Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

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

Fast depth-based subgraph kernels for unattributed graphs

Lu Bai [a,*], Edwin R. Hancock [b]

[a] School of Information, Central University of Finance and Economics, Beijing, China
[b] Department of Computer Science, University of York, York, UK

ARTICLE INFO

Article history:
Received 28 July 2014
Received in revised form
17 June 2015
Accepted 7 August 2015

Keywords:
Depth-based representations
Entropy
Graph kernels
The Jensen–Shannon divergence
Graph isomorphism tests

ABSTRACT

In this paper, we investigate two fast subgraph kernels based on a depth-based representation of graph-structure. Both methods gauge depth information through a family of K-layer expansion subgraphs rooted at a vertex [1]. The first method commences by computing a centroid-based complexity trace for each graph, using a depth-based representation rooted at the centroid vertex that has minimum shortest path length variance to the remaining vertices [2]. This subgraph kernel is computed by measuring the Jensen–Shannon divergence between centroid-based complexity entropy traces. The second method, on the other hand, computes a depth-based representation around each vertex in turn. The corresponding subgraph kernel is computed using isomorphisms tests to compare the depth-based representation rooted at each vertex in turn. For graphs with n vertices, the time complexities for the two new kernels are $O(n^2)$ and $O(n^3)$, in contrast to $O(n^6)$ for the classic Gärtner graph kernel [3]. Key to achieving this efficiency is that we compute the required Shannon entropy of the random walk for our kernels with $O(n^2)$ operations. This computational strategy enables our subgraph kernels to easily scale up to graphs of reasonably large sizes and thus overcome the size limits arising in state-of-the-art graph kernels. Experiments on standard bioinformatics and computer vision graph datasets demonstrate the effectiveness and efficiency of our new subgraph kernels.

1. Introduction

There has recently been an increasing interest in learning and mining data using graph structures. Application includes (a) view-based object recognition [4], (b) bioinformatics [5,6] (e.g., classifying proteins into different families, classifying tissue samples), and (c) social networks (e.g., classifying users based on their feeds on Twitter, Facebook, etc.). One challenge arising in classifying graphs is how to convert the discrete graph structures into numeric features or efficiently compute similarities between graphs for classification. One way to address this problem is to use graph kernels.

1.1. Graph kernels

Graph kernels can characterize graph features in an explicit high dimensional space and thus have the capability of preserving graph structures. A number of graph kernels have been defined in the literature. Generally speaking, most existing graph kernels are usually formulated in terms of instances of the R-convolution kernel family developed by Haussler [5]. R-convolution is a generic way for defining graph kernels based on comparing all pairs of decomposed subgraphs. Specifically, all available graph decompositions can be used to define a kernel, e.g., the graph kernel based on comparing all pairs of decomposed (a) walks, (b) paths and (c) restricted subgraph or subtree structures. With this scenario, Kashima et al. [7] have proposed a random walk kernel by comparing pairs of isomorphic random walks in a pair of graphs. The main drawback of the random walk kernel is the notorious tottering problem. This occurs when a random walk on a graph moves in one direction and then immediately returns to the starting position through the same vertices and edges possibly multiple times. To overcome this shortcoming, Borgwardt et al. [8] have proposed a shortest path kernel by counting the numbers of pairwise shortest paths having the same length in a pair of graphs. Aziz et al. [9] have defined a backtrackless kernel using the cycles identified by the Ihara zeta function [10] in a pair of graphs. The method overcomes the tottering problem using backtrackless substructures, i.e., the shortest paths or cycles in graphs. Unfortunately, shortest paths and cycles are structurally simple, and reflect limited topology information. Moreover, the computational efficiency of the two kernels also tends to be burdensome for graphs of large sizes, e.g., a graph having more than one thousand vertices.

To address the problem of inefficiency, Shervashidze et al. [5] have developed a fast subtree kernel by comparing pairs of

* Corresponding author.
E-mail addresses: lu@cs.york.ac.uk, bailu69@hotmail.com, l.bai@cufe.edu.cn (L. Bai), erh@cs.york.ac.uk (E.R. Hancock).

http://dx.doi.org/10.1016/j.patcog.2015.08.006
0031-3203/

subtrees identified by the Weisfeiler–Lehman (WL) algorithm. Unfortunately, like the random walk kernel, the WL isomorphism based subtree kernel also suffers from tottering. This is because the subtrees identified by the WL algorithm may also include several copies of the same pairwise vertices connected by the same edge. Furthermore, Costa and Grave [11] have defined a neighborhood subgraph pairwise distance kernel by counting the number of pairwise isomorphic neighborhood subgraphs. Both the WL subtree and neighborhood subgraph kernels can be computed in polynomial time. Some alternative graph kernels that specifically form the R-convolution framework include (a) the segmentation graph kernel developed by Harchaoui and Bach [12], (b) the point cloud kernel developed by Bach [13], (c) the subgraph matching kernel developed by Kriege and Mutzel [14], and (d) the (hyper)graph kernel based on directed subtree isomorphism tests developed and described in our previous work [15]. Moreover, it is important to note that, some of the aforementioned R-convolution kernels can accommodate attributed graphs too (i.e., these kernels can accommodate the attributed information residing on the vertices or edges). They can thus capture more characteristics that encapsulate label information on the vertices and edges [14]. Examples include the WL subtree kernel [5], the shortest path kernel [8], the random walk kernel [7], the subgraph matching kernel [14], and the (hyper)graph kernel [15].

One significant drawback of R-convolution kernels is that they compromise to use substructures of limited size, which only roughly capture topological arrangements of a graph. Though this strategy avoids the notorious inefficiency of R-convolution kernels when using large substructures, the limited size can only reflect restricted topological characteristics of a graph. Moreover, some R-convolution kernels still require significant computational overheads for large graphs (e.g., graphs having thousands of vertices).

An alternative way to construct a kernel is to measure the mutual information between pairs of graphs using the classical Jensen–Shannon divergence. In probability theory, the Jensen–Shannon divergence is a dissimilarity measure between probability distributions in terms of the nonextensive entropy difference associated with the probability distributions [16]. It is not only symmetric but also always well defined and bounded. In our previous work [4], we have used the classical Jensen–Shannon divergence to define a Jensen–Shannon kernel for graphs. Here, the Jensen–Shannon divergence between a pair of graphs is defined in terms of the entropy difference between the entropy of a composite graph structure and that of the individual graphs. Unlike the R-convolution kernels, the entropy associated with a probability distribution of an individual graph can be computed without decomposing the graph into substructures. Therefore, the computation of the Jensen–Shannon graph kernel between a pair of graphs avoids burdensome (dis)similarity measurements involved in comparing all substructure pairs. Unfortunately, the existing Jensen–Shannon graph kernel can only capture the global similarity between a pair of graphs, and cannot distinguish the basis of the interior topological information. Furthermore, the required entropy that must be calculated for the composition of a pair of graphs is obtained from the product graph. The vertex number of the product graph is the multiple of the vertex numbers of the pair of graphs being compared. As a result, the entropy difference is dominated by that of the product graph when the graphs being compared are large.

To overcome the shortcomings of existing graph kernels, in this paper we aim to develop novel and fast subgraph kernels. Our new kernels are based on a rapidly computed depth-based graph representation.

1.2. Depth-based representations

Depth-based representations have been widely used for characterizing undirected graphs [17]. One approach to computing a depth-based representation for a graph is based on an information content flow through a family of K-layer expansion subgraphs [1]. These subgraphs can be located from a vertex and have a maximum topology distance K from the vertex to the remaining vertices. Following this approach, Escolano et al. [1] have shown how to compute the thermodynamic based depth complexity for a graph. This is done by measuring the heat flow complexities of expansion subgraphs around the vertices of the graph. Unfortunately, the heat flow complexity measure for a (sub)graph having n vertices requires time complexity $O(n^5)$. As a result, the thermodynamic depth complexity measure cannot be efficiently computed. To overcome this shortcoming, Bai and Hancock [2,18,19] have developed a centroid-based complexity trace from a centroid vertex that has the minimum variance of shortest path lengths to the remaining vertices. This depth-based representation is computed around the centroid vertex, and decomposes a graph into a family of K-layer centroid expansion subgraphs that has a greatest shortest path length K rooted from the centroid vertex. The resulting complexity trace vector is computed by measuring the entropies of the expansion subgraphs. The centroid based method can be computed efficiently. The reason for this is that the entropy based complexity measures are computed on a small set of expansion subgraphs rooted at the centroid vertex, and can be computed in polynomial time.

Unfortunately, the centroid-based complexity trace may generate information loss for a graph structure. This is because the complexity trace vector of a graph can be viewed as an embedding vector, embedding a graph into a vector tends to approximate the structural correlations into a low dimensional space. One way to overcome the problem is to kernelize the embedding vectors (i.e., the complexity trace vectors) of graphs as a kernel function that represents graph structure in a high dimensional space and thus better preserves graph structure. Furthermore, since the centroid vertex is identified through a global analysis of the shortest path length distribution, the centroid expansion subgraphs provide a fine representation of graph structure. As a result, the centroid-based complexity trace and its required centroid expansion subgraphs offer us a potential way of defining a subgraph kernel. Unfortunately, the subgraphs of increasing layer size K tend to be the global graph (i.e., the largest layer subgraph is the graph itself), and straightforwardly measuring the (dis)similarity between whole graphs usually requires burdensome computations.

1.3. Contributions

The aim of this paper is to develop fast subgraph kernels that can not only be efficiently computed for large graphs but can also capture rich topological arrangement information contained within graphs. To this end, we investigate how to kernelize a depth-based representation of graphs. The contributions of this paper are twofold.

First, we develop a new depth-based subgraph kernel, namely the Jensen–Shannon subgraph kernel. This is done by measuring the Jensen–Shannon divergence between depth-based representations rooted at the centroid vertices [2]. To this end, we commence by computing the centroid-based complexity trace developed in our previous work and described in [2,18,19]. The advantage of using the complexity trace to characterize graphs is that it not only reflects dominant depth complexity information around the centroid vertex for a graph but also represents the graph in a high dimensional space. This is because the centroid-based complexity trace for a graph encapsulates information flow from the centroid vertex to the global graph using entropy measures. By contrast, existing entropy measures [21–23] or the depth complexity measures [17,1] only provide us with an uni-valued complexity measure for a graph. They thus reflect limited graph characteristics. With a pair of graphs and their centroid-

Please cite this article as: L. Bai, E.R. Hancock, Fast depth-based subgraph kernels for unattributed graphs, Pattern Recognition (2015), http://dx.doi.org/10.1016/j.patcog.2015.08.006