Pattern Recognition 81 (2018) 484–496

Contents lists available at ScienceDirect

Pattern Recognition

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

Knowledge transfer for spectral clustering

Wenhao Jiang[a,*], Wei Liu[a], Fu-lai Chung[b]

[a] Tencent AI Lab, Shenzhen, Guangdong 518075 China
[b] Department of Computing, Hong Kong Polytechnic University, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 23 January 2017
Revised 23 February 2018
Accepted 19 April 2018

Keywords:
Transfer learning
Spectral clustering
Co-clustering
Multi-task learning

ABSTRACT

Many real-world applications propose the request for sharing knowledge among different tasks or datasets. Transfer learning has been proposed to solve this kind of problems and it has been successfully applied in supervised learning and semi-supervised learning settings. However, its adoption in clustering, one of the most classical research problems in machine learning and data mining, is still scarce. Spectral clustering, as a major clustering algorithm with wide applications and better performance than k-means typically, has not been well incorporated with knowledge transfer. In this paper, we first consider the problem of learning from only one auxiliary unlabeled dataset for spectral clustering and propose a novel algorithm called transfer spectral clustering (TSC). Then, it is extended to the settings with multiple auxiliary tasks. TSC assumes the feature embeddings being shared with the auxiliary tasks and utilizes co-clustering to extract useful information from the auxiliary datasets to improve the clustering performance. TSC involves not only the data manifold information of individual task but also the feature manifold information shared between related tasks. An in-depth explanation of our algorithm together with a convergence analysis are provided. As demonstrated by the extensive experiments, TSC can effectively improve the clustering performance by using auxiliary unlabeled data when compared with other state-of-the-art clustering algorithms.

1. Introduction

Clustering aims at finding groups of objects so that the objects in the same group are relatively similar while the objects in different group are relatively dissimilar. In the past decades, many clustering algorithms have been proposed, such as k-means clustering [1], spectral clustering [2,3], Bregman divergence based clustering [4], etc. Aiming at improving the clustering performance, must-link or cannot-link constraints [5,6], discriminative analysis [7], robustness [8–10], relationships among tasks [11,12] and auxiliary labeled data [13,14] have been employed in clustering.

Spectral clustering algorithms [2,3] are well-known methods that use manifold information contained in the sample distribution to find the corresponding embeddings, which are then used to carry out the clustering tasks by running k-means. Spectral clustering very often outperforms the traditional clustering algorithms such as the k-means algorithm. Hence, spectral clustering has been widely applied in various real-word applications, such as image segmentation [3], signal processing [15] and text clustering [16]. Spectral co-clustering was an extension of spectral clustering, which improves clustering performance with the help of the clustering of features. It seeks embeddings for both samples and features and performs clustering simultaneously. To further improve the performance of spectral clustering, we introduce the process of transferring knowledge from auxiliary unlabeled datasets. The transferring is realized by combing spectral clustering and spectral co-clustering on multiple datasets. We assume that the different tasks share the same *feature embedding*. This assumption is quite intuitive. For example, when we cluster news of this year with the help of old news, the meanings of words do not change with time. The semantic meanings of *intelligence* are very similar in the newspaper of the year 2015 and 2005. Hence, the feature embeddings (representations) of *intelligence* should be the same for the two tasks. Similar observations are also noticed in image clustering. The *tires* as a part of an *aircraft* in an image from one task should have very similar representation with the *tires* as a part of a *car* in an image from the auxiliary task. Therefore, the embeddings of features from different tasks could be used as a bridge to link and facilitate the clustering of samples as illustrated in Fig. 1.

In this paper, a transfer spectral clustering (TSC) algorithm is proposed and its preliminary version has appeared in [17], which can transfer knowledge between only two tasks. To overcome this limitation, we extend it to the setting with multiple auxiliary tasks and thus a multi-task TSC is derived. The proposed method works

* Corresponding author.
E-mail addresses: cswhjiang@gmail.com (W. Jiang), wliu@ee.columbia.edu (W. Liu), cskchung@comp.polyu.edu.hk (F.-l. Chung).

https://doi.org/10.1016/j.patcog.2018.04.018
0031-3203/

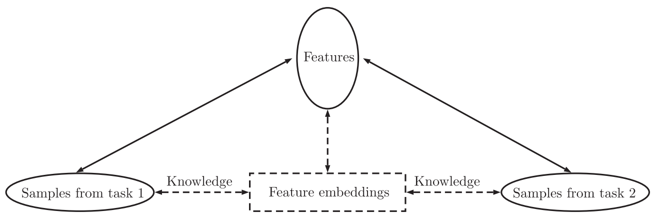

Fig. 1. Illustration of the idea behind transfer spectral clustering.

on the assumption that the related tasks share the same low dimensional feature embedding. This assumption could be achieved by introducing bipartite graph co-clustering. TSC involves not only the data manifold information of the individual tasks but also the feature manifold information shared among tasks. The experimental results show that TSC can greatly improve the clustering performance by effectively using auxiliary unlabeled data.

This work is presented as follows. In Section 2, the related works are firstly highlighted. After revisiting spectral clustering in Section 3, the formulation of our method is given in Section 4, which also describes the corresponding optimization method. In Section 5, the experimental results are reported. In Section 6, we give the conclusions and discuss the future works.

2. Related works

Our method is related to transfer learning, multi-task clustering and multi-view spectral clustering. Some corresponding related works are highlighted below.

2.1. Transfer learning

Transfer learning [18–20] attempts to improve the learning performance on a target dataset by utilizing auxiliary datasets which usually have different distributions. It has been proved to be beneficial in practice [21,22]. Many works have been done on supervised transfer learning [23–26] and unsupervised transfer learning [14,27] depending on whether labeled samples are available. For unsupervised transfer learning problems, Dai et al. [14] proposed self-taught clustering (STC) to cluster a small collection of target data with the help of a large amount of unlabeled auxiliary data. STC extends the information theoretic co-clustering algorithm (ITCC) [28] with the assumption that target dataset and auxiliary dataset share the same feature clustering. STC minimizes the same loss with ITCC for the two datasets simultaneously. In this paper, we propose a method based on a similar assumption. However, unlike STC which is built on information theory, our method is built on graphs.

2.2. Multi-task clustering

Multi-task learning [29–31] performs multiple learning tasks concurrently to improve the individual performance. How to transfer useful knowledge among the tasks are the key to improve the performance. Most of the existing works focused on supervised settings and works on multi-task learning for unsupervised learning is relatively rare. Multi-task clustering can be easily achieved by performing multi-task dimensionality reduction [32,33] and applying k-means on the resulting embeddings. Since the embeddings are obtained by exploring relationships between tasks, these methods are usually better than performing k-means on the origin features individually. The key to the success of multi-task clustering is to transfer useful information among tasks without supervision information. We summarize the works on multi-task clustering as follows.

In some works, *e.g.* [34–37], the knowledge transfer for clustering is realized by borrowing proper instances from other tasks to form better k-nn graphs or distance metrics. But for most works on multi-task clustering, the process of knowledge transfer is achieved by modeling and employing the task relationships. In [38], a subspace is learned for each individual task under the constraint that the subspaces for related tasks are similar. Hence, the task relationships are captured by subspace distances. Since the centroids of clusters are representative, the centroids are easily employed to capture the relationships among tasks. In [39–41], the proposed algorithms learn better spaces, on which the centroids of clusters of different tasks closer to each other. In [11,42], multi-task Bregman clustering (MBC) was proposed, which is an extension of Bregman divergence based clustering [4]. The task relationships are captured by matching centroids. MBC aims at minimizing a local loss function for each single task and carries out a task regularization involving all tasks. The task regularization of MBC is indeed the sum of divergence between two learned density models for any pair of different tasks. It does not always boost the performance as observed in [43], and a smart multi-task Bregman clustering (SMBC) [43] was proposed as a remedy. SMBC utilize a local loss of each task to measure whether the negative effect occurs. If the local loss of one task calculated by MBC is larger than the single-task MBC, SMBC will stop using the boosting. In [44], an affinity matrix for all tasks is constructed and the balance between inter-task and intra-task is achieved by introducing the multi-task coefficient. The relationships among tasks are modeled by the affinity matrix constructed. However, the method proposed in [44] can only deal with binary clustering problem. In [45], the task relationships is modeled by the task covariance matrix, which is an extension of multi-task relationship learning [46] in the unsupervised setting. In [12], a method called multi-task spectral clustering (MTSC) extended spectral clustering to multi-task setting. MTSC considers intertask and intratask correlations. A novel $\ell_{2,p}$-norm regularizer was incorporated to control the coherence of all the tasks based on the assumption that related tasks share a common low-dimensional representation. And for each individual task, a mapping function is learned for predicting cluster labels. It has been shown that MTSC can incorporate discriminant information to further improve clustering performance and MTSC outperforms a number of state-of-the-art clustering approaches. In [47], multiple views are also used to boost the clustering performance. The view relationships are modeled by the agreement between the views of the same task, and the task relationships are captured by the common subspace on which the distributions of related tasks are close to each other. In [48], multi-task clustering was applied on videos and the task relationship was modeled by the distances among clusters. In [49], a method based on information bottleneck was proposed and the task relatedness is quantified by the mutual information of clusters between different tasks.

2.3. Multi-view clustering

Transferring knowledge from other views is another way to boost the performance. The method proposed in [50] is the first multi-view clustering algorithm. The method in [51] is based on canonical correlation analysis, and it explores basis vectors for two sets of variables by mutually maximizing the correlations between the projections onto these basis vectors. The model proposed in [52] integrates all features and learns the weight for every feature with respect to each cluster individually via new joint structured sparsity-inducing norms. Multi-view learning has also been combined with spectral clustering. In [53], the problem of two-view clustering is addressed by constructing a bipartite graph from nodes of both views, with the edges connecting nodes from one view to nodes in the other view. Such a bipartite graph can