Journal of Software Engineering

ISSN 1819-4311

Academic Journals Inc.

www.academicjournals.com

Journal of Software Engineering 8 (1): 14-22, 2014
ISSN 1819-4311 / DOI: 10.3923/jse.2014.14.22

Adaptive Semi-Supervised Clustering Algorithm with Label Propagation

Mingwei Leng, Jinjin Wang, Jianjun Cheng, Hanhai Zhou and Xiaoyun Chen
School of Information Science and Engineering, Lanzhou University, Tianshui South Road 222, Lanzhou, China

Corresponding Author: Xiaoyun Chen, School of Information Science and Engineering, Lanzhou University, Tianshui South Road 222, Lanzhou, China

ABSTRACT

Semi-supervised clustering which uses the limited labeled data to aid unsupervised clustering, has become a hot topic in recent years. But the limited labeled data may be imbalanced and can not cover all clusters in some cases and most of the existing semi-supervised clustering algorithms can not deal with imbalanced dataset well and have no the ability of detecting new clusters. In view of this, an adaptive semi-supervised clustering algorithm with label propagation is proposed. Two most of interesting characteristics of the proposed algorithm are that (1) It uses the limited labeled data to expand labeled dataset based on an adaptive threshold by labeling their k-nearest neighbors, (2) It detects whether there exist new clusters in the unlabeled dataset according to a proposed measure criterion. Three standard datasets are used to demonstrate the performance of the proposed algorithm and the experimental results confirm that the accuracy of the proposed clustering algorithm is much higher than that of three compared algorithm and in addition the proposed algorithm has the ability of detecting new clusters.

Key words: Data mining, semi-supervised clustering, K-nearest neighbors, label propagation

INTRODUCTION

A recent trend in data mining and machine learning researches is combining the techniques developed for unsupervised learning and supervised learning to handle datasets with background knowledge. Semi-supervised clustering is one of the foci and it improves the performance of clustering by learning from the labeled data. Semi-supervised k-means clustering (Wagstaff *et al*., 2001; Basu *et al*., 2002, 2004; Leng *et al*., 2008; Dang *et al*., 2010) and density-based clustering are two important methods (Lelis and Sander, 2009; Ruiz *et al*., 2010; Zhao *et al*., 2012). Wagstaff *et al*. (2001) proposed two kinds of pairwise constraints: the must-link and cannot-link and made the domain knowledge into k-means clustering. Basu *et al*. (2002) exploited labeled data to generate initial seed clusters. Basu *et al*. (2004) utilized EM and Hidden Markov Random Fields to propose a semi-supervised clustering algorithm, HMRF-KMEANS. Leng *et al*. (2008) used labeled data to initialize the process of k-means clustering and obtained the similarity threshold of clusters based on the label information, utilized similarity threshold to guide k-means clustering. Dang *et al*. (2010) presented a novel initialization method by propagating the labels of labeled data to more unlabeled data.

Semi-supervised density-based clustering is another popular semi-supervised clustering method (Lelis and Sander, 2009; Ruiz *et al*., 2010; Zhao *et al*., 2012). Lelis and Sander (2009) exploited labeled data to find values for ε, given a fixed value of MinPts and used minimal spanning tree to partition dataset. Ruiz *et al*. (2010) proposed a semi-supervised clustering C-DBSCAN, which partitioned data space into denser subspace to build a set of initial local clusters, used the must-link constraints to merge density-connected local clusters and merged adjacent neighborhoods while remaining cannot-Link constraints. Zhao *et al*. (2012) proposed a document clustering, Constrained DBSCAN (Cons-DBSCAN), which selected informative document pairs for obtaining user feedback by using active learning approach and incorporated instance-level constraints to guide the clustering process in DBSCAN.

Laterly, researchers also paid attention to semi-supervised hierarchical clustering (Bohm and Plant, 2008), semi-supervised graph clustering (Kulis *et al*., 2009), semi-supervised clustering based on kernel approach (Yin *et al*., 2010; Baghshah and Shouraki, 2009, 2010). Bohm and Plant (2008) expanded the clusters starting at all labeled objects simultaneously. Kulis *et al*. (2009) proposed a new semi-supervised clustering algorithm, SS-KERNEL-KMEANS, which partitioned vector-based and graph-based data by optimizing a semi-supervised clustering objective. Yin *et al*. (2010) tried to solve the violation issue of constrains to propose an adaptive Semi-supervised Clustering Kernel Method (SCKMM), which estimated the parameter of the Gaussian kernel automatically (Baghshah and Shouraki, 2009, 2010) also exploited the method of metric learning to improve the performances of semi-supervised clustering algorithms.

Since the size of labeled data is very small, some clusters may have no data with label and the distribution of labeled data in a given dataset is not same as the whole data space. How to detect those clusters which have no labeled data is not an easy work and it is the problem to be solved in this study. The k-nearest labeled data of an unlabeled data object may not be in the same clusters, most of existing semi-supervised learning algorithms assign this unlabeled data object with a wrong label. However, on the whole data space, the labels of data should be the same as their k-nearest neighbors. In addition, although the size of the labeled data is very small and this leads that they can not cover all the clusters, the labeled data give some priori knowledge about the dissimilarity between clusters, the minimum value of the distances between core objects in different clusters is used to determine whether there needs to increase new clusters. In this study, an adaptive semi-supervised clustering algorithm is proposed based on the facts above. The proposed clustering algorithm has mainly the following two advantages in comparison with other semi-supervised clustering:

- The proposed clustering algorithm achieves label propagation by using the labeled data to expand their k-nearest neighbors according to a criteria which is automatically obtained based on the characters of the given datasets and the expanded model only requires one parameter
- When the size of labeled data is very small, especially for the number of labels is less than the real number of clusters in the given dataset, the proposed method obtains the dissimilarities between clusters by using the distances between core objects in different clusters and uses the dissimilarities to detect whether there exists new cluster automatically, if there exist new clusters, it increases new clusters one by one

MATERIALS AND METHODS

In order to describe the proposed semi-supervised clustering algorithm simply, some definitions of concepts are given as follows.

Definition 1: KNN(x). Given one data object x, KNN(x) is the set of k nearest neighbors of x in C and KNN(x, j) denotes the jth nearest neighbor of x.

Definition 2: k_dis(·). Given a dataset D and one data p, p∈D, k_dis(p) is defined as Eq. 1:

$$k\_dis(p) = \min_{x \in KNN(p)} \{dis(p,x)\} \quad (1)$$

and k_dis(D) is the distance set which is constructed by the all k_dis(P).

Definition 3: Core objects. Given a dataset D, one data p and an integer k, p∈D, if k_dis(p)≥avg(k_dis(D)), then p is a core object. Where the meaning of k_dis(p) and k_dis(·) are the same as definition 1 and avg(k_dis(D)) is the average of k_dis(D).

Definition 4: dis(C_i, C_j). Given two clusters C_i and C_j, dis(C_i, C_j) is defined as Eq. 2:

$$dis(C_i, C_j) = \min \{dis(x, y) \mid x \in Core(C_i), y \in Core(C_j)\} \quad (2)$$

where Core (C_i) and Core (C_j) are the core objects set in C_i and C_j, respectively.

Definition 5: dis(p, C_i). Given one cluster C_i and one data p($p \notin C_i$), dis(p, C_i) is defined as Eq. 3:

$$dis(p, C_i) = \min \{dis(p, x) \mid x \in Core(C_i)\} \quad (3)$$

where Core (C_i) are all the core objects in C_i.

Definition 6: dis(p, C). Given a cluster set C and one data p($p \notin C_i$), where $C = C_1 \cup C_2 \cup ... \cup C_k$, dis(p, C) is defined as Eq. 4:

$$dis(p,C) = \min_{i \in \{1,2,...,k\}} \{dis(p,C_i)\} \quad (4)$$

where the meaning of dis(p, C_i) is the same as definition 5.

Semi-supervised clustering algorithm with label propagation: In general, the distribution of labeled data in a given dataset is not the same as the whole data space, especially for the imbalanced dataset. A data point and its majority k-nearest labeled data may not be in the same cluster, which leads to the result that most of the existing semi-supervised learning algorithms can not work well, especially when the size of labeled dataset is very small. However, in the whole data space, the label of a data point should be the same as that of its majority k nearest neighbors. The proposed semi-supervised clustering with label propagation is based on this idea and it expands the

labeled dataset by labeling k nearest neighbors of labeled dataset. Once an unlabeled data is labeled, it is added into labeled dataset. If the difference of density between clusters is large in multi-density datasets, the expanding process can not use the same threshold and the threshold should be generated automatically according to the density of each cluster which the labeled data point belongs to. The proposed semi-supervised clustering algorithm uses a threshold to expand the neighbors of each labeled data and the threshold is generated automatically based on the cluster which the labeled data belongs to. The detail of the process of label propagating is shown in algorithm 1.

Algorithm 1: Semi-supervised clustering algorithm with label propagation

1. Input datased D(x_1, x_2,...,x_n), labeled dataset Dl, and the value of parameters k.
2. Let flag = 0, which contains n element.
3. Calculate dis_knn, dis_knn(i, j)(i, j)(i≤n, j≤k) is the distance between I-th date and its j-th nearset neighbor.
4. Let *avgdis* denote the average of k-th column of *dis*_knn.
5. Find out the core labeled dataset *core_labeled_data based* on *avagdis*, and let *core_labeled_num* denote the number of core labeled data.
6. Find out the core unlabeled dataset *core_unlabeled_data based* on *avagdis*, and let *core_unlabeled_num* denote the number of core unlabeled data.
7. Take one data x_i with *flag*(i) = 0 from the dateset Dl
8. For j = 1:k
9. If KNN(x_i, j) is unlabeled and *dis_knn*(i, j)≤*avgdis*
10. Use the label of x_i to label its j-*th* neighbor and add its j-*th* neighbor into Dl.
11. If the j-*th* neighbor of x_i is core object
12. Remove it from *core_unlabeled_data* to *core_labeled_data* and let *core_unlabeled_num* core_*unlabeled_num*, -1, *core_labeled_num* = core_labeled_num+1
13. End If
14. End IF
15. End For
16. Set *Flag*(i) = 1, which denotes that x_i has been used to labeled its *k-nearestneighnord*.
17. If There exists one data denotes x_j with the *flag*(j) = 0 in Dl
18. Use the same method of steps 8-15 to deal with KNN (x_j).
19. Een IF
20. Repeat the steps 17-19 until each dta in Dl has been used to deal with its *k nearest* neighbors.
21. Return the D*l*, *core _labeld_data* and *core_unlabeled_data*

Method for detecting new clusters: The proposed algorithm uses the distances between clusters to detect whether there exist new clusters in the rest unlabeled data. If the distance between two clusters is calculated by all the data in these two clusters, then the data objects lie in the boundary determine it. The boundaries of clusters are vague in some datasets, so it is suitable to use all data to calculate the distances between clusters. In order to measure the distances between clusters better, core objects are used to solve the above problem. Since the core objects do not lie in the boundaries of clusters, they are used to define the distances (dissimilarities) between clusters, which mainly eliminates the influence of boundary data. Labeled data are viewed as the priori knowledge and they are used to expand the labeled data. If the labeled data can not cover all clusters, then there exist some clusters which have not one labeled data. If the proposed algorithm does not detect new clusters, then the data in these clusters will be assigned to other clusters which have labeled data forcibly. This subsection tries to detect new clusters by using core objects and proposes an algorithm for detecting new cluster. If there exists one or more core objects have not been labeled, the proposed algorithm calculates dis(C_i, C_j) ($C_i \neq C_j$ and they are the existing clusters) and dis(p, C) (C is the set of existing clusters and p is an unlabeled core object), utilizes dis(C_i, C_j and dis(p, C) to determine whether there exist new clusters. The detail of description for detecting new cluster is given in algorithm 2.

Algorithm 2: Detecting new clusters

1. Input dataset D (x_1, x_2, ..., x_n).
2. Running algorithm 1 to get D_l avgdis, core_labeled_data and core_unlabeled_data.
3. Suppose that the number of different labeled data is m, partition D_l into m cluters C_1, C_2, ..., C_m according to the labels of data in D_l
4. While core_unlabeled_data is not null
5. For i = 1:m
6. For j = 1:m
7. dis (C_i, C_j) = min {dis (x,y)|x∈Core (C_i), y∈core ()C_j)}
8. End for
9. End for
10. MinClsDist = $\min_{1\le i, j\le m, I\ne j}$ dis (C_i, C_j)
11. Max_dis = $\max_{x\in core\_unlabeled\_data}$ dis (x,C) ($C = \cup C_2 \cup, \cdots \cup C_m$)

    $x_i \leftarrow \max\_dis_{x\in core\_unlabeled\_data}$
12. If dis (x_i, C) ≥ 2×max (MinClsDist, avgdis)
13. m = m+1, m is the label of new cluster C_m, add x_i into C_m.
14. core_unlabeled_data = core_unlabeled_data\{x_i}, core_unlabeled_num = core_unlabeled_num-1.
15. core_unlabeled_data = core_unlabeled_data∪{x_i}, core_labeled_num = core_labeled_num +1.
16. Else
17. Break.
18. End If
19. Use the steps 7-20 of algorithm 1 to expand the C_m.
20. End While
21. Partition Dl into clusters C_1, C_2, ...,C_m according to the labeles od data in Dl.
22. Deal with the rest of unlabeled data in D, assign the unlabeled data the cluster which is most similar with it
23. Return the clusters C_1, C_2,...,C_m

Algorithm 2 detects new clusters by comparing dis(x_i, C) with 2*max(Min ClsDist, avgdis). Algorithm 2 adds a new cluster if dis(x_i, C)≥2*max(Min ClsDist, avgdis), which means that there exist at least one unlabeled core object is far enough from the existing clusters and it should be in a new cluster. Once a new cluster is generated, the labels of the core object and the new cluster are given. Algorithm 2 uses the labeled data in the new cluster to expand it. The proposed algorithm adds new clusters one by one until the condition increasing new clusters does not hold.

RESULTS

Three UCI datasets (Bache and Lichman, 2013), IRIS, Wine and Page Blocks are used to demonstrate the proposed semi-supervised clustering algorithm and its performance compared with that of a semi-supervised clustering algorithm SSDBSCAN which is a novel method and uses the labeled data to guide the process of clustering (Ruiz *et al.*, 2010). In order to show the accuracy of the proposed method can reach that of some classification algorithms, the proposed method is compared with two classification algorithms, KNN and Bayes Net. Firstly, one data is selected from each cluster and the rest of labeled data are selected from the dataset randomly. These selected data are viewed as labeled data and the rest of the data in the given dataset as the unlabeled dataset. Secondly, some clusters are removed from the labeled dataset and the rest of labeled dataset are viewed as the labeled dataset to detect new clusters. And in the experiment, the value of k is set to be 5 and its meaning is the same as that in algorithm 1.

IRIS dataset: This subsection selects 8 subsets from IRIS dataset and the experimental results are shown in Fig. 1.

The clustering results of proposed method, SSDBSAN, KNN and BayesNet are shown in Fig. 1a. The experimental results show that the proposed semi-supervised clustering has a better result than SSDBSAN, especially in the case of giving few labeled data. Increasing the number of labeled data does not influence the accuracy of the proposed semi-supervised clustering algorithm

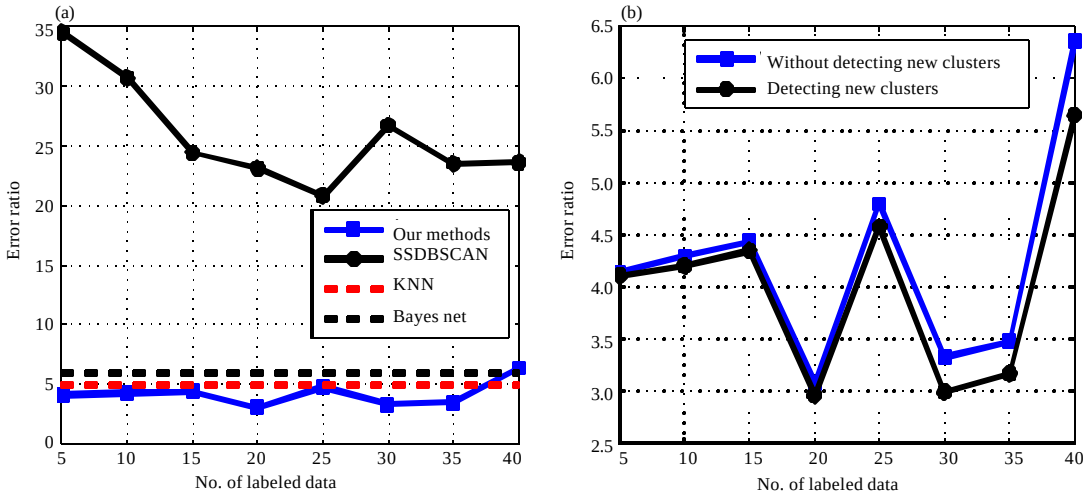

Fig. 1(a-b): Error ratio of clustering(%) on IRIS dataset, (a) Without detecting new clusters and (b) Detecting new clusters

but the accuracy of SSDBSCAN increases greatly with increasing the number of labeled data. The accuracy of KNN and BayesNet are close to that of the proposed clustering algorithm but in most cases, the proposed algorithm has a better result than the compared algorithms. In order to demonstrate that the proposed semi-supervised clustering algorithm has the ability of detecting new clusters, one cluster is removed from the original labeled dataset. The data in the cluster which has no labeled data will be assigned to other clusters by KNN, BayesNet and SSDBSAN. The three algorithms have lower accuracies in the modified dataset than the original dataset. Figure 1b plots only the error ratios of the proposed method in the modified dataset and original dataset. Figure 1b shows that the error ratios with detecting new clusters are close to that of without detecting new clusters, which means that the proposed method can detect new clusters on IRIS dataset.

Wine dataset: The experimental results with 8 labeled datasets are shown as Fig. 2. Figure 2a shows the relation between Clustering accuracies and the number of labeled data. The proposed semi-supervised clustering algorithm has much lower error accuracy than the compared algorithm SSDBSCAN. The most interesting result/phenomenon is that the proposed algorithm has lower error accuracy compared with KNN classification.

The data in the cluster which has no labeled data will be assigned to other clusters by KNN and SSDBSAN. They have lower accuracies in the modified dataset than the original dataset, so Fig. 2b plots only the error ratios of the proposed semi-supervised clustering algorithm under two dataset (modified dataset and original dataset). Although, removing one cluster from each labeled dataset, the error accuracy of the proposed method is influenced little, which shows that the proposed semi-supervised clustering algorithm has low error accuracy with few labeled data, even for the labeled data do not cover all clusters.

Page blocks dataset: This subsection selects 10 labeled datasets, the rates of them to the whole dataset are 1, 2, 3, 4, 5, 6, 7, 8, 9, 10%, respectively and Fig. 3 shows the experimental results. Figure 3a shows the error ratio of the proposed method is much lower than that of

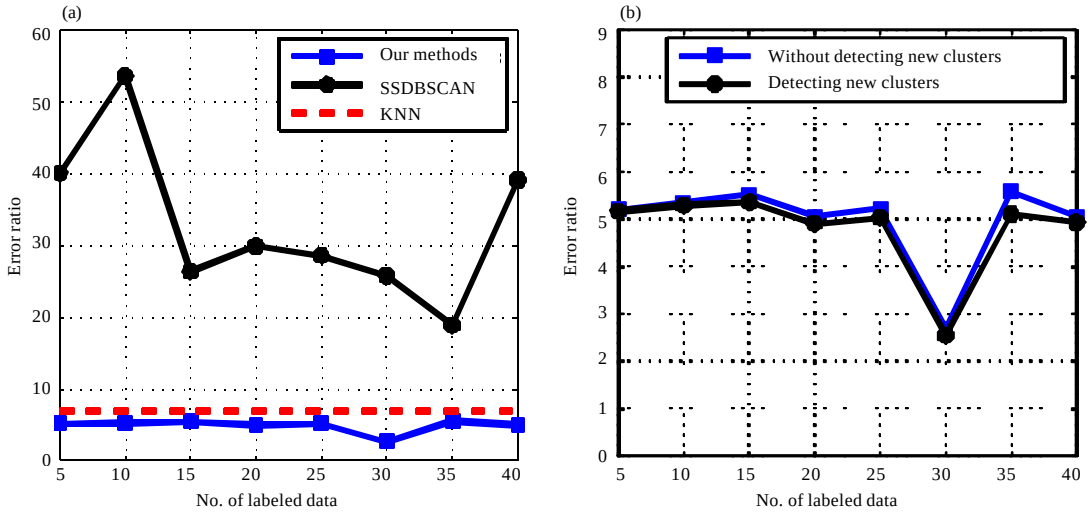

Fig. 2(a-b): Error ratio of clustering (%) on wine dataset, (a) Without detecting new clusters and (b) Detecting new clusters

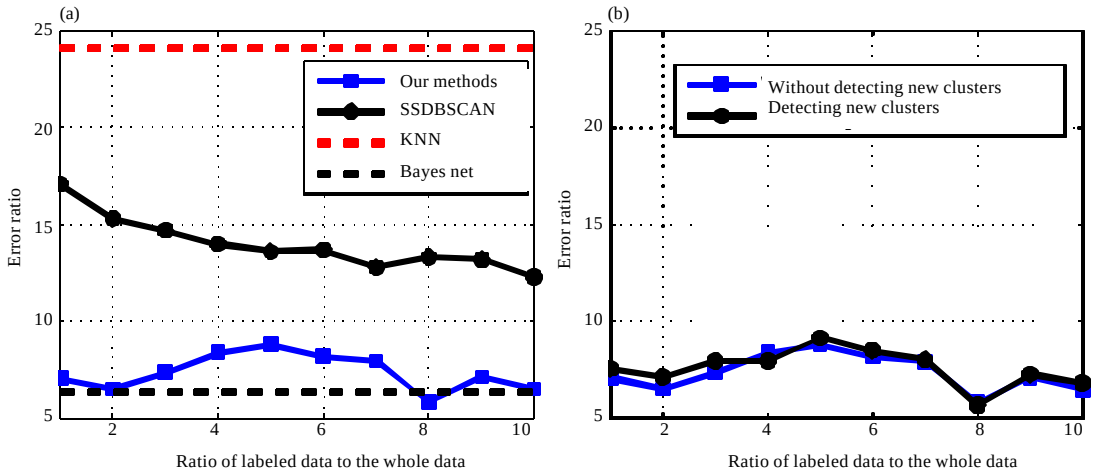

Fig. 3(a-b): Error ratio of clustering (%) on page blocks dataset, (a) Without detecting new clusters and (b) Detecting new clusters

SSDBSCAN and KNN and is close to that of BayesNet. Page Blocks dataset is an imbalanced dataset and in the experiment, using the Euclidean distance as similarity measure and utilizing the similarities between data, finding that many data and their k-nearest neighbors are not in the same cluster and which leads to the error ratio of KNN to be high. Although the error ratio of the proposed clustering algorithm is higher than BayesNet, the difference of error ratios between the proposed clustering algorithm and BayesNet is not significant. The modified dataset is generated by removing two clusters from each labeled dataset and the data in the cluster which has no labeled data will be assigned to other clusters by KNN, BayesNet and SSDBSAN. The three algorithms have lower accuracies in the modified dataset than the original dataset. Figure 3b only plots the error ratios of the proposed semi-supervised method in the modified dataset and original dataset.

Although, removing two clusters from each labeled dataset, the error accuracy of the proposed method is influenced little, which shows that the proposed semi-supervised clustering algorithm has low error accuracy with few labeled data, even for the labeled data can not cover all clusters.

DISCUSSION

The proposed semi-supervised clustering algorithm uses labeled data to expand labeled dataset by labeling k-nearest neighbors of labeled data in order to achieve better clustering results. Detecting new clusters is very important in many semi-supervised learning algorithms, even for online algorithms. In comparison with the algorithms in references (Ruiz *et al*., 2010; Leng *et al*., 2008; Dang *et al*., 2010), the proposed algorithm has ability of detecting new clusters on the dataset in which the differences of densities between clusters are not large, then the performance of the proposed algorithm is better than that of them. In addition, the accuracies of the proposed algorithm are higher than that of KNN on the three datasets. If the differences are large, then there are many data in low density clusters not to be labeled in the process of label expanding. The proposed method uses labeled core objects to guide the process of clustering but employing core objects to expand labeled dataset is not suitable in the multi-density datasets. How to use label information of labeled data adequately in the multi-density datasets will be investigated in the future work.

ACKNOWLEDGMENT

It is a project supported by the IBM 2010 X10 Innovation Awards Project, the Nature Science Foundation of Jiangxi Education Department of P.R. China (No. GJJ11609), the Fundamental Research Funds for the Central Universities (lzujbky-2012-212).

REFERENCES

Bache, K. and M. Lichman, 2013. UCI machine learning repository. University of California, School of Information and Computer Science, Irvine, CA., USA.

Baghshah, M.S. and S.B. Shouraki, 2009. Metric learning for semi-supervised clustering using pairwise constraints and the geometrical structure of data. Intell. Data Anal., 13: 887-899.

Baghshah, M.S. and S.B. Shouraki, 2010. Kernel-based metric learning for semi-supervised clustering. Neurocomputing, 73: 1352-1361.

Basu, B., A. Banerjee and R. Mooney, 2002. Semi-supervised clustering by seeding. Proceedings of the 19th International Conference on Machine Learning, July 8-12, 2002, Morgan Kaufmann Publishers Inc., San Francisco, CA., USA., pp: 27-34.

Basu, S., M. Bilenko and R.J. Mooney, 2004. A probabilistic framework for semi-supervised clustering. Proceedings of the 10th ACM International Conference Knowledge Discovery and Data Mining, August 22-25, 2004, Seattle, USA., pp: 59-68.

Bohm, C. and C. Plant, 2008. HISSCLU: A hierarchical density-based method for semi-supervised clustering. Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology, March 25-29, 2008, Nantes, France, pp: 440-451.

Dang, Y., Z. Xuan, L. Rong and M. Liu, 2010. A novel initialization method for semi-supervised clustering. Proceedings of the 4th International Conference on Knowledge Science, Engineering and Management, September 1-3, 2010, Belfast, Northern Ireland, UK., pp: 317-328.

Kulis, B., S. Basu, I. Dhillon and R. Mooney, 2009. Semi-Supervised graph clustering: A kernel approach. Mach. Learn. J., 74: 1-22.

Lelis, L. and J. Sander, 2009. Semi-supervised density-based clustering. Proceeding of the 9th IEEE International Conference on Data Mining, December 6-9, 2009, Miami, FL., USA., pp: 842-847.

Leng, M., X. Chen and L. Li., 2008. K-means clustering algorithm based on Semi-supervised learning. J. Comput. Inform. Syst., 5: 2007-2013.

Ruiz, C., M. Spiliopoulou and E. Menasalvas, 2010. Density-based semi-supervised clustering. Data Min. Knowl. Discov., 21: 345-370.

Wagstaff, K., C. Cardie, S. Rogers and S. Schrodl, 2001. Constrained K-means clustering with background knowledge. Proceedings of the 18th International Conference on Machine Learning, June 28-July 1, 2001, San Francisco, CA., USA., pp: 577-584.

Yin, X., S. Chen, E. Hu and D. Zhang, 2010. Semi-supervised clustering with metric learning: An adaptive kernel method. Pattern. Recogn., 43: 1320-1333.

Zhao, W., Q. He, H. Ma and Z. Shi, 2012. Effective semi-supervised document clustering via active learning with instance-level constraints. Knowl. Inform. Syst., 30: 569-587.