Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Incremental spectral clustering by efficiently updating the eigen-system

Huazhong Ning[a,*], Wei Xu[b], Yun Chi[b], Yihong Gong[b], Thomas S. Huang[a]

[a]ECE Department, University of Illinois at Urbana-Champaign, USA
[b]NEC Laboratories America, Inc., USA

ARTICLE INFO

Article history:
Received 20 November 2008
Received in revised form 31 May 2009
Accepted 5 June 2009

Keywords:
Incremental clustering
Spectral clustering
Incidence vector/matrix
Graph
Web–blogs

ABSTRACT

In recent years, the spectral clustering method has gained attentions because of its superior performance. To the best of our knowledge, the existing spectral clustering algorithms cannot incrementally update the clustering results given a small change of the data set. However, the capability of incrementally updating is essential to some applications such as websphere or blogsphere. Unlike the traditional stream data, these applications require incremental algorithms to handle not only insertion/deletion of data points but also similarity changes between existing points. In this paper, we extend the standard spectral clustering to such evolving data, by introducing the *incidence vector/matrix* to represent two kinds of dynamics in the same framework and by incrementally updating the eigen-system. Our incremental algorithm, initialized by a standard spectral clustering, continuously and efficiently updates the eigenvalue system and generates instant cluster labels, as the data set is evolving. The algorithm is applied to a blog data set. Compared with recomputation of the solution by the standard spectral clustering, it achieves similar accuracy but with much lower computational cost. It can discover not only the stable blog communities but also the evolution of the individual multi-topic blogs. The core technique of incrementally updating the eigenvalue system is a general algorithm and has a wide range of applications—as well as incremental spectral clustering—where dynamic graphs are involved. This demonstrates the wide applicability of our incremental algorithm.

1. Introduction

Spectral clustering is notable both for its theoretical basis of graph theory and for its practical success. It recently has many applications in data clustering, image segmentation, web ranking analysis, and dimension reduction. Spectral clustering can handle very complex and unknown cluster shapes in which cases the commonly used methods such as K-means and learning a mixture model using EM may fail. It relies on analyzing the eigen-structure of an affinity matrix, rather than on estimating an explicit model of the data distribution [1,2]. In other words, the top eigenvectors of the graph Laplacian can unfold the data manifold to form meaningful clusters [3].

However, nearly all existing spectral approaches are off-line algorithms, and hence they cannot be directly applied to dynamic data set. Therefore, to handle evolving data set, e.g., web data, there is a need to develop efficient algorithms for inductive spectral clustering to avoid expensive recomputation of the solution from the scratch. An intuitive approach is fixing the graph on the training data and assigning new test points to their corresponding clusters by the nearest neighbor in the training data [3]. However, the error will accumulate quickly when more test points that are close to the cluster boundaries are added. In this paper, we extend the spectral clustering to handle evolving data by incrementally updating the eigenvalue system, which achieves more accurate results while requires low computational cost.

There exist incremental clustering algorithms [4–6] that are designed to handle only insertion of new data points. However, data sets, such as web pages and blogs, require the incremental algorithms to handle not only insertion/deletion of nodes but also similarity changes between existing nodes. Fig. 1 gives a toy example where a graph evolves from (a) to (b), with a similarity change of 0.5 added to the edge CD and a new node G connected to node F. In Fig. 1(a), the graph should be cut at the edge CD; while in Fig. 1(b) the cut edge is DE due to the similarity change on edge CD.

We handle the two kinds of dynamics in the same framework by representing them with the *incidence vector/matrix* [7]. The Laplacian matrix can be decomposed into the production of two incidence matrixes. A similarity change can be regarded as an incidence vector appended to the original incidence matrix. And an insertion/deletion of a data point is decomposed into a sequence of similarity changes.

[*] Corresponding author. Tel.: +1 217 417 6820.
E-mail addresses: hning2@ifp.uiuc.edu, hning2@uiuc.edu (H. Ning)
URL: http://www.ifp.uiuc.edu/~hning2.

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.06.001

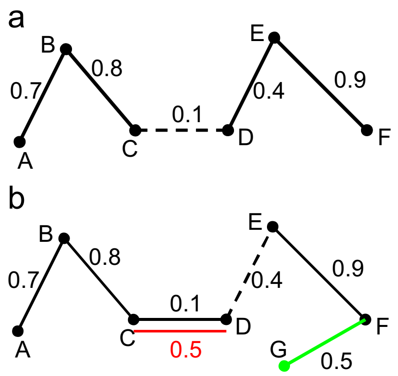

Fig. 1. A toy example of evolving data. (a) Before evolution. (b) After evolution. The dash lines are the cut edges.

Each newly added incidence vector (similarity change) may induce increment to the Laplacian and degree matrixes, and we approximate the corresponding increments of the eigenvalues and eigenvectors by omitting the influence of the data points outside the spatial neighborhood of the updating data points. In this way, the eigen-system and the cluster labels are incrementally updated as data points are inserted/deleted or similarity changes occur.

This approach is useful to the applications where the similarity matrix is sparse and where both the data points and their similarities are dynamically updated. An example is the community discovery of the web–blogs. The key observation is that a link reference from an entry of a source blog to an entry of a destination blog serves as an endorsement of the similarity between the two blogs. A graph can be constructed based on the similarities between the web–blogs, and communities (clusters) can be discovered by spectral clustering. However, web–blogs are evolving, and new blogs and new links are added or removed every day. Therefore, the standard spectral clustering cannot be used to online monitor the web–blogs because of the huge number of blogs and, in turn, of the high computational cost. For sparse similarity matrix, Lanczos method [8] may save much cost to solve the eigenvalue problem. But it is still impractical to recompute the solution from the scratch at each time instance the data set is updated, especially when the web–blogs are huge. On the contrary, our approach applied to the web–blog data achieves similar accuracy but with much lower computational cost, compared with recomputation by the standard spectral clustering.

It is worth to note that the core idea of our incremental clustering is dynamic updating of the (generalized) eigenvalue system. Actually it is a general algorithm that can also be applied to many other problems involving dynamic graphs. These problems require to solve the (generalized) eigenvalue system at each time the graph is updated. In Section 6, three related problems are stated and solved by this algorithm. The first problem is to choose edges from a candidate set to maximize the *algebraic connectivity* of a graph. *Algebraic connectivity* is the second smallest eigenvalue of the graph Laplacian that measures how well connected the graph is [9]. The second problem is to find the most significant edge of a graph. The last problem is related to linear embedding. These problems demonstrate the wide applicability of our algorithm.

This paper is an extension of our previous work [10]. Our previous work presented a basic algorithm to efficiently update the (generalized) eigenvalue system given a small change of the data set. It approximates the increments of eigenvalues and eigenvectors with first order error. Based on our previous work, this paper gives a second order approximation for the increments by alternately refining the eigenvalues and eigenvectors, respectively. Then more experiments are carried to show that the refinement algorithm achieves significant improvement over our previous work. In this version, our algorithm is also applied to some other related problems involving dynamic graphs, which demonstrates the wide applicability of our incremental algorithm. Besides these, discussions on the number of clusters, more related work, and some other content are added in this paper to complement the previous version. The contributions of our work are summarized as follows:

1. We declare two kinds of dynamics existing in the evolving data: similarity change and insertion/deletion of data points. And then the incidence vector/matrix is introduced to represent the two dynamics so that our incremental clustering can handle them in the same framework.
2. Based on (but not limited to) normalized cut, the incremental spectral clustering is formulated as the problem of dynamically updating the eigen-system given a small similarity change. We give a closed-form solution to the eigenvalue increment with first order error and an approximate solution to the eigenvector increment.
3. To improve the accuracy of the increments, we propose an iterative algorithm that alternately refines the eigenvalues and eigenvectors. It approximates the increments with the second order error.
4. Our algorithm is also applied to solve some other related problems involving dynamic graphs. This demonstrates the wide applicability of our algorithm.
5. We carry intensive experiments on the real blog data set. The incremental approach can discover not only the stable blog communities but also the evolution of the individual multi-topic blogs, while the computational cost is very low.

This paper is organized as follows. In the next section we focus on related work. Section 3 describes the basic formulations. Section 4 presents the incremental algorithm for spectral clustering. Then the algorithm is discussed in Section 5. And it is also applied to some other related problems in Section 6. Section 7 gives the experimental results. The paper is concluded in Section 8.

2. Related work

To the best of our knowledge, our approach is the first work accomplishing the task of incremental spectral clustering that can handle not only insertion/removal of data points but also similarity changes. But there is still a large volume of literature related to our work, including topics on spectral methods, stream data, incremental PageRank, evolutionary clustering, and time series data.

The spectral method is where our work starts. Our work is based on normalized cut [2] but can be extended, without major modifications, to many other spectral methods involving solving eigenvalue systems. Spectral clustering evolved from the theory of spectral graph partitioning, an effective algorithm in high performance computing [11]. Recently there is a huge volume of literature on this topic. Ratio cut objective function [12,13] naturally captures both mincut and equipartition, the two traditional goals of partitioning. This function leads to eigenvalue decomposition of the Laplacian matrix. Shi and Malik [2] proposed a normalized cut criterion that measures both the total dissimilarity between the different groups as well as the total similarity within the groups. The criterion is equivalent to a generalized eigenvalue problem. Ding et al. [14] presented a min–max cut and claimed that this criterion always leads to a more balanced cut than the ratio cut and the normalized cut. Unlike the above approaches, Ng et al. [1] proposed a multi-way clustering method. The data points are mapped into a new space spanned by the first k eigenvectors of the normalized Laplacian. Clustering is then performed with traditional methods