Study on A Rapid Export Subtree Mining Algorithm

Xin Guo [1]

[1] Key Laboratory of Eco-tourism in Hunan Province, Jishou University, Zhangjiajie, Hunan, 427000

hunter2011@foxmail.com

Keywords: Data Mining; Frequent Subtree; Export Subtree

Abstract. Export subtree mining has important research value in the field of XML data, bioinformatics, Web structure. In this paper, vector and sub-tree pruning threshold two concepts are taken and take advantage of the sub-tree vector features, combined with a hash table structure, we proposed a sub-tree based vectors quick export subtree mining algorithm-ITMSV (induced subtrees mining based on subtree vector). Vector algorithm based on sub-tree and hash table of data to build a multi-layered structure, in the mining process can reduce tree isomorphic discriminating time and requires only one database scanning operation, reduce scan times to improve the operation of the algorithm effectiveness. Experiments show that this algorithm is feasible and has a higher operating efficiency compared with other similar algorithm.

Introduction

Mining frequent patterns from past frequent item sets and gradually developed into a structured data, including frequent subtree mining and frequent subgraph mining. Compared sets and sequences, because the tree can better express the relationship between things, so mining frequent subtrees more cause for concern. Export subtree [1] as a tree structure, due to its structure as compared with other tree has all the characteristics of the nodes must be connected, and therefore have a high value in the field of bioinformatics, Web structure, XML data, etc. .

Basic Definition

Definitions 1. ordered labeled tree. Tree ordered labeled tree T = (V, E) is a root node has a label and a directed acyclic graph, where V represents a set of nodes in the tree, E = {(x, y) | x, y∈V} represents the set of edges in the tree, with L = {l0, l1, ..., ln} represents a reference set of nodes V and L there is a correspondence between the l: V → L, called node label. Ordered means that for each node in the tree, all of child nods built sibling relationships from left to right in the order.

Preorder same binary tree, first traversal of the tree refers to access to the root pointing from left to right and then turn preorder Meike subtree. Prior preorder process, each node has been accessed in the order form number (root is 0) is called preorder node number, referred to as the node number. Junction reference sequence prior order process, each node has been accessed in the order form is called the tree of string representation. String Figure 1 tree: ABA ## CC ## B

Definition 2. tree node relationship [8] Given a tree root is vo T, consider the path from the beginning of any vo: If u v in front, then u is the ancestor of v, v is of u sons, if u, v only between one edge, then u is the father of v, v is u child. This side is called the branch, denoted by b = (u, v). If some nodes have the same father, so these nodes are brothers.

3. Definition of Export subtree (induced subtree) [1] for tree S = (Vs, Es) and tree T = (V, E), if there is a one to one correspondence φ:. Vs → V, for any (x, y) Es, there are (φ (x), φ (y)) E and l (x) = l (φ (x)), called S for the exported subtree.

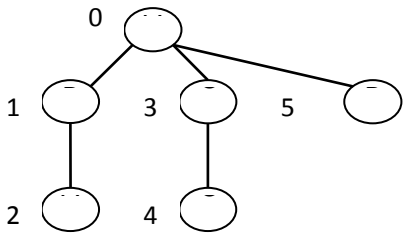


Fig.1 ordered label tree

Quick Export Subtree Vector-Based Mining Algorithm ITMSV

ITMS Algorithm Data Structure. On any node tree T X, with a triple SX = (fx, lx, μx) to uniquely represent a node, the node number where fx represents the parent node X, lx represents X number of nodes, μx represented by X node subtree rooted at the node number rightmost leaf node, if the node is the root of the tree, then fx = -1, called SX triples for the nodes represent.

With a four-tuple (Tid, B, M, S) to represent a sub-tree, which represents the sub-Tid tree in the forest where the tree number (database). B Indicates whether this subtree is exported subtree (1 for the exported subtree, 0 for non-export sub-tree), M represents primer tree prefix, S represents the rightmost leaf node of this subtree node in the tree representation of the original, if only one child node of the tree, then M = -1, said this quad tree representation for the child. In Figure 2, assuming that the serial number in the database tree T is 0 subtree S1 is available (0,0,02, (0,5,5)), said sub-trees S2 is available (0,1,01, (0 , 3,4)) to represent.

Export subtree candidate generation strategy. RME rightmost any two of the same sub-tree all prefixes, RMEx = (X, i, Sx, Vx) and RMEy = (Y, j, Sy, Vy) connection, divided into two steps, the first step pair (X, i) and (Y, j) connected to generate candidate exported subtree, denoted by (X, i) (Y, j) according to the following rules:

If (i = j), and is in the sub-tree generation candidate two sub-tree will be (Y, j + 1) [Scenario 1] is applied to (X, i) represented by generating a candidate exported subtree; if (i = j) and is not a candidate in the second sub-tree will be generated (Y, j) [Case 2] sub-tree and (Y, i + 1) [Case 3] is applied (X, i) represented by the generate two candidates export subtree.

If (i> j) then (Y, j) [Case 4] is applied (X, i) represented by the sub-tree, generate a candidate exported subtree.

If (i <j) are not generating a candidate exported subtree.

The second step in the merger of the rightmost sub-tree vector V pairwise merger, whereby the candidate exported subtree support, and then support can determine whether the frequent, if not often you can export this sub-tree from Kazakhstan Remove the table, reduce the search space. If set (X, i) the vector of any one of the sub-tree element L (X, i) = (tx, bx, mx, sx), (Y, j) subtree elements of the vector of any one of L (Y, j) = (ty, by, my, sy), the subtree of candidate vectors generated for the elements (tn, bn, mn, sn), the connecting operation denoted by L (X, i) L (Y, j), and according to the following rules:

Rule premise: tx = ty (described both belong to the same sub-tree derived tree) and mx = my (described with the same prefix derived two sub-trees)

If Sx⊃Sy and bx = 1, lx = fy then tn = tx, bn = 1, mn = {my ∩ lx}, sn = sy, or if Sx⊃Sy then tn = tx, bn = 0, mn = {my ∩ lx}, sn = sy. This situation corresponds to the case of the first step of 1 and 3.

If Sx <Sy and bx = by = 1, tn = tx, bn = 1, mn = {my ∩ lx}, sn = sy, or if Sx <Sy then tn = tx, bn = 0, mn = {my ∩ lx}, sn = sy. This corresponds to the first step in scenarios 2 and 4.

Note: If my = -1, then {-1 ∩ lx} = lx, is a candidate to generate two sub-tree situation.

If the nature of the sub-tree is not an export frequent subtrees, then it is not a superset of frequent subtree.

Proof: If an export support for subtree T1 s, less than a user-specified minimum support minsup, then the exported subtree T1 will not be frequent, namely P (T1) <minsup, if more add a node to this export sub-tree, the number of new export subtree T2 obtained appear in the entire database will not be more than the number of the original export subtrees T1 that appears, therefore, T2 are not frequent subtrees, namely P (T2) <minsup.

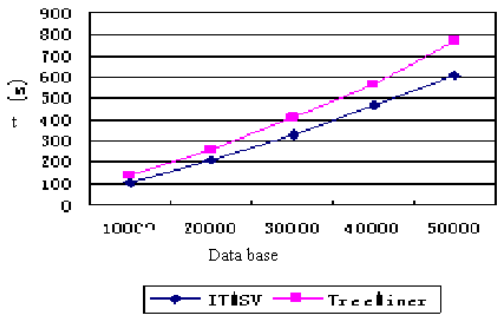
Before it can take advantage of the nature of a second-step merger to prune operation, if the candidate exported subtree all subtrees not frequent, then the second step without performing subtree merge operation vectors, reducing the running time.

ITMS Algorithm. Firstly, scan the database to get a sub-tree frequently, and then all of a sub-tree frequently to construct a hash table, and this hash table into a linear table structure, then the value of the hash table is generated in accordance with the candidate policy pairwise merge two sons get frequent tree and build a new two subtrees containing all frequent hash table, and placed in the table structure, the value of the key in which the hash table tree node values, hash table stored prefix tree and subtrees vector information, and so get all frequent subtrees export.

Experiment and Analysis of Algorithm

We use the program to generate a random tree generator tree database, this random tree generation program can dynamically set various parameters (tree database size, height of the tree, the tree node fan-out, etc.) to control the random tree complexity, and assuming experiment the pruning threshold is 6. We first case under the same parameters (tree height of 7, fan-out of 6), generating in size from 10,000 to 50,000 in five tree database, the same support in the default threshold is 1% of the mining program execution, results As shown in Figure 2.

Then different support threshold experiment, random tree generator parameters using tree size is 10000, tree height of 7, 6 fanout, in support threshold varies from 10% to 0.4% The results shown in Figure 3.



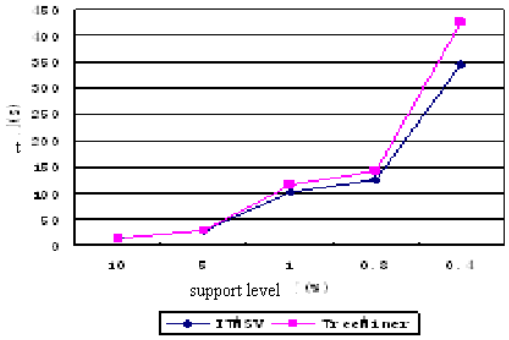


Fig. 2 Run time vs size of database Fig. 3 Run time vs threshold of support

Tree algorithm is also a factor, we conducted experiments under different tree height and fan-out, and this experiment is divided into two small experiments, each of the small experiment in six tasks, each task allocation and the number of export sub-tree experiment frequently produced as shown in Table 1 (parameter setting is expressed as (tree height, fan-out)).

Tab.1 The numbers of frequent induced subtrees

Task No.	Tree height	Export subtree number frequently generated (a)	Fanout tree change	Export subtree number frequently generated (a)
1	(4,6)	3220	(7,3)	1805
2	(5,6)	4502	(7,4)	5307
3	(6,6)	10219	(7,5)	8555
4	(7,6)	38758	(7,6)	38193
5	(8,6)	73093	(7,7)	122418
6	(9,6)	221432	(7,8)	540636

Corresponding to the running time of each task in Figure 4 Figure 5.

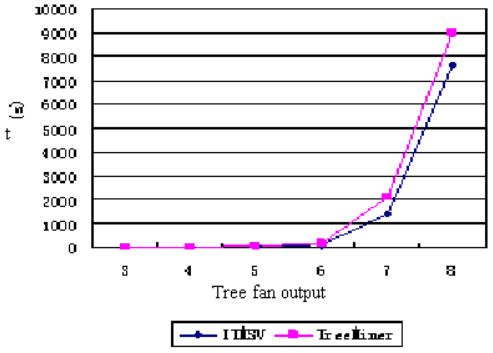


Fig. 4 Run time vs fan-out of tree

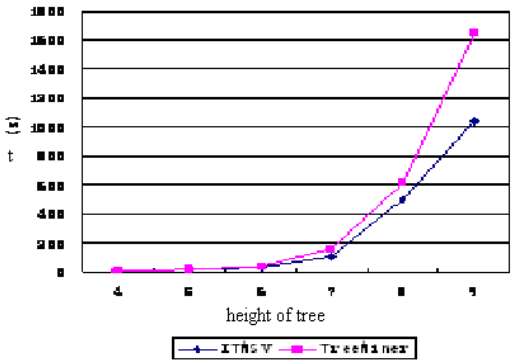


Fig. 5 Run time vs high of tree

Through the above experiments can show, ITMSV algorithm is effective and feasible, and the algorithm than TreeMiner algorithm performance has improved, particularly large amount of data processing database and tree structure is more complex, namely, tree height and fan-out is relatively large, the algorithm there are very good efficiency improvement.

Conclusion

In this paper, two concepts of vector and sub-tree pruning threshold has been put farward, and on this basis, we propose a new rapid export subtree mining algorithm ITMSV, a large number of experiments show that the algorithm is effective and feasible, and have some improvement compared to the similar algorithm. Current mining algorithms is to find frequent mainly rooted ordered tree and the study of disordered trees and freedom tree are much less, so the tree and free tree disordered mining is an important research direction while fast find sub-tree also has certain research value, these are directions of our future work.

Acknowledgements

Project: Open Fund FY2014 Key Laboratory of Eco-tourism in Hunan

References

[1] M.J. Zaki. Efficiently mining frequent trees in a forest:Algorithms and Applications. In IEEE Transaction on Knowledge and Data Engineering,special issue on Mining Biological Data. 2005,Vol.17,No.8:1021~1035.

[2] Wang K. and Liu H., Discovering Typical Structures of Documents:A Road Map Approach. Proceedings of ACM SIGIR the International Conference on Research and Development in Information Retrieval,1998:146-154.

[3] Y.T. Zhu, C. Wang, M.S. Hong. ESPM-frequent subtree mining algorithm. Computer Research and Development, 2004 (10): 1720 – 1726.