Applied Soft Computing 47 (2016) 304–315

Contents lists available at ScienceDirect

Applied Soft Computing

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

Multi-view clustering via simultaneous weighting on views and features

Bo Jiang[a,*], Feiyue Qiu[a], Liping Wang[b]

[a] College of Education Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China
[b] College of Business and Administration, Zhejiang University of Technology, Hangzhou 310023, China

ARTICLE INFO

Article history:
Received 22 July 2015
Received in revised form 19 May 2016
Accepted 11 June 2016
Available online 17 June 2016

Keywords:
Multi-view clustering
Feature weighting
View weighting

ABSTRACT

In big data era, more and more data are collected from multiple views, each of which reflect distinct perspectives of the data. Many multi-view data are accompanied by incompatible views and high dimension, both of which bring challenges for multi-view clustering. This paper proposes a strategy of simultaneous weighting on view and feature to discriminate their importance. Each feature of multi-view data is given bi-level weights to express its importance in feature level and view level, respectively. Furthermore, we implements the proposed weighting method in the classical *k*-means algorithm to conduct multi-view clustering task. An efficient gradient-based optimization algorithm is embedded into *k*-means algorithm to compute the bi-level weights automatically. Also, the convergence of the proposed weight updating method is proved by theoretical analysis. In experimental evaluation, synthetic datasets with varied noise and missing-value are created to investigate the robustness of the proposed approach. Then, the proposed approach is also compared with five state-of-the-art algorithms on three real-world datasets. The experiments show that the proposed method compares very favourably against the other methods.

1. Introduction

Multi-view clustering is concerned with the problems of clustering from data represented by multiple distinct feature groups. Each different feature groups is a specific *view* of the data. For instance, to understand complex images, we often extract both the color and texture view of images [1]. A disease subtype may be recognized using both clinical symptoms (view 1) and genomic data (view 2) [2]. In most cases, different views provide compatible and complementary information that help us understand data more deeply, so it is natural to utilize all available views to obtain better clustering performance rather than relying on a single view [3].

However, different views are often derived from different types of measurements, they have very distinct statistical properties and produce different partitions. To obtain a consensus partition, the first challenge of multi-view clustering is how to discriminate different views in clustering algorithm. Surprisingly, most existing methods rely equally on every view, something may lead to performance degradation in the case of irrelevant views [4–9]. For instance, all the popular co-training based methods are based on the assumption that the true underlying clustering would assign corresponding points in each view to the same cluster [8,10]. Observing the drawbacks of this assumption, many recent work proposed to automatically assign a positive weight to each view express its importance [1,11–16]. For example, Tzortzis et al. [12] applied a kernel matrix to each view and then assigned a weight for each kernel; Liu et al. [13] assigned a weight for each view in the non-negative matrix factorization(NMF) based multi-view clustering; Cai et al. [14] improved the performance of *k*-means on multi-view big data also by view weighting.

Although these newly-proposed methods performed very well, they often only capture view-level importance and ignore the feature-level relationship. Firstly, the integration of multiple representations brings more features than single-view. Take the well-known UCI handwritten digits dataset as example,[1] 6 diversified views cause overall 649 features for each object. In such high dimensional spaces, it is highly likely there exist some features on which objects are distant from each other, so it may not be effective to simply take all features into account with equal contributions. Meanwhile, multi-view data are often collected from multiple sources, some views may contain noisy features. In other

* Corresponding author.
E-mail addresses: bjiang@zjut.edu.cn (B. Jiang), qfy@zjut.edu.cn (F. Qiu), wlp@zjut.edu.cn (L. Wang).

[1] http://archive.ics.uci.edu/ml/datasets/Multiple+Features.

http://dx.doi.org/10.1016/j.asoc.2016.06.010
1568-4946/

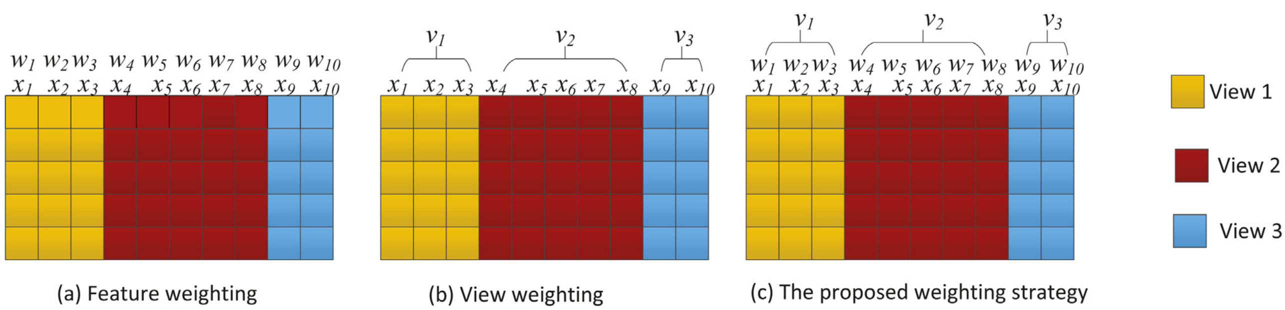

Fig. 1. Illustration of different weighting methods on multi-view data. A three-view dataset with 5 objects and 10 features is used. The features with the same color belong to the same view. (a) Feature weighting method assign each feature x_i a weight w_i; (b) View weighting method assign the same weight for features in the same view; (c) Our method assign each feature bi-level weights.

words, within a specific view, the contribution of different features should also be discriminated to reduce the influence of noise. A popular feature selection method for single-view data is feature weighting, in which each feature is assigned a positive weight to represent its importance [17–26]. However, these feature weighting methods only compute weights for individual features and ignore the contributions of different views. Therefore, they are not suitable for multi-view data.

In this work, we propose a new multi-view clustering framework that simultaneously weighting views and features. Fig. 1 illustrates a simple example to demonstrate the difference between the proposed method and existing weighting methods. Fig. 1 takes a dataset with five objects and ten features as an example. The feature weighting method assigns each feature x_i a weight w_i(Fig. 1(a)). The view weighting method gives each view a weight and the features within the same view i have the same weight v_i(Fig. 1(b)). In contrast, in the proposed method, each feature x_i has bi-level weights w_i and v_i (Fig. 1(c)), the former is used to represent the importance of individual feature and the latter is used to express the difference of views. The key advantage of the proposed weighting strategy is that, it can discriminate views and features simultaneously, and so they are usually more flexible and robust than other methods when dealing with noisy multi-view data.

To implement the idea of simultaneous weighting, we embed it into k-means clustering algorithm to handle multi-view data. Then, we propose an efficient algorithm based on gradient to search the clustering centroids and update the two weight vectors automatically. Furthermore, we theoretically prove the convergence of the updating of the two weight vectors. Finally, we test our method on a number of synthetic and real multi-view datasets and against state-of-the art approaches from literature. The experiments show that the proposed method compares very favourably against the other methods.

The rest of this paper is organized as follows. Section 2 reviews the related work of multi-view clustering and feature weighting clustering. The details of the proposed algorithm and its analysis are given in Sections 3 and 4, respectively. The performance evaluation results are shown in Section 5. Finally, Section 6 concludes this paper.

2. Related work

Feature weighting approach has received much attention because of its competitive clustering quality and straightforward interpretability [17–26]. More recently, the idea of feature weighting has been extended to discriminate the importance of view in multi-view data, and produced some view weighting clustering approaches, such as the weighted multi-view convex mixture model [11], multi-view kernel k-means algorithm [12], two-level weighting k-means algorithm [16] and so on. Compared with most of the state-of-the-art multi-view clustering algorithms, the approaches of view weighting demonstrate competitive clustering quality, good flexibility and strong robustness. However, existing approaches remain open for further improvement. Just as pointed out in [15,16], most of the existing weighting-based approaches only consider the view weights and ignores the difference among features.

2.1. Multi-view clustering

A range of techniques have been proposed for clustering data with multiple views. An earlier work by Bickel and Scheffer [4] presented a two-view oriented spherical k-means and EM algorithms, where the optimization processes are interleaved on different views. de Sa [5] proposed a two-view spectral clustering approach that aims to minimize the disagreement between the two views. Kumar and Daumé [8] proposed a multi-view spectral clustering algorithm that extends idea of co-regulation to multi-view environment to make the clusterings in different views agree with each other. Abhishek Kumar [9] adopted a co-training framework such that the similarity matrix in one view is affected by the similarity estimated based on the eigenvectors of Laplacian matrix in the other view. All of these pieces of work relies on two fundamental assumptions, which are all views are compatible to each other and all views are equally important. Zhou and Burges [27] developed a multi-view spectral clustering method via generalizing the normalized cut from a single view to multiple views. In this approach, the relative importance of each view is determined by a manually-specified parameter. More recently, Liu et al. [13] proposed a non-negative matrix factorization based multi-view clustering algorithm, which formulates a joint matrix factorization process with the constraint that pushes clustering solution of each view towards a common consensus. Similar with [27], this approach also needs user to provide the relative weight of each view prior.

To tackle this problem, Wang et al. [3] formulated the two-view spectral clustering as a two-objective optimization problem and then solved the induced problem using Pareto optimization. The Pareto optimization can provide several diversified Pareto optima solutions that reflect different view preferences, so user doesn't need to specify the weights for different views. Theoretically, this framework can be extended to cluster data with multiple views, but the inefficiency of Pareto optimization in high-dimensional objective space leads its poor scalability to the number of views. Tzortzis and Likas [11] presented a multi-view clustering algorithm based on a weighted multi-view convex mixture model that associates a weight with each view and learns these weight automatically. Recently, they extended the idea of automatic view weighting to kernel k-means and designed a kernel-based weighted multi-view algorithm, namely MVKKM, where each view is expressed in terms of given kernel matrix and a dynamical weight vector is used to combine these kernel matrices [12].