heart)	0.713	0.706	0.714	0.763	0.779	0.812	0.834	**0.862**	0.849

Table 3 Evaluation of clustering performance on F-Measure for nine algorithms

Data set	LCVQE	PMMC	ITML	LMNN	MPCKM	SCKMM	E-SCSSAP	MVSCE	HSCE
Segment	0.768	0.732	0.721	0.728	0.731	**0.969**	0.923	0.928	0.937
Vowel	0.679	0.744	0.749	0.762	0.694	0.843	0.836	0.859	**0.861**
Ecoli	0.828	0.957	0.960	0.971	0.788	0.965	0.971	**0.976**	0.968
Grass	0.624	0.642	0.664	0.697	0.671	0.851	0.875	0.847	**0.919**
Iris	0.771	0.926	0.963	0.933	0.888	0.967	0.958	0.966	**0.973**
Ionosphere	0.976	0.879	0.892	0.980	0.926	0.945	0.979	0.981	**0.987**
Mushroom	0.890	0.971	**0.977**	0.974	0.869	0.976	0.973	0.976	0.972
Statlog(heart)	0.852	0.856	0.864	0.892	0.753	0.894	0.900	0.897	**0.908**

Table 4 Comparison of average running times (s)

Data set	Algorithm	Time
Segment	ITML	0.13
	HSCE	7.65
Vowel	ITML	0.04
	SMUC	18.78
	HSCE	1.47
Ecoli	LCVQE	0.01
	HSCE	1.29
Grass	LCVQE	<0.01
	HSCE	0.72
Iris	LCVQE	<0.01
	ITML	0.03
	HSCE	0.58
Ionosphere	LCVQE	<0.01
	ITML	0.18
	SMUC	2.10
	HSCE	1.32
Mushroom	RSGrid	25.16
	HSCE	23.84
Statlog(heart)	HSCE	0.83

Thirdly, although HSCE fails to achieve the best performance on few date sets, its values still extremely close to the best ones. This can be attributed to the fact that the proposed HSCE algorithm is considerably effective in utilizing the pairwise constraints and learning accurate metric measure to improve clustering performance. In addition, HSCE refers to combine a number of base partitions for a particular data set into a consensus clustering solution. Cluster ensemble sufficiently improved the clustering result of individual clustering algorithm. Thus, the HSCE algorithm provides an appealing clustering performance with high accurate, stable and meaningful.

Fourthly, two different evaluation metric are used to evaluate results, namely NMI and F-measure. The two index results are not always consistent on all data sets. For example, on the same data set Vowel, the NMI value of E-SCSSAP is slightly higher than HSCE. And yet, the F-measure performance of HSCE is better than E-SCSSAP. This phenomenon indicates the chosen of evaluation index is important for evaluating the performance of clustering algorithm. Adopting more than one evaluation index can measure clustering results more accurately and comprehensively.

In this section, the average time consumptions for 20 trials of each clustering algorithm are reported in Table 4. In addition to the aforementioned factors about running environment, the runtime of proposed method is related to data sets, the attributes of data sets, the number of iterations and the number of clusters, and so on. From these experimental results, it can be seen that the time performance of HSCE is generally efficient and considerable within an acceptable time. Although it is not the fastest algorithm on most data sets, we can learn from the experimental results that HSCE

Table 5 The eight chosen original images

Image 1	Image 2	Image 3	Image 4
			
Image 5	**Image 6**	**Image 7**	**Image 8**
			

works stably. Furthermore, it demonstrates the proposed method is feasible, reasonable and ideal on the efficiency.

4.2 Comparison experiments on image data sets

4.2.1 Data set and experimental setting

To illustrate the implementation of the proposed method for image pixels clustering, eight images in the size of 481 × 321 are randomly chosen from the Berkeley Segmentation Data Set (BSDS500) [37], as showed in Table 5. For comparison, we also implement image data clustering using the following four methods. (a) K-means, an unsupervised clustering algorithm. (b) LSSC, a metric-based semi-supervised clustering algorithm proposed in this paper. (c) SSKFCM [14], a semi-supervised clustering approach using the kernel-based method based on KFCM; (d) BSMSCE [26], a semi-supervised clustering ensemble algorithm.

Feature extraction is a foundational step and a premise to generate satisfied clustering results for image data. In the experiments, we extract two parts of characteristics of an image, namely immanent characteristic and spatial information. On one hand, the immanent characteristic we need to extract concretely included the color feature at one pixel site, the surrounding texture feature LBP value, and the color gradient which denotes sudden change when meeting region edge or noise pixels. On the other hand, from the perspective of spatial information, we extract the color feature of pixels based on CIE L*A*B* space and the edge information of image. The result of feature extraction fits more closely with human perception.

In this section, LMNC and SMIP are combined to form the metric function of LSSC algorithm, which is conveyed as $\hat{D}_{ij} = \frac{S_{ij}}{D(x_i,x_j)}$. The LMNC metric method for SSCA algorithm mainly measures based on the extracted immanent characteristic, which are the color gray value at one pixel site, the color gradient value calculated by Sobel operator and the texture feature LBP value of neighborhood pixels (with 3 × 3 windows). Meanwhile the SMIP metric method considers the patch-based color feature similarity and the edge-based spatial feature similarity. In other words, the extracted spatial information is used to compute the SMIP metric.

Likewise, the experimental conditions are set as reported in literature [20, 23, 30]. Firstly, the maximum of iterations is set as 150 in LSSC algorithm. In addition, we need to provide some information. We randomly sample some pixels from different partitions of an image as labeled data. Those labeled pixels are used in metric-based LSSC algorithm. The pixels with the same label indicate they are similar, while with different labels dissimilar. Then the set of must-links M and the set of cannot-links C can be obtained respectively. Those constraint information are used in constraint-based SSCA algorithm. The number of clusters k is set as the same as the ground-true class number.

4.2.2 Evaluation criterion

Here, we set the number of clusters equal to the number of ground-true clusters. Clustering accuracy (ACC) metric is used to compare the clusters generated by these algorithms with the ground-true clusters to evaluate the image data clustering performance [38]. Clustering accuracy discovers the one-to-one relationship between clusters and ground-true clusters and measures the extent to which each cluster contain data points from the corresponding ground-true cluster. It sums up the whole matching degree between all pair ground-true cluster-clusters. ACC can be computed as

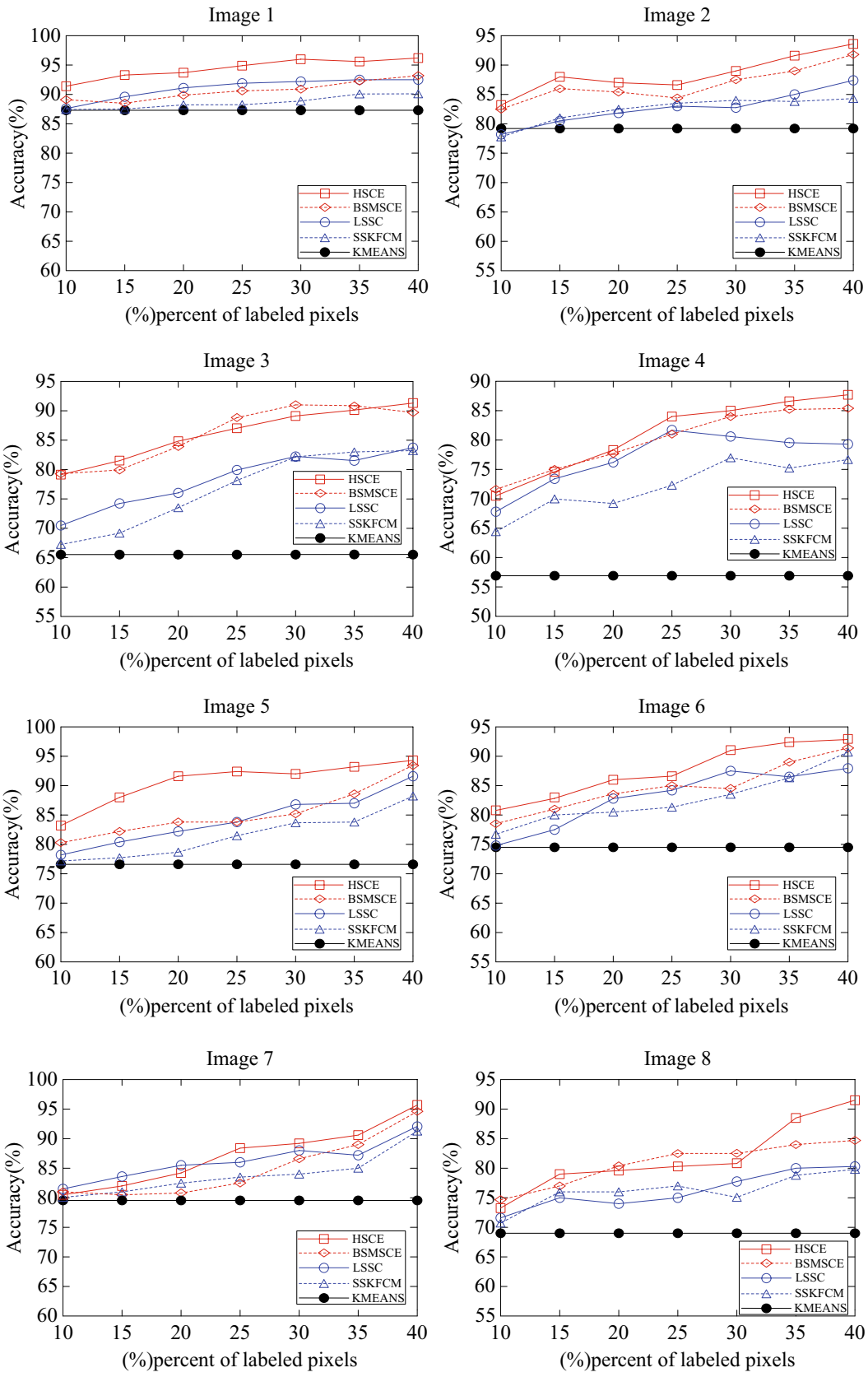

Fig. 3 Clustering performance with different percent of labeled pixels

Table 6 ACC results with 40% of labeled pixels

Image	Unsupervised clustering	Semi-supervised clustering	Semi-supervised cluster ensemble	Ours	
	K-means	SSKFCM	BSMSCE	LSSC	HSCE
1	0.873	0.901	0.932	0.925	0.962
2	0.792	0.843	0.918	0.874	0.936
3	0.655	0.832	0.897	0.837	0.913
4	0.569	0.767	0.854	0.793	0.876
5	0.766	0.882	0.934	0.916	0.943
6	0.745	0.907	0.914	0.879	0.929
7	0.796	0.913	0.946	0.921	0.957
8	0.690	0.790	0.847	0.803	0.915

$Acc = \frac{1}{N} \max\left(\sum_{C_k, L_m} T(C_k, L_m)\right)$, where N is the total number of pixels in an image, C_k denotes the k-th cluster in final results, and C_k denotes the ground-truth m-th cluster. From the perspective of an image, $T(C_k, L_m)$ is the number of pixels belonging to ground-true cluster m that are assigned to cluster k. ACC computes the maximum sum of $T(C_k, L_m)$ for all pairs of clusters and ground-true clusters, and these pairs have no overlaps. The greater clustering accuracy means the better clustering performance.

4.2.3 Experimental results and analysis

Figure 3 illustrates the graphical ACC results of the five methods with different numbers of labeled pixels. As the percent of labeled pixels increases, the unsupervised clustering K-means algorithm is used as the baseline clustering, whose performance is still a steadily numerical value. While the accuracy of other semi-supervised clustering algorithms reveals gradually increasing trend as a whole with the increase of labeled data. What's more, Table 6 displays the experiment results of image data clustering performance on ACC with 40% of labeled pixels, which shows the accuracy of the five algorithms with numerical value intuitively. From these results, we obtain several attractive insights.

Firstly, we observe that SSKFCM and LSSC often achieve the better performance than K-means on the eight images. K-means gets better grade only with few supervised information on image 2. But when the percent of labeled pixels reaches a certain number, both SSKFCM and LSSC algorithms behave better and better, and far exceeds K-means. It demonstrates that comparing with unsupervised clustering, semi-supervised clustering methods can improve the clustering accuracy by effectively exploring the available information that is usually in the form of pairwise constraints and instance labels. As a result, the semi-supervised clustering algorithms outperform other traditional unsupervised clustering methods.

Secondly, we can observe that the performance of the LSSC algorithm is always comparable to, and better than the SSKFCM algorithm when there are enough labeled information, even better than BSMSCE and HSCE few times. It can be understandable that the proposed LSSC method takes into full account of the intrinsic properties and spatial information of each pixel so that the metric function can measure the similarity more accurately, and easily make points from different clusters far apart and those from the same clusters closely. Thus, LSSC provides a more satisfying clustering results.

Thirdly, BSMSCE and HSCE can consistently outperform better than SSKFCM and LSSC in most of our experiments. This observation can be explained by the fact that both BSMSCE and HSCE respectively have a cluster ensemble mechanism in the process of determining accurate and robust clustering decision from an ensemble of base clustering results. In other words, cluster ensemble combined different solutions of various clustering methods can achieve accuracy superior to those of individual clustering.

Fourthly, compare to HSCE, the performance of BSMSCE has a little advantage on image 3, image 4 and image 8, and the performance of LSSC behave a little advantage on image 7. Nonetheless, for overall perspective, HSCE displays better than BSMSCE and LSSC in most cases. In view of image peculiarity, HSCE also obtains more supervised information by propagating limited pairwise constraints and considerably effective in utilizing the pairwise constraints, so that it not only performs the metric accurately, but also solve the violation issue effectively. Thereby, the clustering performance of HSCE is sufficiently improved.

It is worth noting that HSCE improves the robustness as well as the quality of clustering results. HSCE nearly always achieves the best performance with steady development trend on the eight images. This observation can be explained by the fact summing up the above.

5 Conclusion

In this paper, we present a novel semi-supervised clustering ensemble approach for data clustering. Our method is different from the previous studies that integrates the constraint-based method and the metric method into a semi-supervised clustering ensemble approach in the hope of gaining the more optimal accuracy, robustness and stability of clustering dramatically. Specifically, we construct a

new metric function with two forms in our proposed metric-based semi-supervised clustering algorithm. One is for general data clustering based on the LMNC distance metric. The other is for image data clustering by combining the similarity of image pixels with the LMNC metric from the image perspective, concretely it builds collections of the inherent attributes and spatial information of pixels, which efficiently and accurately reflects the relationship between image pixels. Moreover, we conduct two group comparison experiments, respectively on general data sets and image data sets. Multiple comparison results indicate that this proposed scheme can achieve better clustering performance than a number of competing clustering algorithms on the whole. Empirically as well as theoretically, it confirms the feasibility and effectiveness of the proposed method with encouraging results.

However, what we are done still leaves much to be desired. For instance, we should add noise factors into the experiments to test the sensitivity of clustering algorithms to noise, which can reveal the stability and robustness of clustering approaches. At the same time, there are a great many interesting directions to extend our work. As we all known, clustering is often viewed as a foundation technology in image process and computer vision. To investigate further, our future work will develop clustering into the mapping process from low-level features of images to high-level semantic comprehension in conjunction with other related techniques.

Acknowledgements We would like to thank the anonymous reviewers for their insightful comments and suggestions to significantly improve the quality of this paper. This research reported in this paper is supported by the National Natural Science Foundation of China (Nos. 61165009, 61663004, 61262005, 61363035, 61365009), the Guangxi Natural Science Foundation (2016GXNSFAA380146, 2014GXNSFAA118368), the Direct Fund of Guangxi Key Lab of Multi-source information Mining and Security (16-A-03-02),the Guangxi "Bagui Scholar" Teams for Innovation and Research Project, Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing.

References

1. Wu L, Hoi S C H, Jin R, Zhu J, Yu N (2010) Learning bregman distance functions for semi-supervised clustering. IEEE Trans Knowl Data Eng 24(3):478–491
2. Strehl A, Ghosh J, Cardie C (2003) Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J Mach Learn Res 3:583–617
3. Du L, Shen YD, Shen Z, Wang J, Xu Z (2013) A self-supervised framework for clustering ensemble. Lect Notes Comput Sci 7923:253–264
4. Hao ZF, Wang LJ, Cai RC, Wen W (2015) An improved clustering ensemble method based link analysis. World Wide Web-internet & Web. Inform Syst 18(2):185–195
5. Yu Z, Chen H, You J, Wong HS, Liu J, Li L et al (2014) Double selection based semi-supervised clustering ensemble for tumor clustering from gene expression profiles. IEEE/ACM Trans Comput Biol Bioinform 11(4):727–740
6. Yu Z, Luo P, You J, Wong HS, Leung H, Wu S et al (2016) Incremental semi-supervised clustering ensemble for high dimensional data clustering. IEEE Trans Knowl Data Eng 28(3):701–714
7. Xiong S, Azimi J, Fern XZ (2014) Active learning of constraints for semi-supervised clustering. IEEE Trans Knowl Data Eng 26(1):43–54
8. Wang D, Gao X, Wang X (2015) Semi-supervised nonnegative matrix factorization via constraint propagation. IEEE Trans Cybern 46:1–12.
9. Yan Y, Chen L, Nguyen D T (2012) Semi-supervised clustering with multi-viewpoint based similarity measure. IEEE Int Jt Conf Neural Netw (IJCNN), 24, 1–8.
10. Yin X, Shu T, Huang Q (2012) Semi-supervised fuzzy clustering with metric learning and entropy regularization. Knowl-Based Syst 35(15):304–311
11. Bilenko M, Basu S, Mooney RJ (2004) Integrating constraints and metric learning in semi-supervised clustering. The 21st International Conference on Machine Learning, 81–88.
12. Yin X, Chen S, Hu E, Zhang D (2010) Semi-supervised clustering with metric learning: an adaptive kernel method. Pattern Recognit 43(4):1320–1333
13. Lin L, Qu W, Yu X (2009) A semi-supervised clustering algorithm based on rough reduction. International Conference on Chinese Control and Decision Conference, 5427–5431.
14. Zhang H, Lu J (2009) Semi-supervised fuzzy clustering: a kernel-based approach. Knowl-Based Syst 22(6):477–481
15. Arzeno N, Vikalo H (2015) Semi-supervised affinity propagation with soft instance-level constraints. IEEE Trans Pattern Anal Mach Intell 37(5):1041–1052
16. Basu S, Banerjee A, Mooney RJ (2002) Semi-supervised clustering by seeding. In: Proceedings of the nineteenth international conference on machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 27–34
17. Pelleg D, Baras D (2007) K-means with large and noisy constraint sets. In: Kok JN, Koronacki J, Mantaras RL, Matwin S, Mladenič D, Skowron A (eds) Machine learning: ECML 2007. Lecture notes in computer science, vol 4701. Springer, Berlin, Heidelberg, pp 674–682
18. Grira N, Crucianu M, Boujemaa N (2008) Active semi-supervised fuzzy clustering. Pattern Recognit 41(5):1834–1844
19. Zeng H, Cheung Y M, Member S (2012) Semi-supervised maximum margin clustering with pairwise constraints. IEEE Trans Knowl Data Eng 24(5):926–939
20. Ding S, Jia H, Zhang L, Jin F (2014) Research of semi-supervised spectral clustering algorithm based on pairwise constraints. Neural Comput Appl 24(1), 211–219.
21. Davis JV, Kulis B, Jain P, Sra S, Dhillon IS (2007) Information-theoretic metric learning. In: Proceedings of the 24th international conference on Machine learning. ACM, New York, pp 209–216
22. Weinberger KQ, Blitzer J, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. J Mach Learn Res 10(1):207–244
23. Huang M, Chen Y, Liu J, Ji W (2014) A large margin nearest cluster metric based semi-supervised clustering algorithm for brain fibers. International Conference on Game Theory for Networks, 1–5.
24. Nguyen N, Caruana R (2007) Consensus clusterings. In: Proceedings of the 7th IEEE international conference on data mining. IEEE Computer Society, Washington, DC, pp 607–612

25. Wang X, Han D, Han C (2013) Rough set based cluster ensemble selection. In: Proceedings of 16th International Conference on Information Fusion (FUSION). IEEE, Istanbul, Turkey, pp 438–444
26. Wang H, Qi J, Zheng W, Wang M (2010) Semi-supervised cluster ensemble based on binary similarity matrix. IEEE International Conference on Information Management and Engineering, 251–254.
27. Chen D, Yang Y, Wang H, Mahmood A (2013) Convergence analysis of semi-supervised clustering ensemble. International Conference on Information Science and Technology (ICIST), 783–788.
28. Zhang D, Tan K, Chen S (2004) Semi-supervised kernel-based fuzzy c-means. In: Lecture notes computer science, vol 3316, pp 1229–1234
29. Bertsekas DP (1976) On the goldstein-levitin-polyak gradient projection method. IEEE Trans Autom Control 21(2):174–184
30. Na Y, Yu J (2013) A pixel similarity method for spectral clustering image segmentation. J Nanjing Univ Nat Sci 2:159–168
31. Fowlkes C, Martin D, Malik J (2003) Learning affinity functions for image segmentation: combining patch-based and gradient-based approaches. In: IEEE conference on computer vision and pattern recognition, vol 2, pp 54–61
32. Cour T, Bénézit F, Shi J (2005) Spectral segmentation with multiscale graph decomposition. IEEE Comput Soc Conf Comput Vis Pattern Recog 2:1124–1131
33. Martin D, Fowlkes C, Malik J (2004) Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans Pattern Anal Mach Intell 26(5):530–549
34. Sun T, Ren Z, Ding S (2011) Region-based semi-supervised clustering image segmentation. Int Conf Nat Comput 4:1855–1858.
35. Lichman M (2013) UCI Machine Learning Repository. University of California, Irvine, CA School of Information and Computer Science. doi:http://archive.ics.uci.edu/ml.
36. Kuncheva L, Hadjitodorov S B (2004) Using diversity in cluster ensembles. IEEE Int Conf Syst Man Cybern 2:1214–1219.
37. Arbeláez P, Maire M, Fowlkes C, Malik J (2011) Contour detection and hierarchical image segmentation. IEEE Trans Software Eng 33(5):898–916
38. Wang F, Zhang C, Li T (2009) Clustering with local and global regularization. IEEE Trans Knowl Data Eng 21(12):1665–1678