Large Scale Spectral Clustering Using Sparse Representation Based on Hubness

Xiucai Ye, Hongmin Li and Tetsuya Sakurai
Department of Computer Science
University of Tsukuba
Tsukuba, Japan

Zhi Liu
Department of Mathematical and Systems Engineering
Shizuoka University
Shizuoka, Japan

***Abstract*—Spectral clustering has been shown to be more effective than most of the traditional clustering algorithms. However, the heavy computational cost of spectral clustering limits its applicability to large-scale clustering problems. To perform spectral clustering on large datasets, in this paper, we propose an accelerated spectral clustering method based on sparse presentation where each data point is presented as sparse linear combinations of a part of representative data points. The hubs that appear frequently in the nearest neighbor lists of other data points are selected as the representative data points, by which a proper spectral embedding is constructed. Taking advantage of the topological property of hubs, the proposed method is able to achieve scalable and accurate clustering results. We evaluated the proposed method on both synthetic and real-world datasets to show its effectiveness in comparison to the existing related methods.**

***Index Terms*—large scale, spectral clustering, sparse presentation, hubness**

I. Introduction

In the current digital era, new technologies and their widespread usage have led to the emergence of massive data called big data. Big data is the term for data sets that are so voluminous and complex, which presents a significant challenge for data analysis, since traditional data-processing methods are inadequate to deal with them. Clustering is one of the most important issues when dealing with big data [1]. Large scale datasets usually consist of some groups (clusters). It is necessary to find the groups, retrieving important and meaningful information.

Spectral clustering has attracted increasing attention due to its superior performance on some challenging clustering tasks [2]. Because of the capacity of partitioning data with complexed structures, spectral clustering has been widely applied in many research fields, including image segmentation [3], [4], circuit layout [5], video retrieval [6] and bioinformatics [7]. However, when the number of data points becomes large, the applicability of spectral clustering is limited. The general spectral clustering method consists of two main steps: (1) constructing a similarity matrix; (2) calculating the eigen-decomposition of the corresponding Laplacian matrix. For a dataset with n data points, the two steps take computational complexities of $O(n^2)$ and $O(n^3)$ respectively, which is an unbearable burden for large-scale clustering problems.

Many accelerated spectral clustering methods have been proposed to overcome the scalability problem by using sampling techniques. Fowlkes et al. [8] apply the Nyström method to reduce the high complexity in the eigen-decomposition step. By randomly selecting a small subset of samples, a similarity sub-matrix is constructed based on these samples. The calculated eigenvectors based on the similarity sub-matrix are used to estimate an approximation of the eigenvectors of the original similarity matrix. Li et al. [9] further accelerate the Nyström approximation based spectral clustering by using the randomized low-rank matrix approximation algorithms.

Instead of reducing the complexity in the eigen-decomposition step of spectral clustering, several methods reduce the data size beforehand to construct the similarity matrix. The K-means based approximate spectral clustering (KASP) method [10] applies K-means with a large cluster number p to find p center points. The general spectral clustering algorithm is then performed on the p cluster centers, with each data point being assigned to the same cluster as its nearest center. A similar method has been proposed by Shinnou and Sasaki [11], which removes the data points close to the p centers, and the general spectral clustering is performed on the remaining data points plus the p centers. The removed data points are finally assigned to the cluster as their nearest centers.

Inspired by the recent progress on sparse coding [12], a family of accelerated spectral clustering methods based on sparse representation have been proposed [13], which have been shown to be more effective than the aforementioned accelerated spectral clustering methods. These methods select a part of representative data points as the landmarks and represent each data point as linear combinations of these landmarks, by which the computational complexity can be scaled linearly with the problem size. In [13], the authors apply two methods to select the landmarks. One is randomly selecting some data points as the landmarks, the other one is using K-means clustering and finding the cluster centers as the landmarks. Although the methods are effective and easy to implement, a lot of information on the topological properties of the data points is lost when selecting the landmarks. Rafailidis et al. [14] propose an improved method to select the landmarks by considering the topological properties of the data points in the affinity graph. The landmarks are selected by weighted pagerank algorithm, by which the weight values of data points are ranked based on the incoming and outgoing

links. However, this method is no easy to implement and with limited applicability.

In this paper, we propose an effective method to accelerate spectral clustering based on sparse representation, which selects the hubs as the landmarks. Hubness, the emergence of hubs that appear frequently in the nearest neighbor lists of other data points, is an important topological property of data, especially for high-dimensional data [15]. The hubs have close relationship with the data points, since they are the k nearest neighbors of most of the data points. Many methods based on hubness have been reported to be effective for redundancy reduction and clustering accuracy improvement [16]. The proposed method takes advantage of the topological property of hubs to find a good sparse representation of data points, which is easy to implement and is able to achieve scalable and accurate clustering results. Extensive experiments on both synthetic and real-world datasets show the effectiveness of the proposed method in comparison to the existing related methods.

The remainder of this paper is organized as follows. We present the preliminaries in Section II. The proposed method is introduced in Section III. The experimental results are presented in Section IV. Finally, we provide some concluding remarks in Section V.

II. Preliminaries

A. Spectral Clustering

Given a set of n data points $X = \{x_1, x_2, ..., x_n\}$ in $\mathbb{R}^d$ and some measure of similarity s_{ji} between data points x_i and x_j, the goal of spectral clustering is to partition the data points into c clusters based on the spectrum of the similarity matrix. The widely used method to calculate s_{ji} is the Gaussian kernel function

$$\Phi(x_i, x_j) = \exp(-\|x_i - x_j\|^2/2\sigma^2), \tag{1}$$

where $\|x_i - x_j\|$ is the Euclidean distance between data points x_i and x_j ($i \neq j$), and σ is the kernel parameter.

The general spectral clustering algorithm is summarized as follows [17], [18] .

(1) Construct a similarity matrix S, where $S = (s_{ji})$.

(2) Compute the normalized Laplacian matrix L as $L = I - D^{-1/2}SD^{-1/2}$, where D is the $n \times n$ diagonal matrix with $d_i = \sum_{j=1}^{n} s_{ij}$ on the diagonal.

(3) Compute the c smallest eigenvectors of L, and form the matrix $Y = (y_1, y_2, ..., y_c)_{n\times c}$ using these eigenvectors as its columns.

(4) Form the matrix $B = (b_{ij})_{n\times c}$ from Y by normalizing the rows to norm 1, such as $b_{ij} = y_{ij}/(\sum_p y_{ip}^2)^{1/2}$.

(5) Let $b_i \in \mathbb{R}^c$ be a vector corresponding the i^{th} row of B, and cluster these points with the K-means method.

(6) Assign each data point x_i to a given cluster if b_i is assigned to this cluster.

B. Sparse Representation

The method of sparse representation tries to represent each data point by sparse linear combinations of a set of representative data points (i.e., landmarks), as

$$X \approx LH, \tag{2}$$

where $L = [l_1, l_2, ...l_p]$ is the set of landmarks selected from $x_1, x_2, ..., x_n$, $H = [h_1, h_2, ...h_n]$ is the p-dimensional representation of the original data points with respect to the p landmarks and H has sparse constraint, more specifically, on each column.

For spectral clustering methods using sparse representation, the objective is to design the similarity matrix S as [13]

$$S \approx \tilde{H}^T\tilde{H}, \tag{3}$$

where $\tilde{H} = \tilde{D}^{-1/2}H$, $\tilde{D}$ is the $p\times p$ diagonal matrix with $\tilde{d}_i = \sum_{j=1}^{n} h_{ij}$ on the diagonal. H is expressed as the similarities of the n data points to the p landmarks. Thus, the selection of the landmarks is significant to the calculation of similarity matrix and crucial to the clustering result.

III. The Proposed Method

In this section, we introduce the proposed accelerated spectral clustering method based on sparse representation. The proposed method first selects the data points with large hubness score to represent each data point and construct the approximated similarity matrix, then generates clusters based on the eigen-decomposition of the approximated similarity matrix.

A. Sparse Representation of Similarity Matrix

Hubness is an important topological property of data. The hubness score of a data point is measured as the number of times it occurs among the k nearest neighbors of other data points, according to some distance measure. We use the Euclidean distance in this paper. The hubness score of data point x_i is calculated as

$$\Gamma(x_i) = \sum_{j=1}^{n} p_{ij}, \tag{4}$$

where

$$p_{ij} = \begin{cases} 1, & x_i \text{ is among the } k \text{ nearest neighbors of } x_j, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Directly calculating the hubness score of data points involves the calculation of the k nearest neighbor graph of the whole dataset, which is computationally expensive. To reduce the computational complexity, we first divide the data points into some parts, and then calculate the local hubness score of data points in each part. The most efficient way to divide the data points into some parts is random partition. Besides random partition, we can also use the K-means method to cluster the data points into some clusters/parts. For efficiency consideration, we will focus on the random partition although

the comparison between random partition and the K-means based partition is presented in the empirical study.

To form the matrix L in equation (2), we need to select p landmarks $l_1, l_2, ...l_p$. Suppose that all the data points are divided into m parts P_1, P_2,...,P_m. In part P_i ($i = 1, ..., m$), n_i data points with the largest hubness score are selected as the landmarks.

$$n_i = \frac{|P_i|}{n}p, \tag{6}$$

and if necessary some adjustments will be made to satisfy $\sum_{i=1}^{m} n_i = p$. After obtaining the matrix L, we can compute the representation matrix H by solving (2). However, the solution of (2) is time consuming for large-scale dataset. To reduce the complexity, we compute H according to the Nadaraya-Watson kernel regression [19].

We calculate the approximation $\tilde{x_i}$ for x_i as

$$\tilde{x_i} = \sum_{j=1}^{n} h_{ij} l_j. \tag{7}$$

As the assumption in [13], we know that h_{ij} should be larger if x_i is closer to the landmark l_j. We can construct the sparse representation matrix H by setting $h_{ij} = 0$ if l_j is not among the $r(\leq p)$ nearest landmarks of x_i. Denote N_i as the set of landmarks that are among the $r(\leq p)$ nearest landmarks of x_i. For $i = 1, ..., n$ and $l_j \in N_i$, the element h_{ij} is calculated as

$$h_{ij} = \frac{\Phi(x_i, l_j)}{\sum_{l'_j \in N_i} \Phi(x_i, l_{j'})}, \tag{8}$$

where $\Phi(\cdot)$ is the Gaussian kernel function as defined in equation (1).

After constructing the sparse representation matrix H, the approximated similarity matrix S can be calculated based on H as shown in equation (3).

B. Generation of Clusters

Inspired by [13], we cluster the data points based on the eigen-decomposition of the approximated similarity matrix S by an efficient method. Let the Singular Value Decomposition (SVD) of $\tilde{H}$ be calculated as

$$\tilde{H} = U\Sigma V^T, \tag{9}$$

where $\Sigma = diag(\sigma_1, \sigma_2, ..., \sigma_p)$ and $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_p \geq 0$ are the singular of $\tilde{H}$, $U = [u_1, u_2, ..., u_p] \in \mathbb{R}^{p \times p}$ and the column vectors $u_1, u_2, ..., u_p$ are called left singular vectors, $V = [v_1, v_2, ..., v_p] \in \mathbb{R}^{n \times p}$ and the column vectors $v_1, v_2, ..., v_p$ are called right singular vectors. It is easy to verify that the column vectors of U are the eigenvectors of $\tilde{H}\tilde{H}^T$, and the column vectors of V are the eigenvectors of $\tilde{H}^T\tilde{H}$ which is exactly the eigenvectors of the approximated similarity matrix S, as shown in equation (3).

Note that the size of matrix $\tilde{H}\tilde{H}^T$ is $p \times p$. We first calculate the eigenvectors of $\tilde{H}\tilde{H}^T$ to obtain U with complexity $O(p^3)$. Then, we can obtain the eigenvectors of $S = \tilde{H}^T\tilde{H}$ by calculating V according to equation (9) as

$$V^T = \Sigma^{-1}U^T\tilde{H}, \tag{10}$$

Thus, the total time to obtain the eigen-decomposition of S is $O(p^3 + p^2n)$, which is a significant reduction from $O(n^3)$ when $p \ll n$. Let each row of V be a data point. Finally, K-means is applied to these data points to find the c clusters. The proposed method is summarized in Algorithm 1.

Algorithm 1 The proposed method

Input:

Data matrix $X = \{x_1, x_2, ..., x_n\}$; number of clusters c; number of nearest neighbors k; number of partitions m; number of landmarks p; number of nearest landmarks r;

Output:

c clusters;

1: Divide the data points into m parts using K means or random partition;
2: Select n_i landmarks that have the largest hubness score in part P_i ($i = 1, ..., m$);
3: Construct a sparse representation matrix $H \in \mathbb{R}^{p \times n}$ between data points and the p landmarks according to equation (8);
4: Obtain $U = [u_1, u_2, ..., u_c]$ by computing the first c eigenvectors of $\tilde{H}\tilde{H}^T$ where $\tilde{H} = \tilde{D}^{-1/2}H$;
5: Compute $V = [v_1, v_2, ..., v_c]$ according to equation (10) ;
6: Generate c clusters by applying K-means in V with each row being a data point.

C. Computational Complexity Analysis

We analyze the computational complexity of the proposed method in four main steps.

(1) If we divide the data points into m parts by K-means, the time complexity is $O(tmn)$, where t is the number of iterations in K-means. However, if we sue random partition, the time complexity can be ignored comparing to the follow-up steps.

(2) To select the landmarks from each part, we need to measure the local hubness score of the data points in each part, which involves the computation of local k nearest neighbors. In this step, the time complexity is $O(q^2 + kq)$, where $q = max\{|P_i|\}$, $i = 1, ..., m$.

(3) To compute the similarity matrix S, we need to construct the sparse representation matrix H. For each data point, it is necessary to find the r nearest landmarks from the p landmarks. The time complexity in this step is $O(pn)$.

(4) The time complexity of generating the left singular vectors of is $O(p^3)$. The time complexity to derive the right singular vectors is $O(p^2n)$,as shown in equation (10). Thus, the total time complexity to obtain the eigenvectors of S is $O(p^3 + p^2n)$.

IV. Experimental Results

In this section, we conduct several experiments based on both synthetic and real-world datasets to evaluate the performance of the proposed method. We adopt two widely used evaluation metrics, i.e., Normalized Mutual Information (NMI) and Accuracy (ACC), to evaluate the clustering results. The

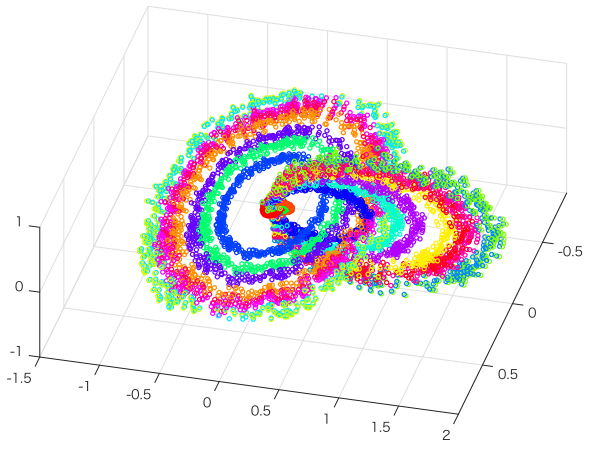


Fig. 1. Chainlinks with 20 clusters.

detailed methods to calculate NMI and ACC can be referred to [20]. A larger value of NMI/ACC denotes a better clustering result. We also record the clustering time of the proposed method.

We compare the proposed method with other existing related methods, including

Nystrom: There are several variants available for the Nyström approximation based spectral clustering. We choose the method that using Matlab implementation with orthogonalization, as that considered in [21].

LSC-R: Sparse representation based spectral clustering by randomly selecting the landmarks, without considering the topological properties of data points [13].

LSC-K: Sparse representation based spectral clustering by using K-means clustering to find the cluster centers as the landmarks [13].

To show the effectiveness of the accelerated spectral clustering methods, we also report the results of the original spectral clustering. For the proposed accelerated spectral clustering method using sparse representation based on hubness, we implement two versions as follows.

LSC-RH: Short for Landmark-based Spectral Clustering using random partition to find the hubs as the landmarks.

LSC-KH: Short for Landmark-based Spectral Clustering using K-means partition to find the hubs as the landmarks.

There are four parameters in our method: the number of nearest neighbors k; the number of partitions m; the number of landmarks p; the number of nearest landmarks r. For the sake of convenience, we set $k = r$ and $p = 2m$. As a result, the number of parameters in our method is reduced to two. Throughout the experiments, k is ranged over $\{3, 4, 5, 6, 7, 8, 9, 10\}$ and p is ranged over $\{100, 200, 300, 400, 500, 600, 700, 800, 900\}$. The best results of all the methods by using different parameters are reported. We repeat each experiment 20 times and report the mean performance.

TABLE I
PROPERTIES OF DATASETS

Dataset	# of samples	# of Features	# of Clusters
Chainlinks	10000	3	20
USPS	9298	256	10
TDT2	9394	36772	30
PenDigits	10992	16	10
SenVehicle	78823	100	3

A. Clustering Results on Synthetic Data

We use a 3D synthetic dataset called Chainlinks to evaluate the performance. Chainlinks contains 10000 data points taken from 20 clusters. The data points in each cluster are distributed in a circle shape and denoted by different colors, as shown in Fig. 1. The property of Chainlinks is summarized in Table I.

We compare the performance of different methods along with the original spectral clustering method on Chainlinks, and summarize the clustering time, clustering results in terms of NMI and ACC in Tables II, III and IV, respectively. The proposed LSC-RH and LSC-KH methods improve the clustering results of the synthetic data, while do not cause a large increase in the computational time compared to other accelerated methods. All the accelerated spectral clustering methods reduce the clustering time of the original spectral clustering. The clustering time of LSC-RH which using random partition is very close to the compared methods. To further examine the behaviors of these methods on the synthetic data, we evaluate the clustering results by varying the number of landmarks. The clustering results in terms of NMI and ACC are shown in Fig. 2. As we can see, LSC-RH and LSC-KH achieve better clustering results as the number of landmarks increases.

B. Clustering results on real-world data

We use four public real-world datasets to evaluate the performance. The datasets are USPS [22], TDT2 [23], PenDigits [24] and SenVehicle [25], which are wildly used to evaluate the performance of large scale data clustering. The properties of the datasets are summarized in Table I and briefly introduced as follows.

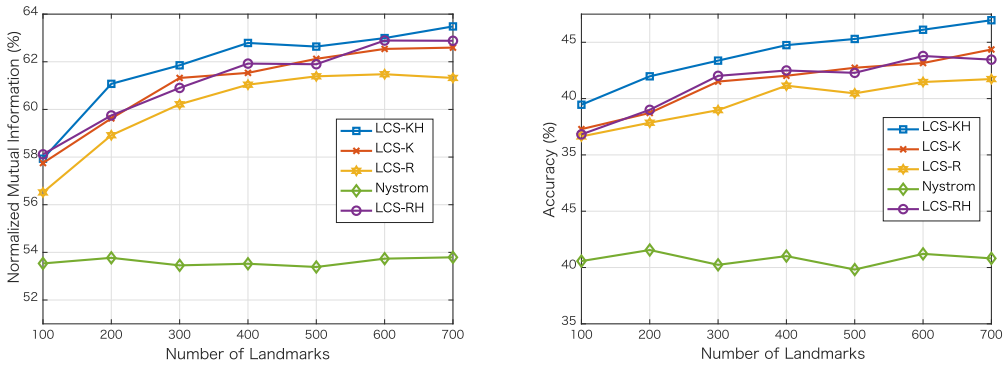
USPS: An image dataset for handwritten text recognition. It is taken from 10 clusters with each image being represented as a 256-dimensional vector.

TDT2: A subset of the original TDT2 corpus from [23]. It consists of document data from 6 media sets including APW, NYT, VOA, RPI, CNN and ABC. The largest 30 categories are kept in this subset.

PenDigits: A dataset of handwritten digits with 250 samples collected from 44 writers. It uses the sampled coordinate information of each digit.

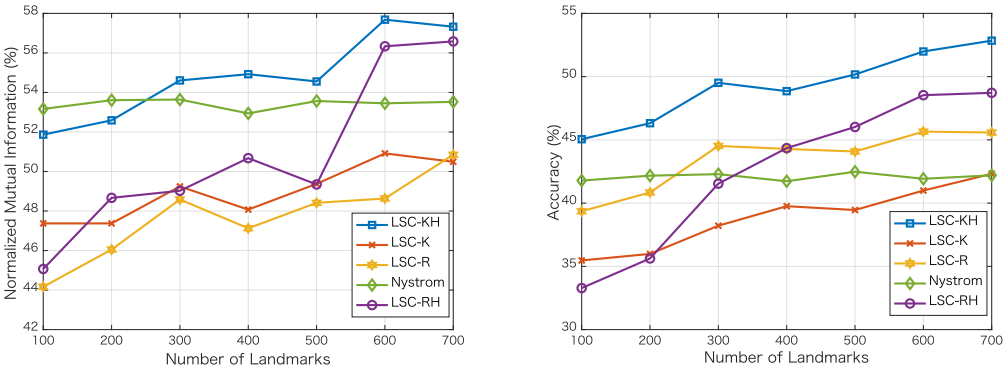
SenVehicle: A dataset initially built to classify the moving vehicles in a distributed wireless sensor network.

The performance on the four real-world datasets using different methods along with the original spectral clustering method are reported in Tables II, III and IV, respectively. We can see that for the clustering results in terms of NMI



(a) NMI (b) ACC

Fig. 2. Clustering results by varying the number of landmarks on Chainlinks.



(a) NMI (b) ACC

Fig. 3. Clustering results by varying the number of landmarks on TDT2.

TABLE II
CLUSTERING TIME OF DIFFERENT METHODS (SECOND)

Dataset	Chainlinks	USPS	TDT2	PenDigits	SenVehicle
Original	52.86	53.27	60.07	61.23	4198
Nystrom	1.84	1.89	5.32	9.01	17.25
LSC-R	1.87	1.58	4.51	3.37	16.73
LSC-K	2.28	2.36	20.36	21.83	65.63
LSC-RH	2.19	2.28	19.24	22.87	62.48
LSC-KH	3.15	3.23	34.52	36.68	86.45

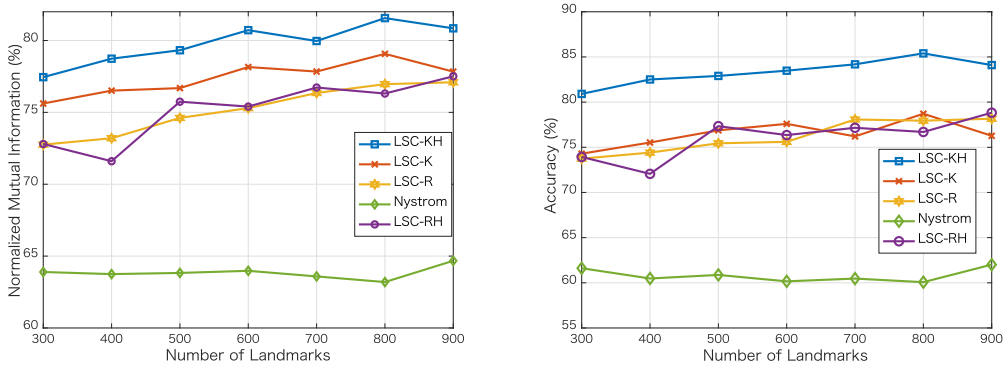
TABLE III
NMI OF DIFFERENT METHODS

Dataset	Chainlinks	USPS	TDT2	PenDigits	SenVehicle
Original	43.92	80.20	58.12	56.24	28.87
Nystrom	33.79	78.19	53.64	63.32	30.50
LSC-R	41.48	74.07	50.93	77.01	30.19
LSC-K	42.59	77.19	50.90	78.51	30.01
LSC-RH	42.89	79.65	56.62	77.51	29.67
LSC-KH	43.48	78.71	57.69	81.55	30.96

TABLE IV
ACC OF DIFFERENT METHODS

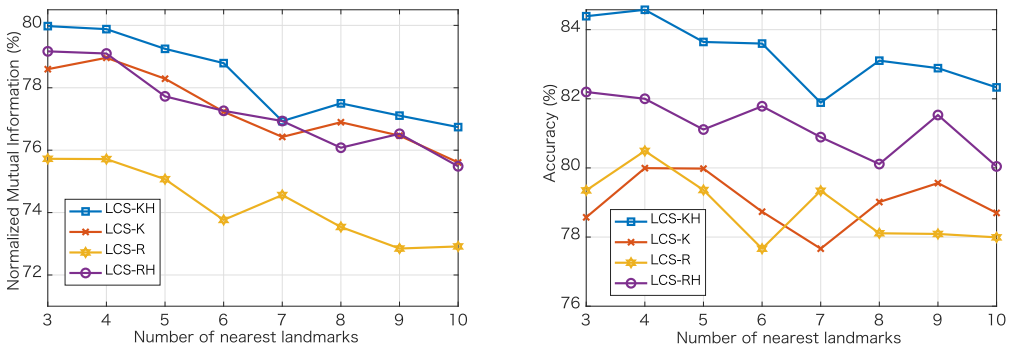
Dataset	Chainlinks	USPS	TDT2	PenDigits	SenVehicle
Original	43.12	69.82	47.45	60.38	63.67
Nystrom	36.59	73.74	42.49	62.03	67.09
LSC-R	42.69	67.85	45.65	78.22	65.99
LSC-K	43.74	68.74	42.31	76.43	66.68
LSC-RH	43.51	69.07	48.60	78.43	65.77
LSC-KH	44.78	74.01	52.79	85.4	67.79

and ACC, the proposed LSC-RH and LSC-KH methods outperform other accelerated methods on the four real-world datasets. Especially on the datasets of TDT2 and PenDigits, the improvement by the LSC-KH method is significant. That is because the LSC-KH method uses K-means for data partition to find the hubs, which can find more important landmarks than the random partition in LSC-RH. However, as reported in Table II, LSC-KH consumes more clustering time than LSC-RH due to the additional time to perform K-means. On the four datasets, all the accelerated spectral clustering methods reduce the clustering time of the original spectral clustering method, especially for the dataset SenVehicle that has the



(a) NMI (b) ACC

Fig. 4. Clustering results by varying the number of landmarks on PenDigits.



(a) NMI (b) ACC

Fig. 5. Clustering results by varying the number of nearest landmarks on PenDigits.

largest size, the time reduction is the most significant. We also note that the clustering time of the LSC-RH method is very close to other compared accelerated methods, while it obtains better clustering accuracy than other compared accelerated methods.

The clustering results in terms of NMI and ACC by varying the number of landmarks p on the datasets of TDT2 and PenDigits are shown in Figs. 3 and 4. The proposed LSC-RH and LSC-KH methods achieve better clustering results. LSC-KH performs better than other compared methods as the number of landmarks varies in most of the cases.

As shown in equation (8), we construct the sparse representation matrix H by measuring the similarity of a single data point to its r nearest landmarks. Thus, we also evaluate the clustering results by varying the parameter r, i.e., the number of nearest landmarks. Fig. 5 shows how the clustering results in terms of NMI and ACC vary with the number of nearest landmarks on the dataset PenDigits. It can be seen that the proposed LSC-RH and LSC-KH methods achieve good performance with the r varying from 3 to 10.

V. Conclusion

In this paper, we proposed an accelerated spectral clustering method based on sparse presentation by selecting the hubs as the landmarks. Taking advantage of the topological property of hubs, the proposed method found a good sparse representation for the original data points, by representing the original data points as the linear combinations of the landmarks. The spectral embedding of the data could be efficiently computed to scale linearly with the problem size. In the experiments, we implemented two versions of the proposed method, i.e., LSC-RH and LSC-KH which are based on random partition and K-means partition, respectively. Experimental results on both synthetic and real-world datasets demonstrated the effectiveness of LSC-RH and LSC-KH in comparision to the existing related methods.

References

[1] O. Kurasova, V. Marcinkevicius, V. Medvedev, A. Rapecka, and P. Stefanovic, "Strategies for big data clustering," in *Proceeding of IEEE 26th International Conference on Tools with Artificial Intelligence*, 2014, pp. 740–747.

[2] X. Ye and T. Sakurai, "Spectral clustering using robust similarity measure based on closeness of shared nearest neighbors," in *Proceeding of International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1–8.

[3] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[4] J. Malik and S. Belongie, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[5] C. J. Alpert and A. B. Kahng, "Multiway partitioning via geometric embeddings, orderings and dynamic programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 4, no. 11, pp. 1342–1358, 1995.

[6] W. Hu, D. Xie, Z. Fu, W. Zeng, and S. Maybank, "Semantic-based surveillance video retrieval," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1168–1181, 2007.

[7] Z. Yu, L. Li, J. You, and G. Han, "Sc3: Triple spectral clustering based consensus clustering framework for class discovery from cancer gene expression profiles," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 6, pp. 175–1765, 2012.

[8] C. Fowlkes, S. Belongie, F. Chung, and J. Malik, "Spectral grouping using the nyström method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 214–225, 2004.

[9] M. Li, J. T. Kwok, and B. L. Lu, "Making large-scale nystrm approximation possible," in *Proceeding of 27th International Conference on Machine Learning*, 2010, pp. 631–638.

[10] D. Yan, L. Huang, and M. I. Jordan, "Fast approximate spectral clustering," in *Proceeding of 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 907–916.

[11] H. Shinnou and M. Sasaki, "Spectral clustering for a large data set by reducing the similarity matrix size," in *Proceeding of International Conference on Language Resources and Evaluation*, 2008, pp. 201–204.

[12] D. L. Donoho and M. Elad, "Optimally sparse representation in general (non-orthogonal) dictionaries via l1 minimization," in *Proceeding of the National Academy of Sciences*, 2003, pp. 2197–2202.

[13] X. Chen and D. Cai, "Large scale spectral clustering with landmark-based representation," in *Proceeding of AAAI Conference on Artificial Intelligence*, 2011, pp. 313–318.

[14] D. Rafailidis, E. Constantinou, and Y. Manolopoulos, "Landmark selection for spectral clustering based on weighted pagerank," *Future Generation Computer Systems*, vol. 68, pp. 465–472, 2017.

[15] N. Tomasev, M. Radovanovic, D. Mladenic, and M. Ivanovic, "The role of hubness in clustering high-dimensional data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 739–751, 2014.

[16] Z. He, "Hub selection for hub based clustering algorithms," in *Proceeding of International Conference on Fuzzy Systems and Knowledge Discovery*, 2014.

[17] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[18] X. Ye and T. Sakurai, "Robust similarity measure for spectral clustering based on shared neighbors," *ETRI Journal*, vol. 38, no. 3, pp. 540–550, 2016.

[19] W. Hrdle, "Applied nonparametric regression," *Cambridge, U.K.: Cambridge Univ. Press*, 1992.

[20] X. Ye and T. Sakurai, "Feature selection via embedded learning based on tangent space alignment for microarray data," *Journal of Computing Science and Engineering*, vol. 11, no. 4, pp. 121–129, 2017.

[21] W. Chen, Y. Song, H. Bai, C. Lin, and E. Chang, "Parallel spectral clustering in distributed systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 568–586, 2011.

[22] J. Hull, "A database for handwritten text recognition research," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 550–554, 1994.

[23] D. Cai, X. He, and J. Han, "Document clustering using locality preserving indexing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 12, pp. 1624–1637, 2005.

[24] F. Alimoglu and E. Alpaydin, "Methods of combining multiple classifiers based on different representations for pen-based handwriting recognition," in *Proceeding of Turkish Artificial Intelligence and Artificial Neural Networks Symposium*, 1996.

[25] M. Duarte and Y. H. Hu, "Vehicle classification in distributed sensor networks," *Journal of Parallel and Distributed Computing*, vol. 64, no. 7, pp. 826–838, 2004.