IMPROVING SPECTRAL CLUSTERING USING SPECTRUM-PRESERVING NODE REDUCTION

Yongyu Wang
University of Southern California, Los Angeles, CA 90089, USA

ABSTRACT
Spectral clustering is one of the most popular clustering methods. However, the high computational cost due to the involved eigen-decomposition procedure can immediately hinder its applications in large-scale tasks. In this paper, we use spectrum-preserving node reduction to accelerate eigen-decomposition and generate concise representations of data sets. Specifically, we create a small number of pseudo-nodes based on spectral similarity. Then, standard spectral clustering algorithm is performed on the smaller node set. Finally, each data point in the original data set is assigned to the cluster as its representative pseudo-node. The proposed framework run in nearly-linear time. Meanwhile, the clustering accuracy can be significantly improved by mining concise representations. The experimental results show dramatically improved clustering performance when compared with state-of-the-art methods.

Index Terms— Spectral Clustering, Node Reduction

1. INTRODUCTION
Clustering is one of the most fundamental machine learning problems [1]. It aims to assign data samples in a data set into different clusters in such a way that samples in the same cluster are more similar compared to those in different clusters.

In the past decades, many clustering techniques have been proposed. Among them, spectral clustering has drawn considerable attention and has been widely applied to computer vision, image processing and speech processing [2, 3]. Although spectral clustering has superior performance, the involved eigen-decomposition procedure has a time complexity of $O(N^3)$, where $N$ is the number of samples in the data set. The high computational cost can easily become an obstacle in many large-scale applications [4, 5]. To solve this problem, considerable effort has been devoted in research related to approximate spectral clustering. [6] proposed a k-means based hierarchical spectral clustering framework (KASP); Inspired by sparse coding theory, [5] proposed a landmark-based representation method (LSC); [7] leveraged the Nyström method to approximate the affinity matrix. However, none of these methods can preserve the spectrum of the original graphs, thus result in a significant degradation of clustering accuracy.

For example, for Covtype data set, the clustering accuracy drops more than 26% when using the LSC method, more than 21% when using the KASP framework and more than 15% when using the Nyström method.

[8] introduced a graph coarsening method for dividing the graph topology into random number of partitions. However, [8] mixed up and garbled spectral clustering and graph partitioning. All the data used for graph partitioning in [8] are without cluster membership (labels), so their claim that it works for clustering is wrong. Clustering aims to group similar samples together, however, the graphs used in [8] are pure topologies, rather than data graphs. Their nodes are not associating with data samples. So, there is no "similarity" between their nodes, how comes clustering? [8] is not applicable to spectral clustering due to the following reasons: 1) spectral clustering aims to discover clusters for all the samples in the data set. However, original samples are missing in the coarsened graph. 2) the number of clusters in spectral clustering is fixed and meaningful (such as handwritten digits data sets have 10 clusters representing the digits 0-9). So clustering of the data is meaningless if the number of clusters is a random number. 3) To measure the clustering performance, direct measurements such as clustering accuracy should be used. In contrast, there is no direct way to evaluate the random graph partitioning. Merits such as quotient-style metrics, edge-cut metric and expansion or conductance are all needed to be considered. Only using the normalized cut as in [8] cannot demonstrate its effectiveness.

In this paper, we improve spectral clustering via spectrum-preserving node reduction. Our contributions are as follows: (1) To the best of our knowledge, this is the first attempt for employing spectrum-preserving node reduction in the general approximate spectral clustering framework [6] to improve the KASP method. (2) The proposed method can significantly improve the clustering accuracy: our method achieves 17% and 18% accuracy gain on USPS and MNIST data sets over the standard spectral clustering method, respectively. (3) Our method fundamentally addressed the computational challenge of eigen-decomposition procedure in standard spectral clustering algorithm. Compared to the existing approximation methods, our method is the only one that enables us to accelerate spectral clustering for very large data set without loss of clustering accuracy.
2. PRELIMINARIES

k-means is the most fundamental clustering method [9]. It discovers $k$ clusters by minimizing the following objective function:

$$ F = \sum_{i=1}^{n} \sum_{j=1}^{k} \eta_{ij} \| x_i - \mu_j \|_2, $$

(1)

where:

$$ \eta_{ij} = \begin{cases} 1 & \text{if } x_i \in C_j \\ 0 & \text{otherwise} \end{cases}, $$

(2)

and $\mu_j$ is the centroid of the cluster $C_j$.

However, it often fails to handle complicated geometric shapes, such as non-convex shapes [10]. In contrast, spectral clustering is good at detecting non-convex and linearly non-separable patterns [11]. Spectral clustering includes the following three steps: 1) construct a Laplacian matrix $L_G$ according to the similarities between data points; 2) embed nodes into $k$-dimensional space using the first $k$ nontrivial eigenvectors of $L_G$; 3) apply k-means to partition the embedded data points into $k$ clusters. As shown in Figure 1, in the two moons data set, there are two slightly entangled non-convex shapes, where each cluster corresponds to a moon. In the two circles data set, the samples are arranged in two concentric circles, where each cluster corresponds to a circle. k-means gives incorrect clustering results for both data sets while spectral clustering performs an ideal clustering. Because k-means only cares about the distance while spectral clustering cares about the connectivity between nodes. The connectivity information is embedded in the spectrum of the underlying graph so that it is critical to preserve the spectrum when manipulating graphs [12].

3. METHODS

3.1. Algorithmic Framework

We first construct a standard k-NN graph $G$. Then, based on the node proximity measure proposed in [13], the spectral similarity of two nodes $p$ and $q$ can be calculated as:

$$ c_{p,q} = \frac{\| (X_p, X_q) \|_2^2}{(X_p, X_p)(X_q, X_q)}, \quad (X_p, X_q) := \sum_{k=1}^{K} \left( x^{(k)}_p, x^{(k)}_q \right), $$

where $X := (x^{(1)}, \ldots, x^{(K)})$ is obtained by applying Gauss-Seidel relaxation to solve $L_G x^{(i)} = 0$ for $i = 1, \ldots, K$, starting with $K$ random vectors. We aggregate the nodes with high spectral similarity to reduce the graph size while preserving the spectrum of the original graph. During this process, The fine-to-coarse graph mapping operators $H_i^{+1}$ and the coarse-to-fine mapping operator $H_i^{-1}$ can be constructed [8]. By aggregating nodes based on spectral similarity, the reduced graph can best represent the original data set in the sense of minimizing the structural distortion [12], as shown in Figure 2.

![Visualization of node reduction results of USPS data set](image)

To retrieve the cluster membership of the samples in the original data set, we build a correspondence table using the mapping operators to associate original samples with aggregated nodes. If the desired graph size is not reached, the graph can be further reduced in a multi-level way [8]: given the graph Laplacian $L_G$ and the mapping operators from the finest level $1$ to the coarsest level $r$, $L_R = H_R^{-1} L_G H_R^T$, where $H_R^G = H_1^G H_2^G \cdots H_{r-1}^G$ and $H_R^G = H_1^G H_2^G \cdots H_r^G$.

Then standard spectral clustering is applied on the aggregated nodes to obtain $k$ clusters of the reduced data set. Finally, we determine the cluster membership of each node by

![Performance of k-means and spectral clustering](image)
checking the cluster membership of its corresponding aggregated node based on the correspondence table. The complete algorithm flow has been shown in Algorithm 1.

### Algorithm 1 spectrum-preserving node reduction-based high-performance spectral clustering framework

**Input:** A data set \( D \) with \( N \) samples \( x_1, ..., x_N \in \mathbb{R}^d \), number of clusters \( k \).

**Output:** Clusters \( C_1, ..., C_k \).

1. Construct a k-nearest neighbor (kNN) graph \( G \) from the input data.
2. Compute the adjacency matrix \( A_G \) and diagonal matrix \( D_G \) of graph \( G \).
3. Compute the Laplacian matrix \( L_G = D_G - A_G \).
4. Perform spectrum-preserving node reduction to obtain the reduced graph Laplacian \( L_r \).
5. Build a correspondence table to associate each node with its corresponding aggregated node.
6. Compute the eigenvectors \( u_1, ..., u_k \) that correspond to the bottom \( k \) nonzero eigenvalues of \( L_r \).
7. Construct \( U \in \mathbb{R}^{V \times k} \), with the \( k \) eigenvectors stored as column vectors.
8. Perform k-means algorithm to partition the rows of \( U \) into \( k \) clusters.
9. Retrieve the cluster membership of each \( x_i \) by assigning it to the cluster as its corresponding aggregated node and return the result.

### 3.2. Algorithm Complexity

The computational complexity of spectrum-preserving node reduction is \( O(|E_G| \log(|V|)) \), where \(|E_G|\) is the number of edges in the original graph and \(|V|\) is the number of nodes. The computational complexity of reduced standard spectral clustering is \( O(P^3) \) where \( P \) is the number of aggregated nodes. The complexity of cluster membership retrieving is \( O(N) \), where \( N \) is the number of samples in the original data set.

### 4. EXPERIMENT

Experiments are performed using MATLAB R2020b running on a Laptop. The reported results are averaged over 10 runs.

#### 4.1. Experiment Setup

One mid-sized data set (USPS), one large data set (MNIST) and one very large data set (Covtype) are used in our experiments. They can be downloaded from the UCI machine learning repository[^1] and LibSVM Data[^2]: USPS includes 9,298 images of USPS hand written digits with 256 attributes; MNIST is a data set from Yann LeCun’s website[^3], which includes 70,000 images with each of them represented by 784 attributes; Covtype includes 581,012 instances for predicting forest cover type from cartographic variables and each instance with 54 attributes is from one of seven classes.

We compare the proposed method against both the baseline and the state-of-the-art fast spectral clustering methods including: (1) the standard spectral clustering algorithm [9], (2) the Nyström method [7], (3) the Landmark-based spectral clustering (LSC) method that uses random sampling for landmark selection [5], and (4) the KASP method using k-means for centroids selection [6]. For fair comparison, we use the same parameter setting in [5] for compared algorithms: the number of sampled points in Nyström method (or the number of landmarks in LSC, or the number of centroids in KASP) is set to 500.

#### 4.2. Evaluation Metrics

Clustering accuracy (ACC) is the most widely used measurement for clustering quality [5, 4]. It is defined as follows:

\[
ACC = \frac{\sum_{i=1}^{N} \delta(y_i, map(c_i))}{N},
\]

where \( N \) is the number of data samples, \( y_i \) is the ground-truth label, and \( c_i \) is the label generated by the algorithm. \( \delta(x, y) \) is a delta function defined as: \( \delta(x, y) = 1 \) for \( x = y \), and \( \delta(x, y) = 0 \), otherwise. \( map(\bullet) \) is a permutation function which can be realized using the Hungarian algorithm [14]. A higher value of \( ACC \) indicates better clustering quality.

#### 4.3. Experimental Results

Table 1 shows the clustering accuracy results of compared methods and our method with node reduction ratios of 5X, 10X and 50X. The runtime of the eigen-decomposition and k-means steps are reported in Table 2. As observed, the proposed method can consistently lead to dramatic performance improvement, beating all competitors in clustering accuracy across all the three data sets: our method achieves more than 10% accuracy gain on USPS, 11% gain on MNIST and 15% gain on Covtype over the second-best methods; for the USPS and MNIST data sets, our method achieves over 17% and 18% accuracy gain over the standard spectral clustering method, respectively. The superior clustering results of our method clearly illustrate that spectrum-preserving concise representation can improve clustering accuracy by removing redundant or false relations among data samples.

As shown in Table 3, the 50X reduced graphs generated by our framework have only 138, 1182 and 8192 nodes for USPS, MNIST and Covtype data sets, respectively, thereby

[^1]: https://archive.ics.uci.edu/ml/
[^2]: https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/
[^3]: http://yann.lecun.com/exdb/mnist/


Table 1: Spectral clustering accuracy (%)

| Data Set | Standard SC | Nyström | KASP | LSC | Ours (5X) | Ours (10X) | Ours (50X) |
|----------|-------------|---------|------|-----|-----------|------------|------------|
| USPS     | 64.31       | 69.31   | 70.62| 66.28| 70.65     | 72.25      | 81.39      |
| MNIST    | 64.20       | 55.86   | 71.23| 58.94| 70.99     | 72.42      | 82.18      |
| Covtype  | 48.81       | 33.24   | 27.56| 22.60| 48.77     | 44.50      | 48.28      |

Table 2: Runtime (seconds)

| Data Set | Standard SC | Nyström | KASP | LSC | Ours (5X) | Ours (10X) | Ours (50X) |
|----------|-------------|---------|------|-----|-----------|------------|------------|
| USPS     | 0.72        | 0.29    | 0.16 | 0.22| 0.21      | 0.18       | 0.17       |
| MNIST    | 252.59      | 0.95    | 0.18 | 0.49| 0.59      | 0.35       | 0.21       |
| Covtype  | 128.70      | 5.48    | 0.70 | 3.77| 0.92      | 0.50       | 0.49       |

Table 3: Graph complexity comparison

| Data Set | |V_orig| |V_reduced| (5X)| |V_reduced| (10X)| |V_reduced| (50X) |
|----------|----------|-------|-------|------|-----|-------|------|-------|------|
| USPS     |         | 9,298 | 1,692 | 767  | 138 |
| MNIST    |         | 70,000| 13,658| 6,368| 1,182|
| Covtype  |         | 581,012| 104,260| 44,188| 8,192|

Fig. 3: ACC VS reduction ratio for the MNIST data set.

Fig. 4: Graph size VS reduction ratio for the MNIST data set.

allowing much faster eigen-decompositions. As shown in Fig. 3 and Fig. 4, with increasing node reduction ratio, the proposed method consistently produces high clustering accuracy. For the very large data set Covtype, our method is the only one that enables us to accelerate spectral clustering algorithm without loss of clustering accuracy.

5. CONCLUSION

We presented a novel framework for aggressively accelerating eigen-decomposition and improving accuracy for spectral clustering. We use spectrum-preserving node reduction to reduce the data size and remove the redundant relations among samples. Experimental results demonstrate that our method outperforms state-of-the-art methods by a large margin.
6. REFERENCES

[1] Feiping Nie, Dong Xu, Ivor Wai-Hung Tsang, and Changshui Zhang, “Spectral embedded clustering,” in Twenty-First International Joint Conference on Artificial Intelligence, 2009.

[2] Francis Bach and Michael Jordan, “Learning spectral clustering,” Advances in neural information processing systems, vol. 16, no. 2, pp. 305–312, 2004.

[3] Xia Wan and C-CJ Kuo, “A new approach to image retrieval with hierarchical color clustering,” IEEE transactions on circuits and systems for video technology, vol. 8, no. 5, pp. 628–643, 1998.

[4] Jialu Liu, Chi Wang, Marina Danilevsky, and Jiawei Han, “Large-scale spectral clustering on graphs,” in Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. AAAI Press, 2013, pp. 1486–1492.

[5] Xinlei Chen and Deng Cai, “Large scale spectral clustering with landmark-based representation,” in AAAI, 2011.

[6] Donghui Yan, Ling Huang, and Michael I Jordan, “Fast approximate spectral clustering,” in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009, pp. 907–916.

[7] Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik, “Spectral grouping using the nystrom method,” IEEE transactions on pattern analysis and machine intelligence, vol. 26, no. 2, pp. 214–225, 2004.

[8] Zhiqiang Zhao, Ying Zhang, and Zhuo Feng, “Towards scalable spectral embedding and data visualization via spectral coarsening,” in Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 869–877.

[9] Ulrike Von Luxburg, “A tutorial on spectral clustering,” Statistics and computing, vol. 17, no. 4, pp. 395–416, 2007.

[10] Feiping Nie, Cheng-Long Wang, and Xuelong Li, “K-multiple-means: A multiple-means clustering method with specified k clusters,” in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 959–967.

[11] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al., “On spectral clustering: Analysis and an algorithm,” Advances in neural information processing systems, vol. 2, pp. 849–856, 2002.

[12] Yongyu Wang, High Performance Spectral Methods for Graph-Based Machine Learning, Ph.D. thesis, Michigan Technological University, 2021.

[13] Oren E Livne and Achi Brandt, “Lean algebraic multi-grid (lamg): Fast graph laplacian linear solver,” SIAM Journal on Scientific Computing, vol. 34, no. 4, pp. B499–B522, 2012.

[14] Christos H Papadimitriou and Kenneth Steiglitz, Combinatorial optimization: algorithms and complexity, Courier Corporation, 1982.