Mining Graph Structures Preserved Long Period

Takeaki Uno[1] and Yushi Uno[2]

[1] National Institute of Informatics, Japan. uno@nii.jp
[2] Graduate School of Science, Osaka Prefecture University, Japan.
uno@mi.s.osakafu-u.ac.jp

Abstract. Data mining from sequences of graphs is now increasing its importance in practice. In this research, we newly focus on "preserving structures" while other studies have focused on "changes". We particularly address connected induced subgraphs and cliques that do not change in a long period in the given graph sequence. We propose new polynomial delay algorithms for the problems. The delay is $O(|V||E|^3)$ for the former, and and $O(\min\{\Delta^5, |E|^2\Delta\})$ for the latter, where the input graph is $G = (V, E)$ with maximum degree Δ.

Introduction

In a recent and practical situation, graph structures may change over time (i.e., "dynamic"), and such data is collected periodically along a time series (and thus the data becomes bigger). In this setting, not only information acquired separately from single graphs but also from graph patterns appearing sequentially could be important. Along this direction, there are some research topics of interest so far. Finding graph patterns that appear periodically in a graph sequence is studied. Graph patterns frequently appear during a certain period such as burst patterns are also studied. On the other hand, some research address the change patterns that appear frequently in a graph sequence composed of graphs with edge insertions/deletions, such as changes between two time periods and changes of subsequences. Several studies focus on clustering of vertices. Surprisingly, all these researches look at "changes", but no research does "stability".

We propose a new concept of graph mining; finding graph/subgraph structures belonging to a graph structure class, that are appearing in a long period in a series of dynamically changing graphs. We call such structures *preserving structures* in a graph sequence, and the problem for enumerating all such structures *preserving structure mining* in general. As for such properties, we consider maximal connected induced subgraphs and maximal cliques. For example, a topic on the Web that is controversial for a long time may correspond to a clique that exists in a consecutive sequence of webgraphs during a certain period. As another example, a group of a species in a wildlife environment may constitute a consecutive sequence of connected vertex subsets in a sequence of graphs that are constructed from its trajectory data. To the best of our knowledge, this study is the first case in which a "long-lasting" or "unchanging" structure is regarded as the target structure to be captured.

Related works

(1) Pattern mining in graph sequences. This is already explained in the Introduction, and it tends to capture changes.

(2) Dynamic flow. On a dynamic network defined by a graph with capacities and transit times along its edges, the dynamic flow problem asks the maximum flow from a specified source to a sink within a given time bound. As explained

later, our model for a graph sequence can be naturally generalized so that it implies dynamic flows.

(3) Dynamic graph algorithms. Dynamic graph problems of constructing data structures that enables to answer a given graph property quickly, with small update cost for edge insertions/deletions. Typical properties of concern include connectivity, transitive closures, cliques, bipartiteness, shortest path distance, and so on. Dynamic graph algorithms could also find a period during which a property is satisfied. However, since they are not well designed so that they can extract local structures efficiently, they are not suitable for this purpose.

Contributions

In this paper, we first propose a new concept, that is, a preserving structure in a graph sequence. Then by adopting this notion, as an onset, we pose two problems of mining preserving structures: one for maximal cliques and the other for maximal connected induced subgraphs. As we have seen so far, both structures or properties will have significant meanings in a sequence of graphs that appear in practical situations.

Let $G_1, \ldots, G_T$ be the graph sequence we are given. For a vertex set S, the *active time set* is the set of indexes of the graphs of the graph sequence in which S induces the connected graph. When there is an interval of the active time set is of length at least τ, we say that S is *preserving*. Our first problem is to enumerate all maximal preserving connected induced subgraphs. Our enumeration algorithm for the problem is based on a recursive graph partition. Consider the intersection of the partitions, given by the connected components of graphs $G_{t_1}, \ldots, G_{t_2}$, such that the result is composed of disjoint maximal vertex sets that are not subdivided by the connected components of any graph. Look at a component S of the resulted partition. If S is connected in any graph in the graphs, it is a solution to the problem. Otherwise, its subsets are solutions. Thus, we recursively do this for the set of graphs $G_i[S], i = t_1, \ldots, t_2$, recursively, until the components will be connected in any graph. This idea motivates us to find all solutions to short period, and update them with increase the time span. In this way, we can enumerate all solutions to the problem. When for each edge, the graphs having the edge form an interval, we have the following theorem.

Theorem 1. *All maximal preserving connected induced subgraphs can be enumerated in $O(|V||E|^3)$ time for each, where the input graph is $G = (V, E)$.*

Our algorithm for enumerating maximal preserving cliques is based on the reverse search, which is a framework for designing efficient enumeration algorithms. The idea is to introduce a parent-child relation.

While a straightforward application of maximal clique enumeration to our problem may require a long delay per output, our algorithm exploits properties of the time intervals of edges so that the algorithm will be polynomial delay. Compared to a naive algorithm, this reduces the time complexity with a factor of the number of edges of an input graph. Although these algorithms may seem to be relatively simple, our problem setting is quite fundamental and new. Therefore, it gives a new perspectives for graphs that change over time, together with a way of data representations and analysis of algorithms, and it would be a first step to pioneer a new research field.

Theorem 2. *All maximal preserving cliques can be enumerated in $O(\min\{\Delta^5, |E|^2\Delta\})$ time for each where the input graph is $G = (V, E)$ with maximum degree Δ.*