Neurocomputing 153 (2015) 271–277

Contents lists available at ScienceDirect

Neurocomputing

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

Combinative hypergraph learning for semi-supervised image classification

Binghui Wei [a,b], Ming Cheng [a,*], Cheng Wang [a], Jonathan Li [a,c]

[a] Computer Science Department, School of Information Science and Engineering, Xiamen University, Xiamen, 361005, China
[b] College of Applied Science, Jiangxi University of Science and Technology, Ganzhou, 341000, China
[c] Faculty of Environment, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

ARTICLE INFO

Article history:
Received 4 August 2014
Received in revised form
30 September 2014
Accepted 13 November 2014
Communicated by M. Wang
Available online 24 November 2014

Keywords:
Image classification
Transductive learning
Hypergraph learning
spaRse representation

ABSTRACT

Recent years have witnessed a surge of interest in hypergraph-based transductive image classification. Hypergraph-based transductive learning models the high-order relationship of samples by using a hyperedge to link multiple samples. In order to extend the high-order relationship of samples, we incorporate linear correlation of sparse representation to hypergraph learning framework to improve learning performance. In this paper, we present a new transductive learning method called combinative hypergraph learning (CHL). CHL captures the similarity between two samples in the same category by adding sparse hypergraph learning to conventional hypergraph learning. And more, we propose two strategies to combine the two hypergraph learning methods. Experimental results on two image datasets have demonstrated the effectiveness of CHL in comparison to the state-of-the-art methods and shown that our proposed method is promising.

1. Introduction

Currently, there is widespread interest in the development of image classification using transductive learning. Because transductive learning explores not only labeled data but also unlabeled data, it achieves a performance better than the methods that learn about classifiers based only on labeled data. Its success is based on one of the following two assumptions: cluster and manifold assumptions. The cluster assumption supposes that the decision boundary should not cross high-density regions, whereas the manifold assumption means that each class lies on an independent manifold. Therefore, more and more researchers have been devoted to transductive learning based on these assumptions in recent years [1–36]. Rosenberg et al. proposed self-training to train object detection systems [8]. Blum and Mitchell used co-training to classify web pages and provided empirical results which achieved significant improvement of hypotheses on real web-page data in practice [9]. Joachims introduced transductive support vector machines (TSVMs) for text classification [10]. Another important transductive learning method is graph-based learning, which is the derived form of hypergraph-based learning on which we have focused in this paper.

The graph-based learning [15–41] achieves a promising performance between the existing transductive learning methods. Its development goes through two stages: simple-graph learning and hypergraph learning. This type of learning is built on a graph, in which vertices are samples and edge weights indicate the similarity between two samples. However, the simple-graph learning methods consider only the pairwise relationship between two samples, and they ignore the relationship in a higher order. Hypergraph learning aims to get the relationship between several samples in a higher order. Unlike a simple graph that has an edge between two vertices, a set of vertices is connected by a hyperedge in a hypergraph, and each hyperedge is assigned a weight. Hypergraph learning derives from simple-graph learning, and thus it achieves a promising performance in many applications. For example, Agarwal et al. utilized hypergraph to clustering by using clique average [14]. Zass and Shashua adopted hypergraph in image matching by using convex optimization [15]. Sun et al. utilized hypergraph to problems of multi-label learning [16]. Huang et al. applied hypergraph cut to video segmentation [17]. Tian et al. proposed hypergraph-based learning algorithm to classify gene expression data by using biological knowledge as a constraint [18]. Huang et al. formulated the task of image clustering as a problem of hypergraph partition [52]. A hypergraph-based image retrieval approach is proposed in [19]. This approach constructs a hypergraph by generating a hyperedge from each sample and its neighbors, and hypergraph-based ranking is then performed. Wong and Lu proposed hypergraph-based 3-D

* Corresponding author.
E-mail address: chm99@xmu.edu.cn (M. Cheng).

http://dx.doi.org/10.1016/j.neucom.2014.11.028
0925-2312/

object recognition [20]. Bu et al. [21] developed music recommendation by modeling the relationship of different entities through a hypergraph to include music, tag, and users.

In hypergraph learning, the weights of the hyperedges are empirically set according to certain rules. Zhou et al. connected k neighbor points of a given point as a hyperedge [37]. Yu et al. proposed a hypergraph learning method by adaptively coordinating the weights of the hyperedge [38]. In essence, they chose multiple neighborhoods to construct multiple hyperedges for a given point. By this way, how to define a hyperedge and the weight of a hyperedge is the succeeding problem we should take into account. There are mainly two methods to select the neighborhood of a given point, one is to define a hyperedge with fixed number neighbor points, and the other one is to define a ball whose radius is less than some threshold. However, these hypergraph-based methods focus only on proximity relation for distance. There is some other high-order relationship, such as linear relationship. For example, given two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ for existing $\boldsymbol{a}=k^*\boldsymbol{b}$, the distance between $\boldsymbol{a}$ and $\boldsymbol{b}$ may be very large, but, they are similar for one factor difference based on linear relationship. Therefore, the linear relationship is another one we should take into account.

This paper takes into account the clustering assumption that the similar points in feature space more likely belong to the same category. Then, we define the similarity by two assumptions that we call as distance-similarity and linear-similarity. Distance-similarity is that the points derived from a category are located close to each other. The neighborhood size of the hypergraph based on this assumption is chosen as a fixed number. Linear-similarity is that a data point can more likely be represented linearly by the data points nearby which belong to the same category as this data point. It is analogous to manifold assumptions aforementioned. The neighborhood size of the hypergraph based on this assumption is unfixed since it is decided by the sparse representation method. Inspired by the two assumptions, we construct the following two kinds of hypergraphs on a data set, one is conventional hypergraph which is based on the distance-similarity, the other one is sparse hypergraph which is based on linear-similarity and derived from the thinking in Refs. [53,54]. We linearly combine the two hypergraph learning methods by two strategies. The first strategy is to combine the hyperedges of the two hypergraphs to form a new hyperedge set. The second strategy is to linearly combine the confidence of the labeling of the two hypergraph learning methods to form a new confidence of the labeling to define the label of data points.

The contributions of this paper are as follows:

1) A novel algorithm named Combinative Hypergraph Learning (CHL) is proposed for image classification. To consider the high-order information, we incorporate sparse representation into the standard hypergraph learning framework. Hence, our algorithm achieves a better performance than that with only conventional hypergraph learning or with only sparse hypergraph learning.
2) We provide two strategies to combine linearly conventional hypergraph learning and sparse hypergraph learning. The essence is that we combine linearly the results of the two hypergraph learning methods with same weights for one strategy and with different weights for the other strategy.
3) We conduct comprehensive experiments to empirically analyze our algorithm on two image databases. The experimental results demonstrate that our algorithm outperforms other methods including Transductive SVM [10], Simple Graph-Based Learning [50], and Semi-supervised Discriminant Analysis (SDA) [42,51] classifications.

The rest of this paper is organized as follows. Section II describes the proposed image classification by combinative hypergraph learning. Section III shows experiments on practical image datasets. Section IV concludes the paper.

2. Hypergraph learning

This section introduces conventional hypergraph learning and sparse hypergraph learning first, and shows combinative hypergraph learning theory with two strategies in the following text. In Table 1, we provide important notations used in the rest of this paper to present the technique details of the proposed method. Fig. 1 illustrates the whole framework of our method.

2.1. Conventional hypergraph learning

Given c categories of images including m training data points $(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_m, \boldsymbol{y}_m)$, and n testing data points $(\boldsymbol{x}_{m+1}, \mathbf{0}), \ldots, (\boldsymbol{x}_{m+n}, \mathbf{0})$, where $\boldsymbol{x}_i \in \mathrm{R}^d$, $1 \le i \le m+n$ is sampled from the input space; $\boldsymbol{y}_i = [0, \ldots, 1, \ldots, 0]^{\mathrm{T}} \in R^c$, $1 \le i \le m$, is the label vector of $\boldsymbol{x}_i$, where the g-th component is 1 if $\boldsymbol{x}_i$ belongs to the g-th category, otherwise, 0; and $\mathbf{0}$ is a vector with c components of zero.

Table 1
Important notations used in this paper.

Notations	Descriptions
$\mathbf{H}=(\mathbf{x}, \varepsilon)$	The representation of a hypergraph, where $\mathbf{x}$ and ε indicate the sets of vertices and hyperedges, respectively
$\mathbf{A}$	The incidence matrix for the hypergraph
dist	The distance between two samples
$\boldsymbol{\varphi}$	The diagonal matrix of the vertex degrees
$\boldsymbol{\gamma}$	The diagonal matrix of the hyperedge degrees
$\boldsymbol{\omega}$	The diagonal weight matrix and its (i,i)-th element is the weight of the i-th hyperedge
$\mathbf{L}$	The constructed hypergraph Laplacian matrix
$\boldsymbol{y}_{\mathrm{i}}$	The label vector for i-th class. Its j-th element is 1 if the j-th object belongs to the i-th class, and otherwise it is 0.
$\mathbf{F}=[f_1, f_2,\ldots f_c]$	$\mathbf{F}$ represents the relevance score matrix for all samples, and f_i is the to-be-learnt relevance score vector for class i.
c	The number of classes in the dataset
m	The number of labeled images in the dataset
n	The number of images in the dataset
l	The number of hyperedges
$\mathbf{A}^{sp}$, dist^{sp}, $\mathbf{F}^{sp}$	The superscript “sp” denotes sparse-based which show difference to $\mathbf{A}$, dist and $\mathbf{F}$, respectively
$\mathbf{A}^*$, $\mathbf{F}^*$	The superscript “*” denotes the concatenated results.
$\sum=[\varsigma_1, \varsigma_2, \ldots \varsigma_d]$	The base of sparse representation
$w=[w_1, w_2, \ldots w_d]^T$	The coefficient vector of sparse representation
$\mathbf{W}=[\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots \boldsymbol{w}_n]$	The coefficient matrix of sparse representation