Similarity Graph Learning and Non-linear Deep Representations for Spectral Clusterings

Ni Li\textsuperscript{1,2,4,5}, Manman Peng\textsuperscript{4}, Buwen Cao\textsuperscript{2,4,5}, Kenli Li\textsuperscript{1}, Keqin Li\textsuperscript{1,3}

\textsuperscript{1}College of Computer Science and Electric Engineering, Hunan University, Hunan, Changsha, 410082, China

\textsuperscript{2}College of Information and Electronic Engineering, Hunan, Yiyang, 413000, China

\textsuperscript{3}Department of Computer science, State University of New York, New Paltz, NY 12561, USA

\textsuperscript{4}Engineering Research Center of DongTing Lake regional ecological environment intelligent monitoring and disaster prevention and Mitigation, Hunan, Yiyang, 413000, China

\textsuperscript{5}Key Laboratory of city computing and IoT, Hunan City University, Hunan, Yiyang, 413000, China

* lini28698@163.com

Abstract. Spectral clustering is one of the most popular clustering approaches. Despite its good performance and strong theoretical supports, it is limited to high complexity of the graph Laplacian similarity matrix construction and eigen-decomposition problems. Recently, deep learning has been successfully adopted in graph representation. In the paper, we jointly learn the manifold graph construction and non-linear low-dimension mapping of the graph. In addition, we theoretically proved that our model according to spectral clustering theory. Meanwhile, we use the proposed non-linear coders as the building blocks to formulate a deep structure to further refine features of layer wise fashion. Extensive experiments on clustering tasks demonstrate that our method performs well in terms of both clustering accuracy and normalized mutual information (NMI).

Keywords: spectral clustering, similarity matrix, graph representation, deep learning

1. Introduction

With network technological progress, a large number of images, texts, and graph information have emerged. However, these data often need artificial intelligence or a specialized machine learning technology to study them. Clustering provides a useful tool to analyze such data and reveal the underlying structure.

Spectral Clustering [1-2] is a highly popular clustering algorithm. It embeds the graph in eigenspace of the Laplacian matrix, derived from the graph similarity matrix. It is a clustering...
algorithm based on three steps: the similarity matrix construction, the low dimensional representation and Kmeans clustering. Recently, its non-parametric property and fast implementations [3-4] made it very suiTab for large-scale data analysis under complex networks. However, the similarity matrix construction and eigen-decomposition are the main factor [5-10] that affect the performance of the spectral clustering algorithm.

So two important issues must be addressed. Firstly, we must understand how to more accurately capture the graph structural similarity matrix construction for weighted graphs? Secondly, we must understand how to develop an improved graph of a graph with a low-dimensional representation.

In this paper, it offer a new perspective of spectral clustering. Firstly, we argue that for the similarity matrix construction step, the original K-nearest neighbor (Knn ) graph [11] of manifold learning is a representation of the graph. We can improve the Knn graph to better capture the structure of large-scale complex network. Secondly, the low dimensional representation step, SVD(Singular Value Decomposition) [3-12] used to create the graph performs a linear graph embedding, however the non-linear graph embedding method is better.

Therefore, we investigate the effectiveness of using deep learning [10-15] and manifold learning [11-16]. The deep learning method is applied to the manifold similarity matrix instead of the Laplacian similarity matrix used in the spectral clustering. We use sparse representations for the graph similarity matrix, which usually can improve the clustering index-ACC(accuracy), since it can reduce noises that influence the clustering results. In representation learning we take advantage of deep structure to build features. Deep structure can employ advantageous layer wise training to obtain fine tuning and help to find a better local solution to meet the clustering objective. We call our proposed method SEG_SC (Sparse Encoder Graph _Spectral Clustering ). Our contributions are as follows:

- We jointly study manifold learning similarity matrix of large-scale networks and nonlinear low-dimension mapping of the similarity matrix. The model architecture with the kmeans is defined. The model is similar to the spectral algorithm.
- Theoretically, at first, we prove that our model employs a clustering method consistent with the objective function of spectral clustering. Secondly, we capture the non-linear feature of large-scale networks.
- Experiments show that our model can improve the clustering performance of large-scale networks.

This paper presents the Knn graph and deep learning background knowledge and our related work in Section 2; theoretically illustrates the overall model and describes the SEG_SC algorithm in Section 3; the experimental results are shown and analyzed in Section 4, and finally concludes the paper in Section 5.

2. Background and Related Work

For the construction of the graph, a similarity matrix expressing the vertexes and complex relationships is used. Traditional methods [2-6] usually adopt a high dimensional dense matrix to express the graph. Additionally, various application algorithms are designed on this similarity matrix [5-9]. However, with the increasing amount of graph scale, this kind of high latitude dense representation is faced with more computing space and time, and the accuracy of the graph structure is very poor.

Knn graph [11] is a manifold learning method. Graph construction of Isometric Mapping (Isomap) [15] and Locally Linear Embedding (LLE)[16] all are included in the Knn graph. The model has good generalization ability, and draws a topological manifold. If a high latitude manifold is embedded in a low dimensional manifold space, then the property of Euclidean space is still present locally. The Knn algorithm has the characteristics of stability, simplicity, and efficiency, but in the case of large data, it also has its own shortcomings; large-scale data or high-dimensional data will lead to a dramatic increase in computational complexity, and reduce the clustering results. Considering the nearest neighbor search method, D. Cai et al. [3-13] improved the clustering algorithm based on CLUE
(Clustering Using Representatives). The generated representative points represent each cluster to complete the calculation. Noise has little effect on this method.

For the low-dimension graph representation, there have been many effective methods[3-20], which have assigned a low dimensional dense vector representation of graph node. Recently, most models[3-20] of learning graph representations have employed SVD to perform a linear dimension reduction. Linear coding [6-20] had yielded sparse representations where each node was represented as a linear representation of basic vectors.

However, these methods have been not explored in non-linear dimension reduction techniques, or their graph constructions were not suitable for the large-scale network. Recently, deep structure[13-18] has been widely applied for learning graph representations. The greedy layer training strategy and improvement of computing performance were implemented. Stacked autoencoders[17-18] is regarded as an effective method of mapping from primitive space to low dimensional space, which yield highly non-linear projections. Recently, sparse autoencoders were used to replace the SVD step[17-18] in spectral clustering. F. Tian et al. [18] proposed a nonlinear spectral clustering algorithm based on deep learning. In this method, the graph is embedded by an automatic encoder in depth learning, and then clustered by using a K-means clustering algorithm. The work of Fei et al.[18] is of great significance to the study of spectral clustering algorithm for nonlinear mapping, but there are some problems as follows: 1) For large-scale data, the representation of the graph structure by using a high-dimensional similarity matrix directly can not reflect the essential characteristics of the graph structure; 2) Simply put the framework of the nonlinear spectral clustering algorithm has been suggested, but no theoretical proof has been shown:

![Fig 1. Main components](image)

3. Model Description

Our proposal model is motivated by the three-step scheme for spectral clustering, which consists of the similarity graph construction step, the low-dimension graph representation and K-means clustering algorithm. Our research will first explain the similarity matrix reconstruction of spectral clustering, and then demonstrate the low-dimensional graph representation. Our main components is shown in Fig 1.

If our network is an undirected weighted graph $G = (V,A,E)$, where $V = \{v_1, v_2, ..., v_m\}$ is node set, $E = \{e_i\}$ is the edge set and $A = \{A_i\}$ is weighted value of node set. The SEG.SC spectral clustering aims to find a disjoint partition $C = \{C_1, C_2, ..., C_k\}$, where $k$ is the cluster number.

3.1. Sparse Normalized Similarity Matrix
In the traditional spectral clustering algorithm, the similar matrix is a normalized Laplacian matrix $L = \frac{D-A}{D}$ where $D_{ii} = \sum_j A_{ij}$ is degree matrix, and $S = \frac{A}{D}$ is a kind of graph similarity matrix, and actually an expression as:

$$S(v_i, v_j) = \frac{A_{ij}}{\sum_{t \in N(v_j)} A_{it}}$$  \hspace{1cm} (3.1)

, which $N(v_i)$ is neighbors of node $v_i$.

Knn learning is a commonly manifold learning method. Using a test sample, based on a distance measure, we can find out the K’s nearest neighbors. Based on the information about these k neighbors, we can combine the normalized Laplacian matrix of spectral clustering with the idea of KNN graph construction similarity matrix, and obtain the suiTab similarity matrix of the weighted network. It can be described in the following formula:

$$S(v_i, v_j) = \frac{d(v_i, v_j)}{\sum_{t \in N(v_i), t \in \{1, 2, ..., k\}} A_{it}}$$  \hspace{1cm} (3.2)

where $d()$ is the distance measured method. $v_i$ is selected as the $\kappa$ nearest neighbor to $v_j$. For the purpose of the large-scale graph, we use a random method to select $\rho$ landmark samples from the network, generating set $Q = \{q_1, q_2, ..., q_\rho\}$. Each vertex only calculates the relation matrix $Z$ with its $\kappa$ nearest Q. It expressed as follows:

$$Z(v_i, v_j) = \frac{d(v_i, u_j)}{\sum_{t \in Q \cap N(v_i), t \in \{1, 2, ..., k\}} d(v_i, u_t)}$$  \hspace{1cm} (3.3)

Meanwhile, $q_k$ is the $\kappa$ nearest neighbor of $v_i$. After getting the value of $Z$, the normalized similarity matrix is:

$$S = \tilde{Z}^2 \tilde{Z}$$  \hspace{1cm} (3.4)

where $\tilde{Z} = D^{-\frac{1}{2}}Z$, $D$ is diagonal matrix, and $D_{ii} = \sum_j D_{ij}$

3.2. Nonlinear Low-dimensional graph Representation

First, we will evaluate the objective function of our nonlinear low-dimensional graph representation, then analyze that linear expression based on the idea that our similarity matrix $S$ is equal to traditional spectral clustering. Finally, we present a deep network framework based on objection function.

3.2.1. Single-layer Nonlinear Low-dimensional Representation

In a graph dimension reduction method based on a nonlinear projection, the projection of the sample point in the low-dimensional coordinated system is as follows:

$$R_{ij} = f(W_j^T S_i)$$  \hspace{1cm} (3.5)

where $S_i$ is the i-th row vector of the similarity graph matrix $S$ in Eq (3.4), $W_j$ is a parameter vector. $R_{ij}$ is the j-th column value of the low-dimension space projection of $S_i$, $R_i = (R_{i1}, R_{i2}, ..., R_{id})$. In addition, we can reach the equation
where  is the projection reconstruction of . Then, the distance between the original sample and the projection reconstruction sample is expressed as:

\[
\sum_{j=1}^{m_i} \left\| \sum_{i=1}^{m} g(R_i W_j) - S_i \right\|^2 = \sum_{i=1}^{m} g(R_i W) g(W^T R_i) - 2 \sum_{i=1}^{m} g(R_i W) S_i + \text{const} \\
\propto - \text{tr}(\sum_{i=1}^{m} g(W R_i^T) g(R_i W^T))
\]  

(3.7)

where  ,  is the sample number and  is the dimension number after dimensional reduction, and  .

Meanwhile, sample covariance of  is \[ \sum_j g(W^T Z_j) g(Z_i^T W) \]. If the projective samples are separated as much as possible, the sample variance of all nodes should be maximized, so the optimization objective is

\[
\max \text{tr}(\sum_{i=1}^{m} g(W R_i^T) g(R_i W^T)) \text{ s.t. } W^T W = I
\]  

(3.8)

Therefore, the model satisfies the distance from the sample point of the hyperplane, such that it is enough to close (the nearest reconstruction) and the projection samples in hyperplane should be separated as far as possible (maximum separability).

Especially, when function  is a linear function, and  , the objective function E.q (3.8) is equivalent to

\[
\max \text{tr}(WSW^T), \text{s.t. } W^T W = I
\]  

(3.9)

Therefore, it is a typical spectral clustering algorithm objection function.

### 3.2.2. Stacked Deep Network

As stated in the previous descriptions, we can treat  as the training set containing  nodes. We consider the deepening network in each layer.

It is similar to the autoencoder, we reorganize the objective function as:

\[
\arg \max_{\eta_i} \sum_{i=1}^{n} \text{tr}(\sum_{j=1}^{m_i} g(W^{(i)}(R_j^{(i)} W_j^T)^T) g(R_j^{(i)} (W^{(i)})^T)) \text{ s.t. } W^T W = I
\]  

(3.10)

where  is the layer number of the deep network.

Combinatorial clustering of Sparse normalized similarity matrix + Stacked deepening network encoder+ Kmeans names it Sparse-Encoder-Graph Spectral Clustering (SEG SC) in the paper. The clustering process is described in Algorithm 1

---

**Algorithm 1 Sparse-Encoder-Graph spectral clustering**

Require:  node graph  , degree matrix  , where  is the degree of node  , network layer number  , the number of nodes in layer 1 is  is original data samples
Ensure: $\kappa$ clusters $C_1, C_2, \ldots, C_\kappa$

1: the similarity matrix $S$ according to Eq (3.4)
2: for $i = 1 \text{ to } n$
3: Build a stacked denoise deepening network with the similarity matrix $S^{(i)}$ based on Eq (3.4)
4: Train the stacked denoise deepening network by optimizing Eq (3.10). Obtain the hidden layer activation $h^{(i)}$
5: Set $X^{(i+1)} = R^{(i)}$
6: end for
7: Run k-means on $X^{(n)}$.

4. Experimental Evaluation

Firstly, we introduce the datasets, evaluation metrics, and the benchmark algorithms. Then, several experiments were conducted to demonstrate the effectiveness of the proposed SEG_SC.

4.1. Datasets

We evaluate its clustering performance on real data sets. Their details are described as:

| Data Set    | Number of samples | Number of features | Number of clusters |
|-------------|-------------------|--------------------|--------------------|
| wine        | 178               | 13                 | 3                  |
| pendigit    | 3498              | 16                 | 10                 |
| mnist       | 4000              | 784                | 10                 |
| blogdata    | 10312             | 1                  | 39                 |
| 20newgroups | 18846             | 26214              | 20                 |

| Tab.1 Dataset details

- **Wine** The data set describes the chemical analysis of wines grown. They are divided into three categories.
- **Mnist** They are handwritten digits from Yann LeCun’s page. Each image is represented as a 784 dimensional vector. There are 4000 images.
- **Pendigit** A handwritten digital data set of 3498 samples with 16 features from 44 writers.
- **Blogdata** A relationship network of social relationships provided by blogger authors. The network has 10,321 nodes, 333,983 edges and 39 different cluster\_labels. The labels represent blogger interests inferred and provided by the bloggers.
- **20newgroups** A language network, which has 18,846 newsgroup documents with 26,214 features and is partitioned by 20 different groups.

The specific informations of these data sets are shown in Tab 1. For these real data sets, we use ACC and NMI(normalized mutual information) to evaluate their clustering performance.

4.2. Evaluation Metric

We introduce two performance measurements in our evaluations ACC and NMI of the clustering results.

Let $\text{cluster\_label}$ and $\text{truth\_label}$ be known as the clustering results of the algorithm and the actual clustering results are expressed, respectively. The ACC is defined as follows:

$$ACC = \frac{\sum(\text{cluster\_label} == \text{truth\_label})}{m} \tag{4.1}$$

where $m$ is the total number of data samples.
NMI is the mutual information entropy between the resulting clustering labels and the ground truth labels, and followed by normalization operation which guarantees the NMI ranges from 0 to 1, which can be written as:

\[
NMI = \frac{\sum_{i=1}^{K} \sum_{j=1}^{K} n_{i,j} \log \left( \frac{n_{i,j}}{n_i n_j} \right)}{\sqrt{\left( \sum_{i=1}^{K} n_i \log \left( \frac{n_i}{n} \right) \right) \left( \sum_{j=1}^{K} n_j \log \left( \frac{n_j}{n} \right) \right)}}
\]  

(4.2)

where, \( n_i \) expresses the sample number of cluster \( C_i (1 \leq i \leq K) \). \( n_{i,j} \) denotes the sample number in both cluster \( C_i \) and category \( C_j \).

4.3. Benchmark Algorithms

Following is a list of information concerning the experimental setting of each method:

- **Spectral Clustering** [2] is original spectral clustering with the Gaussian kernel. The sparse Knn graph is used for graph construction.

- **PCA** [16] is a common matrix dimension reduction method. PCA defined a loss function based on the distance between the original samples and the projection reconstruction samples. It is a linear graph embedding method. Then we run a k-means algorithm to obtain the clustering results.

- **graphencoder** [18] is a recently proposed method for learning graph representations. It explores the nonlinear graph embedding by employing deep learning, then the Kmeans is executed.

4.4. Experiment Settings and Results

**Similarity Matrix** Note for all datasets, each sample is normalized to have the same unit length. We set the number of neighbors in sparse Knn graph search at 5. In addition, we select the Gaussian kernel function, which is one of the most commonly used in the literature in function \( d() \) of Eq(3.3).

We experiment with our model scheme and vary the number of landmarks from 200 to 1200 in mnist and pendigit dataset, which is shown in Fig. 1. From Fig.1, it shows that the ACC values are tortuous ascending curves with changing landmark values, which is because of the random selection method. Apparently, more landmarks improve the final clustering performance. Without loss of generality, we set the number of landmarks at 1000, which also provides moderate accepTab performance.

![Fig.2 ACC vs. number of landmarks on mnist and pendigit datasets](image_url)
Deep Structures We implemented our SGE SC to show the number of layers in the deep structure. We Figd out the ACC values of SGE SC on the pendigit dataset in Fig.3. The Fig shows that the value of ACC increases with the increase of the number of layers.

The number of deep layers in wine, pendigit and mnist is 3 layers, and the other graphs is 5 layers. The number of nodes in each layer is listed in Tab 2. In deep network, we use sigmoid function as the activation function of each layer. The sparse value, which is the value that the hidden layer activations aim to reach, is 0.05.

Clustering Performance. The clustering results on 4 kinds of datasets are summarized in Fig 3 and Fig 4. Results from these datasets demonstrate that our method(SEG SC) provides comparable high ACC and NMI.

We can be seen from the results in Fig 3 and Fig 4 that SEG_SC does have a better ACC and NMI compared than other benchmark algorithms. In Fig 4, my method gains a larger value in the clustering accuracy, especially the large-scale data .blogdata and 20newsgroups, which is because of the graph similarity optimization. Meanwhile, since the graph employs a non-linear projection, from Fig. 1 we can see that NMI value tends to be the best in the Blogdata and 20newsgroup datasets.

The advantage of our method lies in the sparse similarity feature representation, instead of the original graph similarity and in the nonlinear coding scheme, instead of the eigen-decomposition. Specifically, the low dimension features fed to kmeans in the last step are favored by the spectral clustering. In addition, the nonlinear coding scheme reduces the dimensions favored by the natural scene.
Fig. 4  Clustering NMI vs. cluster algorithms on blogdata and 20newsgroup datasets

|       | wine  | 178-128-64 | mnist  | 4000-1280-640 | blogdata | 10312-5096-2048-1280-640 | 20newgroup | 18846-5096-2048-1280-640 |
|-------|-------|------------|--------|--------------|----------|-------------------------|------------|-------------------------|
| Spectral Clustering | 0.71   | 0.52       | 0.58   | 0.48         |          |                         |            |                         |
| PCA   | 0.73   | 0.50       | 0.59   | 0.59         |          |                         |            |                         |
| Graphencoder | 0.84   | 0.38       | 0.59   | 0.51         |          |                         |            |                         |
| SEG_SC | 0.86   | 0.55       | 0.65   | 0.62         |          |                         |            |                         |

Tab. 2  Neural network structures

Tab. 3  Comparisons with clustering methods on accuracy

5. Conclusion

In this paper, SEG_SC is obtained via optimization of graph similarity matrix and low-dimension representations of a similarity matrix. Our work is estimated as follows.

Firstly, it is proposed a spectral clustering scheme based on a deep neural network took a sparse similarity graph based on the landmarks and Knn method and proposed a graph encoding that each node is representation as a low dimensional vector.

Secondly, we have theoretically proved our method effectiveness. Our method describes the objection function Eq (3.8) based on the sparse normalized similarity matrix and nonlinear low-dimensional graph representation. In Eq (3.9), it points out that spectral clustering is the linear version of our method.

Thirdly, experiments on real-world datasets have showed that the performance of the SEG_SC outperformed several state-of-art baseline algorithms. We conclude that the spectral clustering via deep learning can improve the performance of large-scale graphs.

This study introduces a new spectral clustering scheme, this strategy, even if imperfect, may explore the graph underlying structure via sparse similarity graph and deep learning. It leads to improve cluster performance for graphs.

References

[1] U Shaham, K Stanton, H Li, B Nadler, R Basri and Y Kluger 2018 Spectralnet: Spectral clustering using deep neural networks Proc. Int Conf on Learning Representations (Canada: Vancouver)

[2] A Y Ng, M I Jordan and Y Weiss 2001 On spectral clustering: analysis and an algorithm Proc. Int Conf on Neural Information Processing Systems: Natural and Synthetic(Canada: Vancouver) pp849–856

[3] D Cai and X Chen 2015 Large scale spectral clustering via landmark-based sparse representation IEEE Transactions on Cybernetics 45 pp1669–1680
[4] D Cai, X He, J Han and T S Huang 2011 Graph regularized nonnegative matrix factorization for data representation *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33** pp1548

[5] C Liu, J Liu and Z Jiang 2014 A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks *IEEE Trans Cybern* **44** pp2274–2287

[6] D L Donoho and M Elad 2003 Optimally sparse representation in general (nonorthogonal) dictionaries via minimization *Proc of the National Academy of Sciences of the United States of America* **100** pp2197

[7] S Cao, W Lu and Q Xu 2015 Grarep: Learning graph representations with global structural information Proc. *Inf on Information and Knowledge Management* (Australia:Melbourne)

[8] B Perozzi, R Altfou and S Skiena 2014 Deepwalk: online learning of social representations *ACM SIGKDD Conf on Knowledge Discovery and Data Mining* (America:New York) pp701–710

[9] J Tang, M Qu, M Wang, M Zhang, J Yan and Q Mei 2015 Line: Large-scale information network embedding *Int Conf of World Wide Web* (Italy:Florence) **2** pp1067–1077

[10] Y Xia, D He, T Qin, L Wang, N Yu, T Y Liu and W Y Ma 2016 Dual learning for machine translation *Conf and Workshop on Neural Information Processing Systems* (Canada:Vancouver)

[11] N S Altman 1992 An introduction to kernel and nearest-neighbor nonparametric regression *American Statistician* **46** pp175–185

[12] G H Golub and C Reinsch 1970 Singular value decomposition and least squares solutions *Numerische Mathematik* **14** pp403–420

[13] M Shao, S Li, Z Ding and Y Fu 2015 Deep linear coding for fast graph clustering *Proc. Inte Joint Conf on Artificial and Intelligence* (Argentina:Buenos Aires) pp3798–3804

[14] M Chen, Z Xu, K Weinberger and S Fei 2012 Marginalized denoising autoencoders for domain adaptation *Proc. Inte Conf on Machine Learning* (UK:London)

[15] G E Hinton and R S Zemel 1993 Autoencoders, minimum description length and helmholtz free energy *Inte Conf on Neural Information Processing Systems* (USA:Colorado) pp3–10

[16] H. Abdi and L. J. Williams 2010 Principal component analysis *Wiley Interdisciplinary Reviews Computational Statistics* **2** pp433-459

[17] S Cao, W Lu and Q Xu 2016 Deep neural networks for learning graph representations *in Thirtieth AAAI Conf on Artificial Intelligence* (USA :Phoenix) pp1145–1152

[18] F Tian, B Gao, Q Cui, E Chen and T Y Liu 2014 Learning deep representations for graph clustering *in Twenty-Eighth AAAI Conference on Artificial Intelligence* (Canada:Quebec) pp1293–1299

[19] Y Zhang and Z H Zhou 2010 Multilabel dimensionality reduction via dependence maximization *Acm Transactions on Knowledge Discovery from Data* **4** pp1–21

[20] J Hou and R Nayak 2015 Robust clustering of multi-type relational data via a heterogeneous manifold ensemble *IEEE Inte Conf on Data Engineering* (South Kerman:Seoul) pp615–626