**Base concept:**

* Vertical approach for enumeration and support counting
* Based on the novel notion of primal block encoding based on prime factorization theory.
* Problem definition

items I = {i1, i2,..., im}

Itemset: is a non-empty unordered collection of items: (i1i2 ··· ik)

A sequence is an ordered list of itemsets: (s1 → s2 → ··· → sq)

Theorem 1.1

Square-free:

n = x x …x

prime factor all have multiplicity = 1

Theorem 2.2:

**Frequent pattern mining**

Frequent pattern mining problem

* **N-List**

**PPC-tree**

**PP-code:** each node of PPC-tree is a PP-code

**N-list:** a sequence of all the PP-codes of nodes registering the item in the PPC-tree.

The N-list of frequent items in Example 1:

b 🡪 < (4,8):4>

c 🡪 < (1,2):1> --- < (5,6):3>

e 🡪 < (6,5):3>

f 🡪 < (2,1):1> --- < (8,4):1> --- < (9,7):1>

a 🡪 < (4,8):4> --- < (7,3):1>

bf 🡪 < (4,8):2>

**PrePost algorithm**

1. Construct PPC-tree and identify all frequent 1-itemsets.
2. Based on PPC-tree, construct the N-list of each frequent 1-itemsets.
3. Scan PPC-tree to find all frequent 2-itemsets.
4. Mine all frequent k-itemsets.

Nodelist datastructure

Definition of Nodelists:

**Nodesets**

* definition and properties
* N-infos: For a node in POC-tree, we call the pair of its pre-order and the count registering in it, (pre-order,count), the N-info of N.
* Nodesets of items: Give a POC-tree, the Nodeset of frequent item i is a sequence of all the N-infos of nodes registering i in the POC-tree.

**Nodesets of 2 itemsets:** Nodesets(,) = {(,) | ∃ a node, N, registering , N is an ancestor of the node corresponding to (,) }

F: {(2, 1), (8, 1), (9, 1)}

B: {(4,4)}

BF: {(8,1), (9,1)}

**Nodesets of k-itemsets**: Nodesetk = Nodesetp1 ∩ Nodesetp2

**Sequential Pattern mining**

Squential pattern mining problem

A fast updated sequential pattern tree (FUSP-tree) Lin 2008

Lin et al. proposed the FASTUP(M. Y. Lin and S. Y. Lee, “Incremental update on

sequential patterns in large databases,” The Tenth IEEE

International Conference on Tools with Artificial

Intelligence, pp. 24-31, 1998.)

[8] T. P. Hong, C. Y. Wang, S. S. Tseng, “Incremental data

mining for sequential patterns using pre-large

sequences,“ The International Multiconference on

Systemics, Cybernetics and Informatics, Vol. 14, pp.

543-548, 2001.

FreeSpan is a projected-based sequential pattern mining like FP-Growth method

Prefix derive from FreeSpan (add Prefix and Suffix concept)

* Sequence:
* Subsequence:
* project database:

**Prefix:**

**Horizon vs vertical data format**

Horizon

|  |  |
| --- | --- |
| TID | List of item\_IDs |
| T100 | I1, I2, I5 |
| T200 | I2, I4 |
| T300 | I2, I3 |
| T400 | I1, I2, I4 |
| T500 | I1, I3 |
| T600 | I2, I3 |
| T700 | I1, I3 |
| T800 | I1, I2, I3, I5 |
| T900 | I1, I2, I3 |

Vertical

|  |  |
| --- | --- |
| itemset | TID\_set |
| I1 | T100, T400, T800, T900 |
| I2 | T100, T400, T800, T900 |
| I3 | T100, T400, T800, T900 |
| I4 | T100, T400, T800, T900 |
| I5 | T100, T400, T800, T900 |

FUSP tree, PPC tree chi tiết hai phần này.: nêu ví dụ, cho cả hai.

Kết hợp 🡪 Modify lại, hiệu chỉnh thuật toán.

Phải có thời gian viết bài báo.

Cạnh lề

Các công trình 🡪 các nghiên cứu liên quan

Nhiều hơn 3 🡪 chuyển người nghiên cứu chính và các động sự

Lớp thành họ

Nội dung và phương pháp tham khảo Tài, Time line copy từ nội dung và phương pháp

Nếu 6 tháng 🡪 bỏ ra phần đọc tài liệu

Công việc tiếp theo