Information Sciences 310 (2015) 97–117

Contents lists available at ScienceDirect

Information Sciences

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

# Ordered subtree mining via transactional mapping using a structure-preserving tree database schema

Fedja Hadzic [a,b], Michael Hecker [a,b], Andrea Tagarelli [c,*]

[a] Dept. Computing, Curtin University, Perth, Australia
[b] Skrydata Pty Ltd, Perth, Australia
[c] DIMES, University of Calabria, Rende (CS), Italy

ARTICLE INFO

*Article history:*
Received 22 April 2014
Received in revised form 25 February 2015
Accepted 9 March 2015
Available online 19 March 2015

*Keywords:*
Frequent subtree mining
Position-constrained subtree discovery
Transactional representation
Semistructured data

ABSTRACT

Frequent subtree mining is a major research topic in knowledge discovery from tree-structured data, whose importance is witnessed by the pervasiveness of such data in several domains. In this paper, we present a novel approach to discover all the frequent ordered subtrees in a tree-structured database. A key aspect is that the structural aspects of the input tree instances are extracted to generate a transactional format that enables the application of standard itemset mining techniques. In this way, the expensive process of subtree enumeration is avoided, while subtrees can be reconstructed in a post-processing stage. As a result, more structurally complex tree data can be handled and much lower support thresholds can be used. In addition to discovering traditional subtrees, this is the first approach to frequent subtree mining that can discover position-constrained subtrees. Each node in the position-constrained subtree is annotated with its exact occurrence and level of embedding in the original database tree. Also, disconnected subtree associations can be represented via virtual connecting nodes. Experiments conducted on synthetic and real-world datasets confirm the expected advantages of our approach over competing methods in terms of efficiency, mining capabilities, and informativeness of the extracted patterns.



## 1. Introduction

The discovery of associations among the constituents of tree data objects has as a prerequisite the extraction of the sub-structures that occur frequently, given a user-specified support threshold. This is known as the *frequent subtree mining* (FSM) problem [16,9]. Intuitively, FSM tasks present additional challenges than the mining of frequent itemsets in transactional/relational data [19,41]. A general explanation is due to the semistructured nature of tree data, which enables the representation and description of real-life objects and their relationships. The variety of structure and content information in tree data allows us to devise various scenarios of data management and knowledge discovery, at different levels of complexity (e.g., it may be advisable to consider structure features alone, or in combination with content features). Any FSM method needs to take into account various characteristics of the input tree data (e.g., the depth, fan-out, and label set size of the tree instances) that impact on the amount and structures of the mined subtrees. Moreover, the existing FSM methods can be broadly categorized depending on their ability in discovering different types of frequent subtrees. These include ordered

* Corresponding author.
*E-mail addresses:* fedja.hadzic@curtin.edu.au (F. Hadzic), michael.hecker@curtin.edu.au (M. Hecker), tagarelli@dimes.unical.it (A. Tagarelli).

http://dx.doi.org/10.1016/j.ins.2015.03.015
0020-0255/

induced [2,27], ordered embedded [29,40], unordered induced [24], and unordered embedded subtrees [39]. All such variants have their respective motivations in order to comply with specific application requirements [16,9].

An important aspect that is usually ignored by traditional FSM methods is the node positional information that might be considered in the discovery of frequent subtrees. One reason for that is the increased burden of complexity that this type of information would introduce in the whole FSM approach. Moreover, the use of relatively low minimum-support thresholds (to increase the number of mined subtrees) would also negatively affect the performance of FSM methods, or even lead to computationally infeasible mining tasks. However, considering node positional information is essential to make the FSM task more expressive and meaningful in several applications scenarios. These would share the requirement that the mined subtrees should reflect different contexts of occurrence of frequent subtree associations.

To illustrate a motivating application scenario, consider Fig. 1 which shows a fragment of a process instance from the *HospitalLogs* dataset. (We have included this dataset in our experimental evaluation; cf. Section 4.2.) This dataset describes events that correspond to different patient cases, concerning information related to patient treatments, such as, e.g., when certain treatment activities took place, the group that performed the activity. Many treatment activities can repeat in the patient treatment life-cycle; as a consequence, in the dataset there are repeating attributes for a patient, indicating that a patient went through different/overlapping phases consisting of diagnosis/treatment combinations. With such properties, while the subtrees extracted under the traditional FSM framework could detect frequent treatment activities that occur across different cases, they would fall short in indicating exactly at which phase of the treatment the activities occur. In other words, there would be no indication of whether any other activities took place and in which particular phase. Ordered subtrees would capture the order of activities; however, in different process instances, many more activities could have occurred in between, and in others at completely different phases. Grouping subtree patterns regardless of where they have occurred in the process instance, could be considered incorrect in this domain, as the specific phase may indicate the context of the treatment; for example, certain activities should only occur after other activities are completed and any non-compliance to this should be known. Generally speaking, in cases when the users are investigating the conformance of business process instances to a business process model, it may be important to know when an action or sets of actions occur in a phase of the workflow other than expected by the model [36].

In this work, we propose an approach to FSM which has the unique property of distinguishing the mined subtrees by their node positional information, according to a schema of the structures of the input tree data. By imposing a *position-constrained subtree generation*, only subtrees where all nodes have the same label and node position will be grouped together. In our example, this will avoid the grouping of activities that occur in completely different contexts or phases of a patient treatment, and the subtrees themselves would be more informative. In fact, one challenge in FSM problems is related to complexity issues due to the repetition of nodes within a tree instance which are used to denote attributes of objects represented in the tree. For instance, in the example of Fig. 1, "AuditTrail" is such a node, which repeats many times and in every instance: by ignoring the positional information of each occurrence of the "AuditTrail" node, the number of the activity in the whole treatment phase that the occurrence indicates will be lost, causing the extraction of many large non-informative subtrees.

Note also that in the last few years, a number of significant achievements in complexity reduction and interestingness measures have been developed for transactional/relational and sequential data, based on data compression, pattern reduction and constraint definition [19]. Moreover, we have witnessed the development of mining techniques that have been adapted from itemset to sequence and, more recently, to tree data. Given the importance and popularity of pattern mining tasks in several application domains, one can expect that more well-established techniques will be explored in the future.

Within this view, a major goal of our study is to hasten the progress in research on FSM with a novel approach that aims to overcome the general difficulty of traditional FSM methods in leveraging the structural complexities of labeled tree data. Our key idea is to perform a task of itemset mining that is aware of the structure of the input tree data. We accomplish this by exploiting a unique characteristic of the proposed approach, which is a structure-preserving transactional mapping of the input tree data. More precisely, the scope of our approach applies within the FSM field where the ordering among the sibling nodes needs to be preserved, i.e., our focus is on *ordered subtree mining*. The proposed approach is comprised of four main stages that apply over any given input tree database:

Stage 1: extraction of a structure-preserving schema of the tree database, called Database Structure Model (DSM), which encompasses all structural characteristics of the input tree instances;
Stage 2: translation of the tree instances matched to the DSM into a transactional format;
Stage 3: mining of the frequent closed itemsets from the transactional representation of the trees;
Stage 4: translation of the enumerated frequent itemsets to subtrees by matching back to the DSM.

We can summarize the major features of our approach as follows:

- *Simplification of subtree pattern generation*: The transactional conversion of the tree database adopted by our approach allows the use of itemset mining techniques, thus making the FSM problem easier to be solved. Eventually, the extracted frequent itemsets can be converted into structurally valid subtrees at the most expressive level, that is, embedded