CHƯƠNG 2: KHAI PHÁ LUẬT KẾT HỢP

2.1. MỘT SỐ KHÁI NIỆM CƠ BẢN

2.2. TÌM TẬP PHỔ BIẾN VỚI GIẢI THUẬT APRIORI

2.3. SINH LUẬT KẾT HỢP TỪ CÁC TẬP PHỔ BIẾN

2.4. TÌM TẬP PHỔ BIẾN VỚI GIẢI THUẬT FP - GROWTH

2.1. MỘT SỐ KHÁI NIỆM CƠ BẢN

2.1.1. Khái niệm mục (item) và tập mục (item set)

- Cho một tập gồm n đối tượng $I = \{I_1, I_2, I_3, \ldots, I_n\}$, mỗi phần tử $I_i \in I$ được gọi là một mục (item). Một tập con bất kỳ $X \subseteq I$ được gọi là một tập mục (item set).
- Cho một tập $D = \{T_1, T_2, \ldots, T_m\}$, mỗi phần tử $T_j \in D$ được gọi là một giao dịch (transaction) và là một tập con nào đó của I ($T_j \subseteq I$). Người ta gọi D là cơ sở dữ liệu giao dịch (transaction database). Số giao dịch có trong D ký hiệu là |D|.

Ví dụ: I = {A, B, C, D, E, F},
X = {A, D, E} là một tập mục. Một cơ sở dữ liệu giao dịch D gồm các tập con T_j khác nhau của I:

T_1	{A, B, C, D}
T_2	{A, C, E}
T_3	{A, E}
T_4	{A, E, F}
T_5	{A, B, C, E, F}

Milk, Bread, Coke
10:05

Beer, Bread
10:12

Beer, Milk, Diaper, Coke
10:15

Beer, Milk, Diaper, Bread
10:23

Milk, Diaper, Coke
10:30

2.1.2. Độ hỗ trợ (support) ứng với một tập mục

"Độ hỗ trợ ứng với tập mục X là xác suất xuất hiện của X trong cơ sở dữ liệu giao dịch D"

Hoặc

"Đỗ hỗ trợ ứng với tập mục X là tỷ lệ các giao dịch có chứa X trên tổng số các giao dịch có trong cơ sở dữ liệu giao dịch D"

$$\sup(X) = \frac{C(X)}{|D|}$$

Trong đó: C(X) là số lần xuất hiện của X hay số giao dịch có chứa X

Ví dụ: X = {A, E} thì C(X) = 4 và sup(X) = 4/5 = 80%

T_1	{A, B, C, D}
T_2	{A, C, E}
T_3	{A, E}
T_4	{A, E, F}
T_5	{A, B, C, E, F}

Các tập mục có độ hỗ trợ lớn hơn một giá trị ngưỡng minsup nào đó cho trước được gọi là các tập phổ biến (frequent item set).

2.1.3. Luật kết hợp (Association Rule)

- Cho hai tập mục X, Y ⊆ I, X ∩ Y = ϕ. Luật kết hợp ký hiệu là X → Y chỉ ra mối ràng buộc của tập mục Y theo tập mục X, nghĩa là khi X xuất hiện trong cơ sở dữ liệu giao dịch thì sẽ kéo theo sự xuất hiện của Y với một một tỷ lệ nào đấy.
- Luật kết hợp được đặc trưng bởi:

Độ hỗ trợ của luật: là tỷ lệ (hay xác suất) xuất hiện cả X và Y trong cùng một giao dịch.

$$\sup(X \to Y) = \sup(X \cup Y) = \frac{C(X \cup Y)}{|D|}$$

Độ tin cậy của luật: là tỷ lệ các giao dịch có chứa cả X và Y so với các giao dịch có chứa X.

$$\text{conf}(X \to Y) = \frac{C(X \cup Y)}{C(X)} = \frac{\sup(X \to Y)}{\sup(X)}$$

Trong đó: C(X ∪ Y): Số giao dịch có chứa cả X và Y.
C(X): Số giao dịch có chứa X.

- **Luật mạnh**: Các luật có độ hỗ trợ lớn hơn một giá trị ngưỡng **minsup** và độ tin cậy lớn hơn một giá trị ngưỡng **minconf** cho trước được gọi là các luật "mạnh" hay "luật có giá trị" (strong association rules).

Cụ thể:

Nếu đồng thời sup(X→Y) ≥ minsup và conf(X→Y) ≥ minconf thì X→Y được gọi là luật mạnh (strong association rule).

2.1.4. Bài toán khai phá luật kết hợp

Input: Cơ sở dữ liệu giao dịch D.

Các giá trị ngưỡng minsup, minconf.

Output: Tất cả các luật mạnh.

Để giải quyết bài toán khai phá luật kết hợp bao giờ cũng thường trải qua hai pha:

Pha 1: Sinh tất cả các tập phổ biến có thể có. Ở pha này ta sử dụng các giải thuật tìm tập phổ biến như: Apriori, FP-Growth,...

Pha 2: Ứng với mỗi tập phổ biến K tìm được ở pha 1, tách K thành hai tập X, Y không giao nhau (K = X ∪ Y và X ∩ Y = ϕ). Tính độ tin cậy của luật kết hợp X → Y, nếu độ tin cậy trên ngưỡng **minconf** thì nó là luật mạnh. Chú ý là nếu tập K có k phần tử thì số tập con thực sự của K sẽ là $2^k - 2$, tức là từ K ta sẽ sinh được tối đa là $2^k - 2$ luật.

Lưu ý: *Trong một số giải thuật, để xác định một tập là phổ biến người ta không sử dụng khái niệm* ***độ hỗ trợ*** *mà sử dụng khái niệm* ***số lần xuất hiện*** *(support count). Nếu* ***số lần xuất hiện*** *của tập mục trong cơ sở dữ liệu giao dịch lớn hơn một* ***giá trị ngưỡng*** *nào đấy thì nó là tập phổ biến. Giá trị ngưỡng này được xác định là:*

$$\text{mincount} = \lceil \text{minsup} * |D| \rceil$$

2.2. TÌM TẬP PHỔ BIẾN VỚI GIẢI THUẬT APRIORI

2.2.1. Nguyên lý Apriori

"Nếu một tập mục là tập phổ biến thì mọi tập con khác rỗng bất kỳ của nó cũng là tập phổ biến"

Chứng minh:

Xét X' ⊆ X. Ký hiệu p là ngưỡng độ hỗ trợ minsup. *Một tập mục xuất hiện bao nhiêu lần thì các tập con chứa trong nó cũng xuất hiện ít nhất bấy nhiêu lần*, nên ta có:

$$C(X') \geq C(X) \quad (1).$$

X là tập phổ biến nên:

$$\sup(X) = \frac{C(X)}{|D|} \geq p \Rightarrow C(X) \geq p|D| \quad (2)$$

Từ (1) và (2) suy ra: $C(X') > p|D| \Rightarrow \sup(X') = \frac{C(X')}{|D|} \geq p$

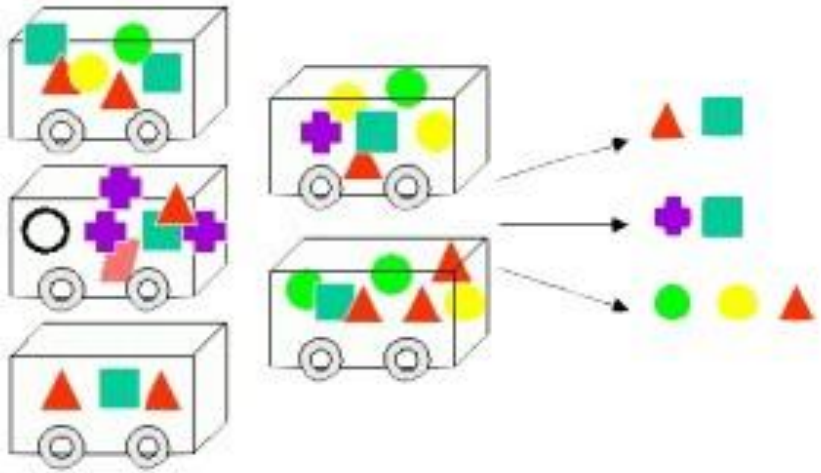
Tức là X' cũng là tập phổ biến (đpcm).

2.2.2. Giải thuật Apriori

Mục đích: Tìm ra tất cả các tập phổ biến có thể có.

- Dựa trên nguyên lý Apriori.
- Hoạt động dựa trên Quy hoạch động:

Từ các tập F_i = { c_i | c_i là tập phổ biến, $|c_i| = i$} gồm mọi tập mục phổ biến có độ dài i ($1 \leq i \leq k$), đi tìm tập F_{k+1} gồm mọi tập mục phổ biến có độ dài k+1. Các mục $I_1, I_2, \ldots, I_n$ trong tập I được coi là sắp xếp theo một thứ tự cố định.



Input: - Cơ sở dữ liệu giao dịch D = {t_1, t_2,…, t_m}.
- Ngưỡng độ hỗ trợ tối thiểu minsup.

Output: - Tập hợp tất cả các tập phổ biến.

$\text{mincount} = \lceil \text{minsup} * |D| \rceil$;

F_1 = { các tập phổ biến có độ dài 1};

for(k=1; F_k != $\varnothing$; k++)

{

C_{k+1} = **Apriori_gen**(F_k);

for each t ∈ D

{

$C_t = \{ c \mid c \in C_{k+1} \text{ và } c \subseteq t\}$;

for each c ∈ C_t

c.count++;

}

$F_{k+1} = \{c \in C_{k+1} \mid c.\text{count} \geq \text{mincount}\}$;

}

return $F = \bigcup_k F_k$

Thủ tục con Apriori_gen

- Thủ tục con Apriori_gen có nhiệm vụ sinh ra (generation) các tập mục có độ dài k+1 từ các tập mục có độ dài k trong tập F_k.
- Được thi hành qua hai bước: nối (join) các tập mục có chung các tiền tố (prefix) và sau đó áp dụng nguyên lý Apriori để loại bỏ bớt những tập không thỏa mãn.

Cụ thể:

❖ Bước nối: Sinh các tập mục c là ứng viên của tập phổ biến có độ dài k+1 bằng cách kết hợp hai tập phổ biến l_i và $l_j \in F_k$ có độ dài k và trùng nhau ở k-1 mục đầu tiên: $c = l_i + l_j = \{i_1, i_2,\ldots, i_{k-1}, i_k, i_{k'}\}$.
Với $l_i = \{i_1, i_2,\ldots, i_{k-1}, i_k\}$, $l_j = \{i_1, i_2,\ldots, i_{k-1}, i_{k'}\}$, và $i_1 \le i_2 \le \ldots \le i_{k-1} \le i_k \le i_{k'}$.

❖ Bước tỉa: Giữ lại tất cả các ứng viên c thỏa thỏa mãn nguyên lý Apriori tức là mọi tập con có độ dài k của nó đều là tập phổ biến ($\forall s_k \subseteq c$ và $|s_k| = k$ thì $s_k \in F_k$).

function **Apriori_gen**(F_k: tập các tập phổ biến độ dài k): Tập ứng viên có độ dài k+1

{

  $C_{k+1} = \varnothing$;

  for each $l_i \in F_k$

    for each $l_j \in F_k$

    if ($l_i[1]=l_j[1]$) and ($l_i[2]=l_j[2]$) … and ($l_i[k-1]=l_j[k-1]$) and ($l_i[k]<l_j[k]$) **then**

    {

      c = {$l_i[1], l_i[2], l_i[3],\ldots, l_i[k], l_j[k]$};

      if **has_infrequent_subset**(c, F_k) **then**

      delete c;

      else $C_{k+1} = C_{k+1} \cup \{c\}$;

    }

  return C_{k+1};

}

Hàm has_infrequent_subset làm nhiệm vụ kiểm tra xem một ứng viên có độ dài k+1 có chứa tập không phổ biến hay không, nếu có thì ứng viên lập tức bị loại. Đây là bước tỉa dựa trên nguyên lý Apriori nhằm loại bỏ nhanh các ứng viên không thỏa mãn.

function **has_infrequent_subset**(c: Ứng viên có độ dài k+1, F_k: Tập các tập phổ biến độ dài k): Boolean

```
{
        for each s_k ⊂ c
                if s_k ∉ F_k then return True;
        return False;
}
```

2.3. SINH LUẬT KẾT HỢP TỪ CÁC TẬP PHỔ BIẾN

Để sinh các luật kết hợp:

- Với mỗi tập phổ biến X ∈ F, ta xác định các tập mục không rỗng là con của X.
- Với mỗi tập mục con S không rỗng của X ta sẽ thu được một luật kết hợp là S→(X\S). Nếu độ tin cậy của luật thỏa mãn ngưỡng minconf thì luật đó là luật mạnh.

$$\text{conf}(S \rightarrow (X \setminus S)) = \frac{C(X)}{C(S)} \geq \text{minconf}$$

function Rules_Generation(F: Tập các tập phổ biến): Tập các luật kết hợp mạnh

{

R = ∅;

F=F \ F_1; *//Các tập phổ biến độ dài 1 không dùng để sinh luật*

for each X ∈ F

for each S ⊂ X

if conf(S→(X\S)) ≥ minconf **then**

R = R ∪ { S→(X\S)};

return R;

}

BÀI TẬP ÁP DỤNG

Bài tập số 1: Cho I = {A, B, C, D, E, F} và cơ sở dữ liệu giao dịch D:

T1	{A, B, C, F}
T2	{A, B, E, F}
T3	{A, C}
T4	{D, E}
T5	{B, F}

Chọn ngưỡng minsup = 25% và minconf = 75%. Hãy xác định các luật kết hợp mạnh.

$mincount = \lceil \min sup * |D| \rceil = \lceil 25\% * 5 \rceil = \lceil 1.25 \rceil = 2$

Tập mục	Số lần xuất hiện
{A}	3
{B}	3
{C}	2
~~{D}~~	~~1~~
{E}	2
{F}	3

Sinh các tập phổ biến có độ dài 1

F_1	Số lần xuất hiện
{A}	3
{B}	3
{C}	2
{E}	2
{F}	3

Sinh các tập có độ dài 2 bằng cách nối các tập có độ dài 1

Tập mục
{A, B}
{A, C}
{A, E}
{A, F}
{B, C}
{B, E}
{B, F}
{C, E}
{C, F}
{E, F}

C_2	Số lần xuất hiện
{A, B}	2
{A, C}	2
~~{A, E}~~	~~1~~
{A, F}	2
~~{B, C}~~	~~1~~
~~{B, E}~~	~~1~~
{B, F}	3
~~{C, E}~~	~~0~~
~~{C, F}~~	~~1~~
~~{E, F}~~	~~1~~

F_2	Số lần xuất hiện
{A, B}	2
{A, C}	2
{A, F}	2
{B, F}	3

Sinh các tập mục có độ dài 3 từ tập phổ biến F_2

Tập mục
~~{A, B, C}~~
{A, B, F}
~~{A, C, F}~~

Loại các tập mục không thỏa mãn nguyên lý Apriori

C_3	Số lần xuất hiện
{A, B, F}	2

F_3	Số lần xuất hiện
{A, B, F}	2

F_3 chỉ có một phần tử nên không thể tiếp tục kết nối để sinh F_4. Thuật toán kết thúc. Ta có tập các tập phổ biến là:

$F = \{\{A\}, \{B\}, \{C\}, \{E\}, \{F\}, \{A, B\}, \{A, C\}, \{A, F\}, \{B, F\}, \{A, B, F\}\}$

{A, B} có thể sinh các luật: {A}→{B}, {B}→{A}

$$\text{conf}(\{A\} \rightarrow \{B\}) = \frac{C(\{A,B\})}{C(\{A\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{B\} \rightarrow \{A\}) = \frac{C(\{A,B\})}{C(\{B\})} = \frac{2}{3} = 66.7\%$$

{A, C} có thể sinh các luật: {A}→{C}, {C}→{A}

$$\text{conf}(\{A\} \rightarrow \{C\}) = \frac{C(\{A,C\})}{C(\{A\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{C\} \rightarrow \{A\}) = \frac{C(\{A,C\})}{C(\{C\})} = \frac{2}{2} = 100\%$$

{A, F} có thể sinh các luật: {A}→{F}, {F}→{A}

$$\text{conf}(\{A\} \rightarrow \{F\}) = \frac{C(\{A,F\})}{C(\{A\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{F\} \rightarrow \{A\}) = \frac{C(\{A,F\})}{C(\{F\})} = \frac{2}{3} = 66.7\%$$

{B, F} có thể sinh các luật: {B}→{F}, {F}→{B}

$$\text{conf}(\{B\} \rightarrow \{F\}) = \frac{C(\{B,\ F\})}{C(\{B\})} = \frac{3}{3} = 100\%$$

$$\text{conf}(\{F\} \rightarrow \{B\}) = \frac{C(\{B, F\})}{C(\{F\})} = \frac{3}{3} = 100\%$$

{A, B, F} có thể sinh các luật: {A}→{B, F}, {A, B}→{F}, {B}→{A, F}, {B, F}→{A}, {F}→{A, B}, {A, F}→{B}

$$\text{conf}(\{A\} \rightarrow \{B, F\}) = \frac{C(\{A, B, F\})}{C(\{A\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{A, B\} \rightarrow \{F\}) = \frac{C(\{A, B, F\})}{C(\{A, B\})} = \frac{2}{2} = 100\%$$

$$\text{conf}(\{B\} \rightarrow \{A, F\}) = \frac{C(\{A, B, F\})}{C(\{B\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{B, F\} \rightarrow \{A\}) = \frac{C(\{A, B, F\})}{C(\{B, F\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{F\} \rightarrow \{A,B\}) = \frac{C(\{A,B,F\})}{C(\{F\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{A,F\} \rightarrow \{B\}) = \frac{C(\{A,B,F\})}{C(\{A,F\})} = \frac{2}{2} = 100\%$$

Như vậy các luật kết hợp mạnh thu được gồm:
{C}→{A}, {B}→{F}, {F}→{B}, {A, B}→{F}, {A, F}→{B}

Bài tập số 2: Cho I = {A, B, C, D, E, F} và cơ sở dữ liệu giao dịch D:

T1	{D, E}
T2	{A, B, D, E}
T3	{A, B, D}
T4	{C, D, E}
T5	{F}
T6	{B, C, D}

Chọn ngưỡng minsup = 20% và minconf = 70%. Hãy xác định các luật kết hợp mạnh.

$mincount = \lceil \min sup * |D| \rceil = \lceil 20\% * 6 \rceil = \lceil 1.2 \rceil = 2$

Tập mục	Số lần xuất hiện
{A}	2
{B}	3
{C}	2
{D}	5
{E}	3
{F}	1

F_1	Số lần xuất hiện
{A}	2
{B}	3
{C}	2
{D}	5
{E}	3

Tập mục
{A, B}
{A, C}
{A, D}
{A, E}
{B, C}
{B, D}
{B, E}
{C, D}
{C, E}
{D, E}

C_2	Số lần xuất hiện
{A, B}	2
~~{A, C}~~	~~0~~
{A, D}	2
~~{A, E}~~	~~1~~
~~{B, C}~~	~~1~~
{B, D}	3
~~{B, E}~~	~~1~~
{C, D}	2
~~{C, E}~~	~~1~~
{D, E}	3

F_2	Số lần xuất hiện
{A, B}	2
{A, D}	2
{B, D}	3
{C, D}	2
{D, E}	3

Tập mục
{A, B, D}

C_3	Số lần xuất hiện
{A, B, D}	2

F_3	Số lần xuất hiện
{A, B, D}	2

Tập F_3 chỉ có một phần tử nên không thể tiếp tục kết nối để sinh ứng viên cho tập F_4. Thuật toán kết thúc. Tập các tập phổ biến thu được:

F = {{A}, {B}, {C}, {D}, {E}, {A, B}, {A, D}, {B, D}, {C, D}, {D, E}, {A, B, D}}

{A, B} sinh luật: {A}→{B}, {B}→{A}

$$\text{conf}(\{A\} \rightarrow \{B\}) = \frac{C(\{A,B\})}{C(\{A\})} = \frac{2}{2} = 100\%$$

$$\text{conf}(\{B\} \rightarrow \{A\}) = \frac{C(\{A,B\})}{C(\{B\})} = \frac{2}{3} = 66.7\%$$

{A, D} sinh luật: {A}→{D}, {D}→{A}

$$\text{conf}(\{A\} \rightarrow \{D\}) = \frac{C(\{A,D\})}{C(\{A\})} = \frac{2}{2} = 100\%$$

$$\text{conf}(\{D\} \rightarrow \{A\}) = \frac{C(\{A,B\})}{C(\{D\})} = \frac{2}{5} = 40\%$$

{B, D} sinh luật: {B}→{D}, {D}→{B}

$$\text{conf}(\{B\} \rightarrow \{D\}) = \frac{C(\{B,D\})}{C(\{B\})} = \frac{3}{3} = 100\%$$

$$\text{conf}(\{D\} \rightarrow \{B\}) = \frac{C(\{B,D\})}{C(\{D\})} = \frac{3}{5} = 60\%$$

{C, D} sinh luật: {C}→{D}, {D}→{C}

$$\text{conf}(\{D\} \rightarrow \{C\}) = \frac{C(\{C,D\})}{C(\{D\})} = \frac{2}{5} = 40\%$$

$$\text{conf}(\{C\} \rightarrow \{D\}) = \frac{C(\{C,D\})}{C(\{C\})} = \frac{2}{2} = 100\%$$

{D, E} sinh luật: {D}→{E}, {E}→{D}

$$\text{conf}(\{D\} \rightarrow \{E\}) = \frac{C(\{D,E\})}{C(\{D\})} = \frac{3}{5} = 60\%$$

$$\text{conf}(\{E\} \rightarrow \{D\}) = \frac{C(\{D,E\})}{C(\{E\})} = \frac{3}{3} = 100\%$$

{A, B, D} sinh luật: {A}→{B, D}, {A, B}→{D}, {B}→{A, D}, {B, D}→{A}, {D}→{A, B}, {A, D}→B

$$\text{conf}(\{A,B\} \rightarrow \{D\}) = \frac{C(\{A,B,D\})}{C(\{A,B\})} = \frac{2}{2} = 100\%$$

$$\text{conf}(\{A\} \rightarrow \{B,D\}) = \frac{C(\{A,B,D\})}{C(\{A\})} = \frac{2}{2} = 100\%$$

$$\text{conf}(\{B\} \rightarrow \{A, D\}) = \frac{C(\{A, B, D\})}{C(\{B\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{B, D\} \rightarrow \{A\}) = \frac{C(\{A, B, D\})}{C(\{B, D\})} = \frac{2}{3} = 66.7\%$$

$$\text{conf}(\{D\} \rightarrow \{A, B\}) = \frac{C(\{A, B, D\})}{C(\{D\})} = \frac{2}{5} = 40\%$$

$$\text{conf}(\{A, D\} \rightarrow \{B\}) = \frac{C(\{A, B, D\})}{C(\{A, D\})} = \frac{2}{2} = 100\%$$

Các luật kết hợp mạnh thu được gồm:

1. {A}→{B}
2. {A}→{D}
3. {B}→{D}
4. {C}→{D}
5. {E}→{D}
6. {A}→{B, D}
7. {A,B}→{D}
8. {A, D}→B

2.4. TÌM TẬP PHỔ BIẾN VỚI GIẢI THUẬT FP-GROWTH

Tư tưởng: Cho phép phát hiện ra các tập phổ biến mà không cần khởi tạo các ứng viên.

BƯỚC 1: Xây dựng một cấu trúc dữ liệu thu gọn gọi là cây FP. Bước này chỉ yêu cầu quét CSDL giao dịch 02 lần.

BƯỚC 2: Kết xuất các mục phổ biến dựa trên cây FP. Thao tác duyệt cây được thực hiện tại bước này.

BƯỚC 1: XÂY DỰNG CÂY FP

Input:

- D, a transaction database;
- *min_sup*, the minimum support count threshold.

Output: The complete set of frequent patterns.

Method:

1. The FP-tree is constructed in the following steps:
   (a) Scan the transaction database D once. Collect F, the set of frequent items, and their support counts. Sort F in support count descending order as L, the *list* of frequent items.
   (b) Create the root of an FP-tree, and label it as "null." For each transaction *Trans* in D do the following. Select and sort the frequent items in *Trans* according to the order of L. Let the sorted frequent item list in *Trans* be $[p|P]$, where p is the first element and P is the remaining list. Call **insert_tree**$([p|P], T)$, which is performed as follows. If T has a child N such that $N.item\text{-}name = p.item\text{-}name$, then increment N's count by 1; else create a new node N, and let its count be 1, its parent link be linked to T, and its node-link to the nodes with the same *item-name* via the node-link structure. If P is nonempty, call **insert_tree**(P, N) recursively.
2. The FP-tree is mined by calling **FP_growth**(*FP_tree*, *null*), which is implemented as follows.

(Jiawei Han and Micheline Kamber, ***Data Mining Concepts and Techniques)***

- Quét CSDL giao dịch và đếm số lần xuất hiện ứng với mỗi mục.
- Loại bỏ các mục không phổ biến.
- Sắp lại thứ tự các mục trong mỗi giao dịch theo thứ tự giảm dần của số lần xuất hiện.
- Mỗi nút của cây tương ứng với một mục và được gắn trọng số là số lần xuất hiện.
- Giải thuật FP-Growth đọc lần lượt từng giao dịch và ánh xạ tương ứng với mỗi đường đi (xuất phát từ nút gốc) trên cây FP.

- Thứ tự sắp xếp của các mục được tuân thủ trong suốt quá trình xây dựng cây FP.
- Các đường đi có thể có thể có những đoạn trùng nhau do các giao dịch có các phần tử chung (chung tiền tố trong dãy). Mỗi lần có phần tử trùng thì trọng số của đỉnh ở vị trí trùng được tăng lên 1.
- Con trỏ được sử dụng để duy trì danh sách kết nối đơn giữa các nút đại diện cho cùng một mục.

BƯỚC 2: SINH TẬP PHỔ BIẾN (duyệt cây FP)

```
procedure FP_growth(Tree, α)
(1)   if Tree contains a single path P then
(2)     for each combination (denoted as β) of the nodes in the path P
(3)       generate pattern β ∪ α with support_count = minimum support count of nodes in β;
(4)   else for each a_i in the header of Tree {
(5)     generate pattern β = a_i ∪ α with support_count = a_i.support_count;
(6)     construct β's conditional pattern base and then β's conditional FP_tree Tree_β;
(7)     if Tree_β ≠ ∅ then
(8)       call FP_growth(Tree_β, β); }
```

(Jiawei Han and Micheline Kamber, ***Data Mining Concepts and Techniques)***

Ứng với mỗi mục phổ biến I_i:

- **Xây dựng tập các cơ sở mẫu có điều kiện (conditional pattern base). Mỗi mẫu có điều kiện là một đường đi nối từ đỉnh gốc tới đỉnh cha kề với đỉnh có chứa mục I_i. Mỗi mẫu được gán trọng số bằng với trọng số của đỉnh có chứa mẫu I_i ở cuối đường đi.**
- **Xây dựng cây FP có điều kiện (conditional FP-tree) dựa trên việc kết hợp các mẫu có chung tiền tố (nếu có). Khi đó trọng số ứng với mỗi đỉnh là tổng các trọng số được ghép.**
- **Duyệt cây FP có điều kiện để sinh các tập phổ biến có hậu tố là I_i.**

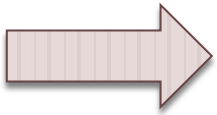
Ví dụ 1: Cho cơ sở dữ liệu giao dịch D gồm các giao dịch:

TID	*Items bought*
100	{*f*, *a*, *c*, *d*, *g*, *i*, *m*, *p*}
200	{*a*, *b*, *c*, *f*, *l*, *m*, *o*}
300	{*b*, *f*, *h*, *j*, *o*}
400	{*b*, *c*, *k*, *s*, *p*}
500	{*a*, *f*, *c*, *e*, *l*, *p*, *m*, *n*}

Biết ngưỡng minsup = 60%. Hãy tìm các tập phổ biến.

- Quét CSDL để tính số lần xuất hiện (support count) ứng với mỗi mục:

TID	Items bought
100	{f, a, c, d, g, i, m, p}
200	{a, b, c, f, l, m, o}
300	{b, f, h, j, o}
400	{b, c, k, s, p}
500	{a, f, c, e, l, p, m, n}



Item	frequency
f	4
c	4
a	3
b	3
m	3
p	3

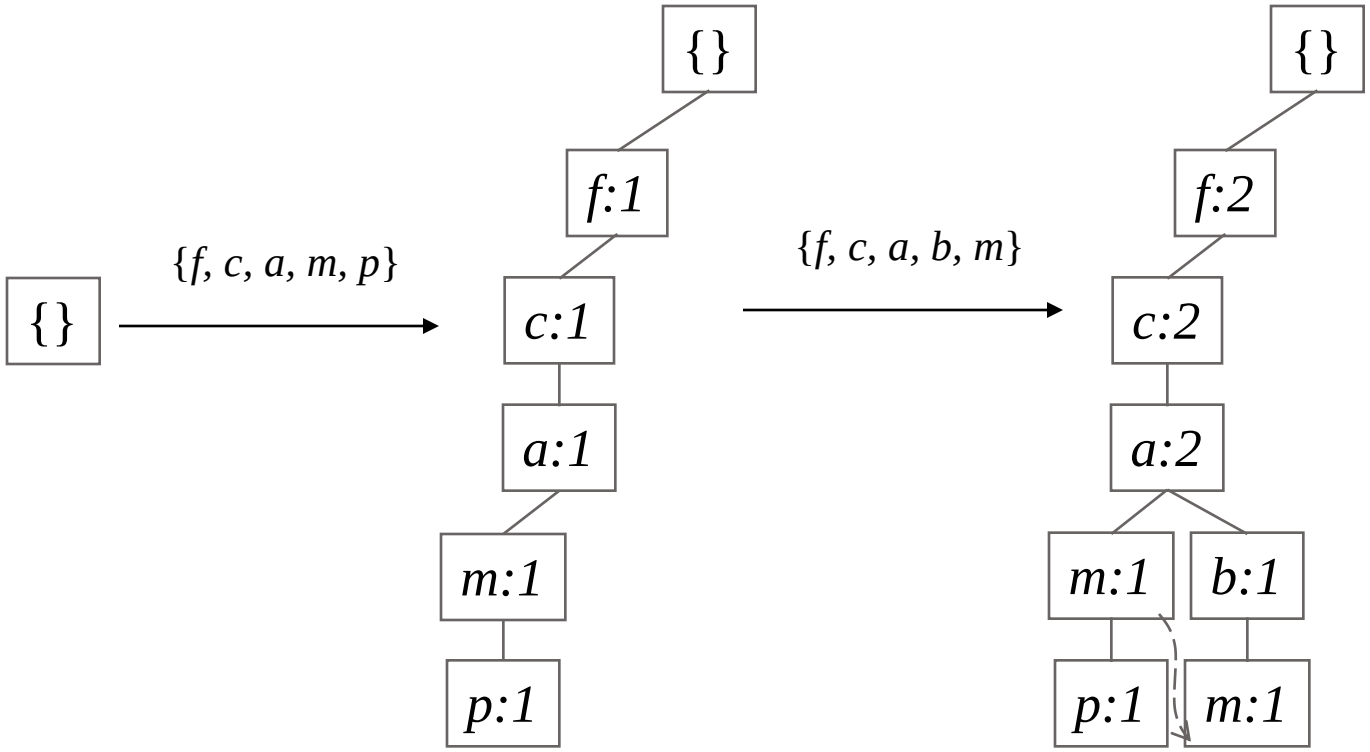
mincount = 3

- Loại bỏ các mục không phải là phổ biến.
- Sắp các mục trong mỗi giao dịch theo thứ tự giảm của support count.

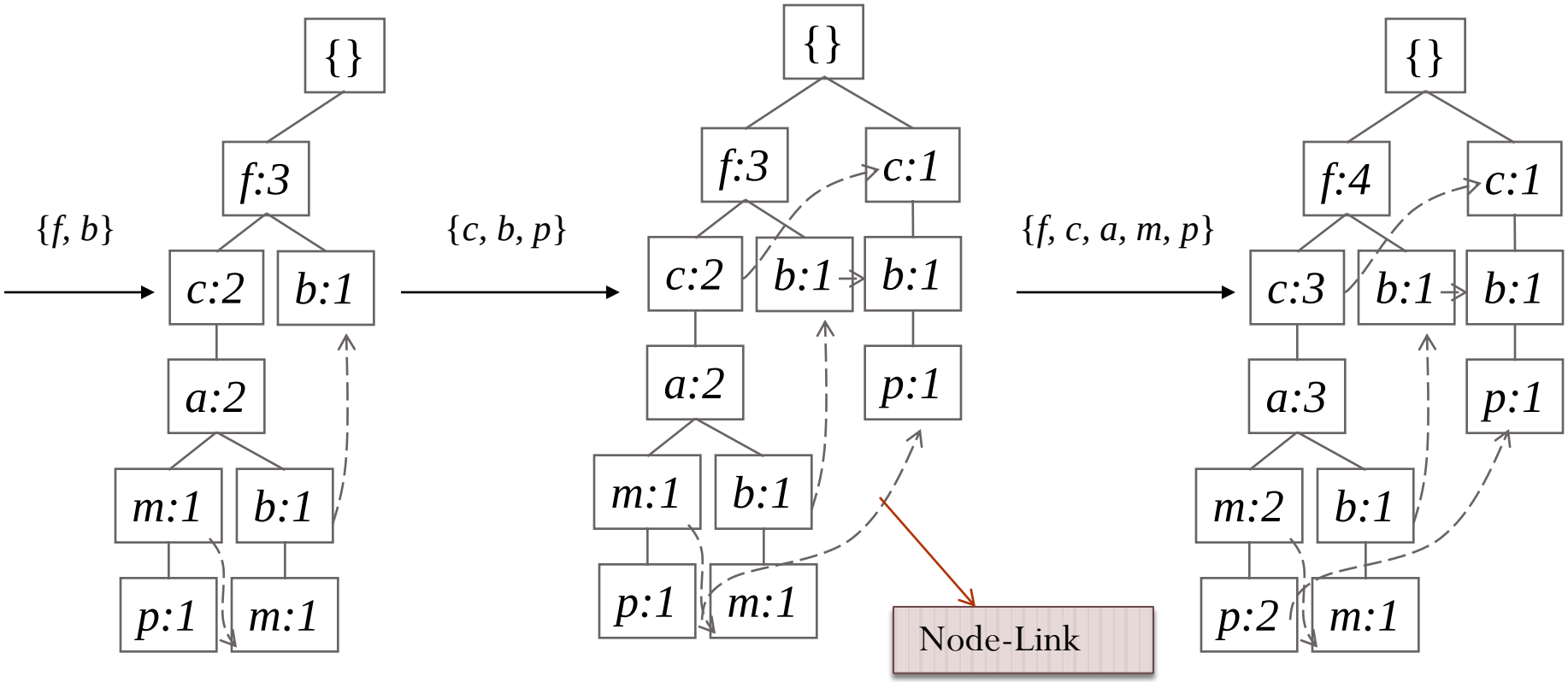
TID	Items bought	(ordered) frequent items
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

TID	Items bought	(ordered) frequent items
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

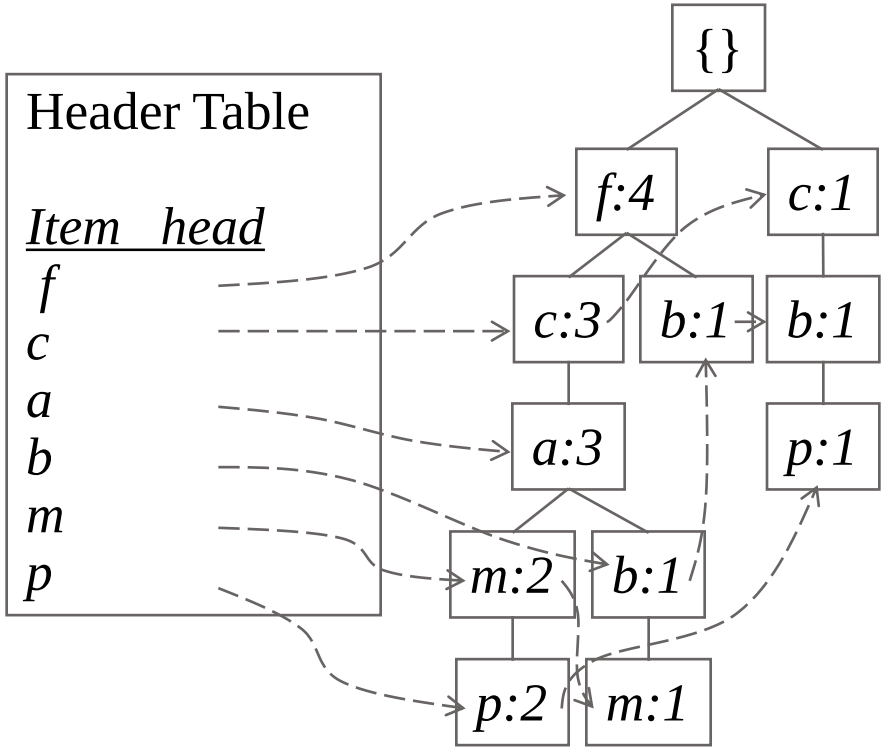
Đọc từng giao dịch và ánh xạ vào cây FP:

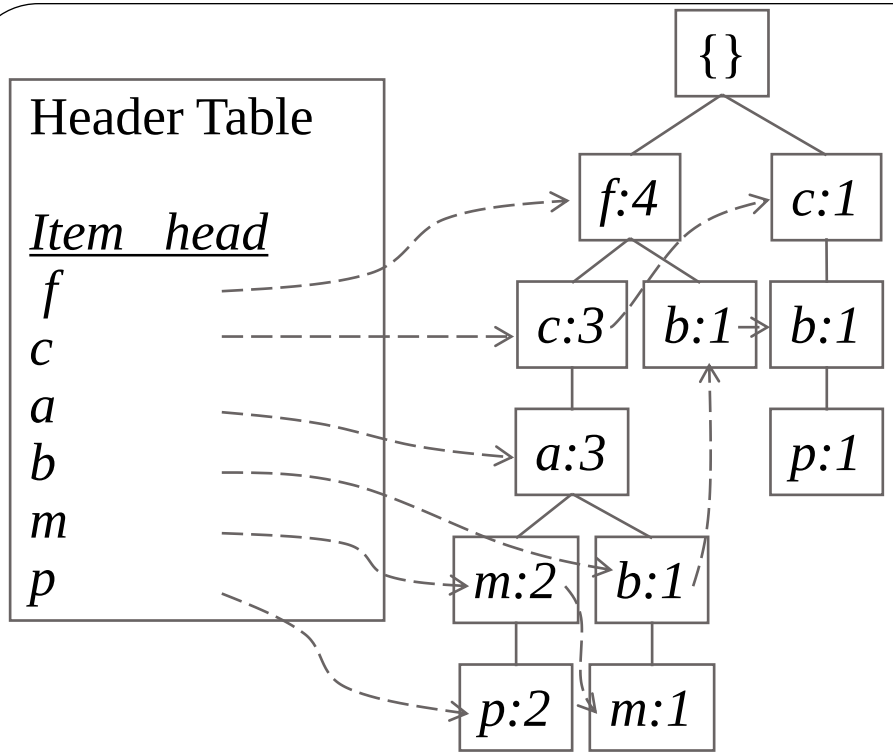


Đọc từng giao dịch và ánh xạ vào cây FP (tiếp)



Cây FP hoàn chỉnh:





Mục	Cơ sở mẫu có điều kiện	Cây FP có điều kiện	Tập phổ biến
p	*fcam:2, cb:1*	*{c:3}*	*p:3, cp:3*
m	*fca:2, fcab:1*	*{f:3, c:3, a:3}*	*m:3, fm:3, cm:3, am:3, fcm:3, fam:3, cam:3*
b	*fca:1, f:1, c:1*	*Null*	*b:3*
a	*fc:3*	*{f:3, c:3}*	*a:3, fa:3, ca:3*
c	*f:3*	*{f:3}*	*c:3, fc:3*
f	*Null*	*Null*	*f:3*

Ví dụ 2: Cho cơ sở dữ liệu giao dịch D gồm các giao dịch:

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

Biết ngưỡng minsup = 22%. Hãy tìm các tập phổ biến.

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

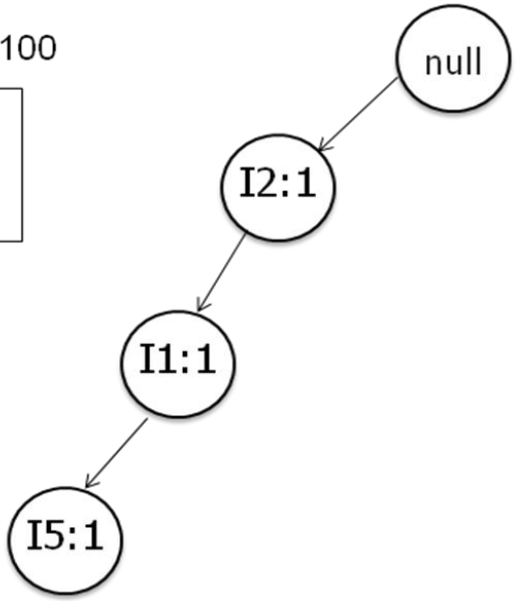
Đếm số lần xuất hiện của các mục và sắp theo thứ tự giảm dần:

Tập mục	Số lần xuất hiện
I_2	7
I_1	6
I_3	6
I_4	2
I_5	2

Giao dịch	Danh sách mục
T100	I_2, I_1, I_5
T200	I_2, I_4
T300	I_2, I_3
T400	I_2, I_1, I_4
T500	I_1, I_3
T600	I_2, I_3
T700	I_1, I_3
T800	I_2, I_1, I_3, I_5
T900	I_2, I_1, I_3

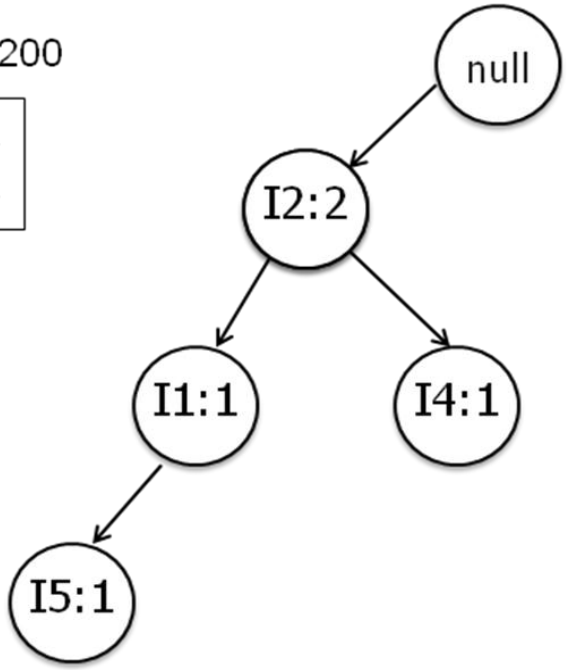
Đọc :T100

I2,
I1,
I5



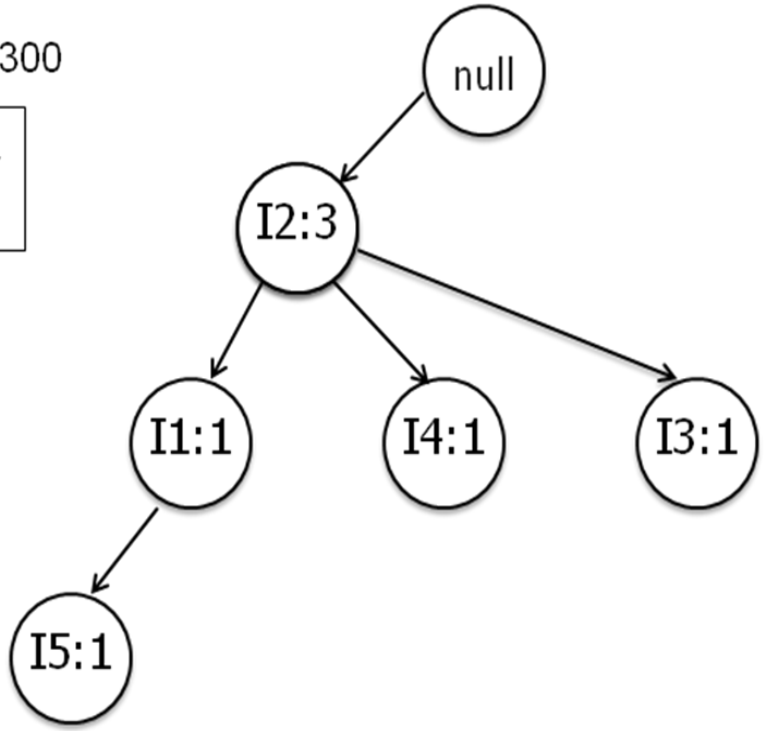
Đọc :T200

I2,
I4,



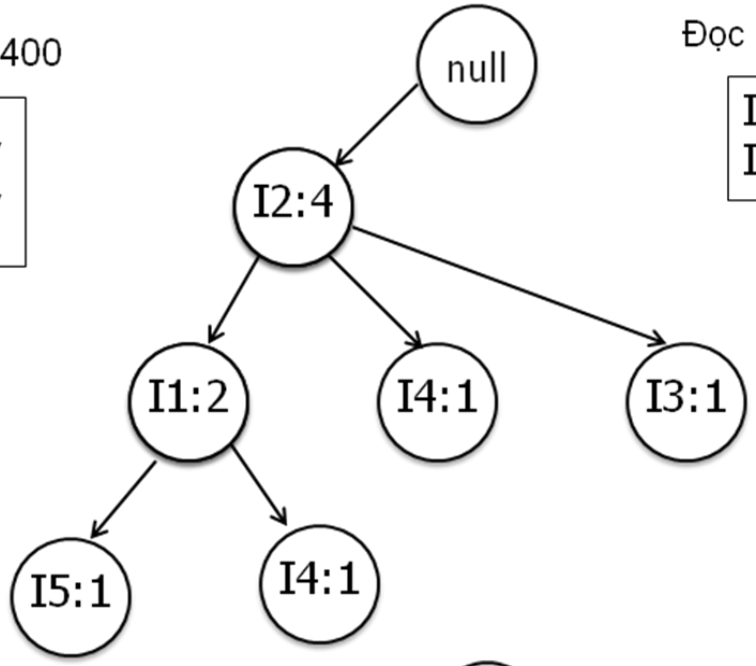
Đọc :T300

I2,
I3



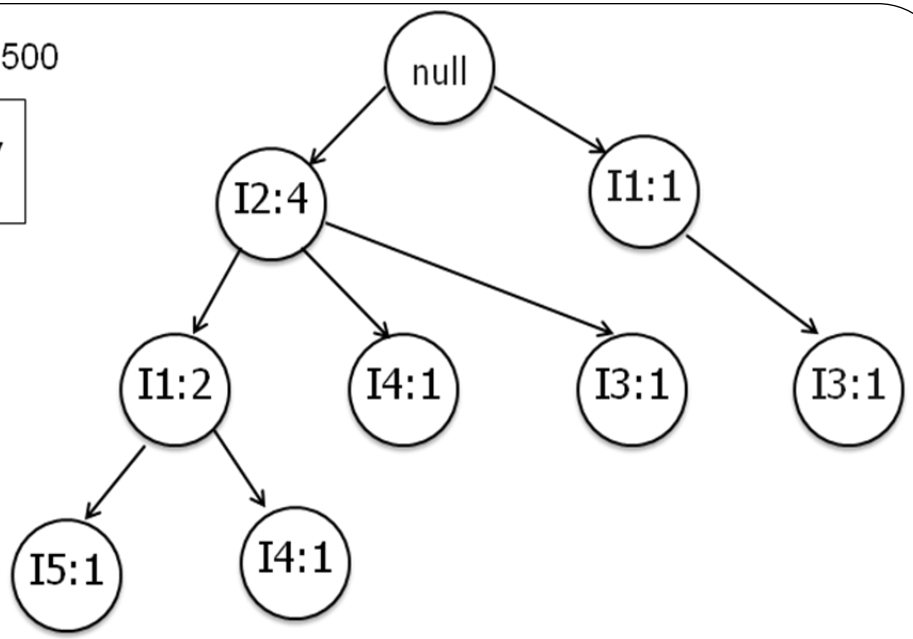
Đọc :T400

I2, I1, I4



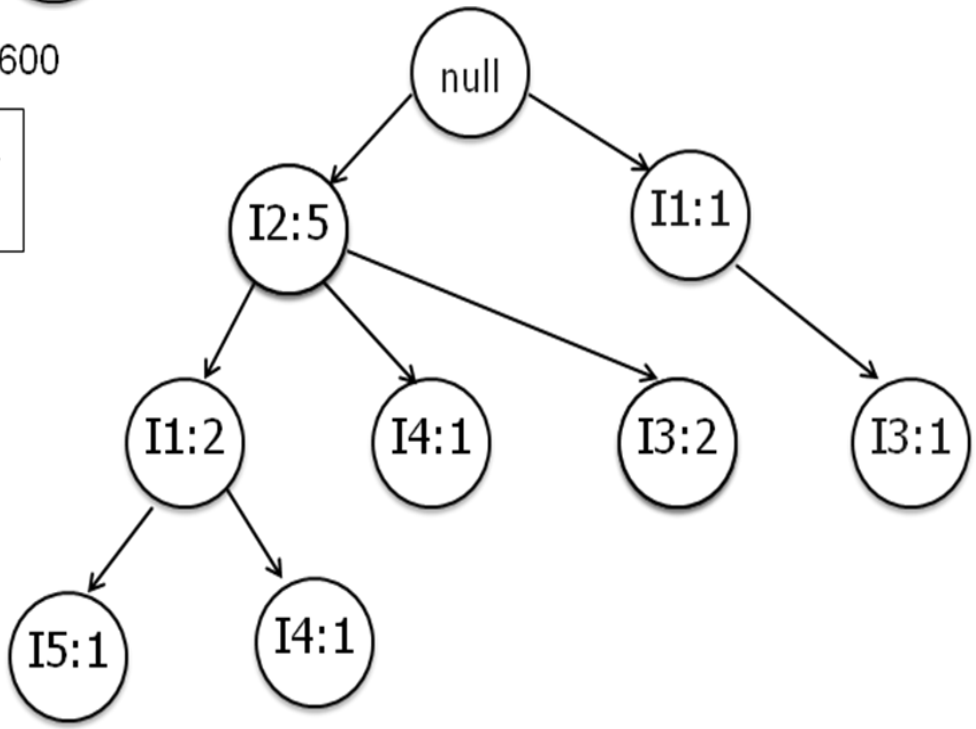
Đọc :T500

I1, I3



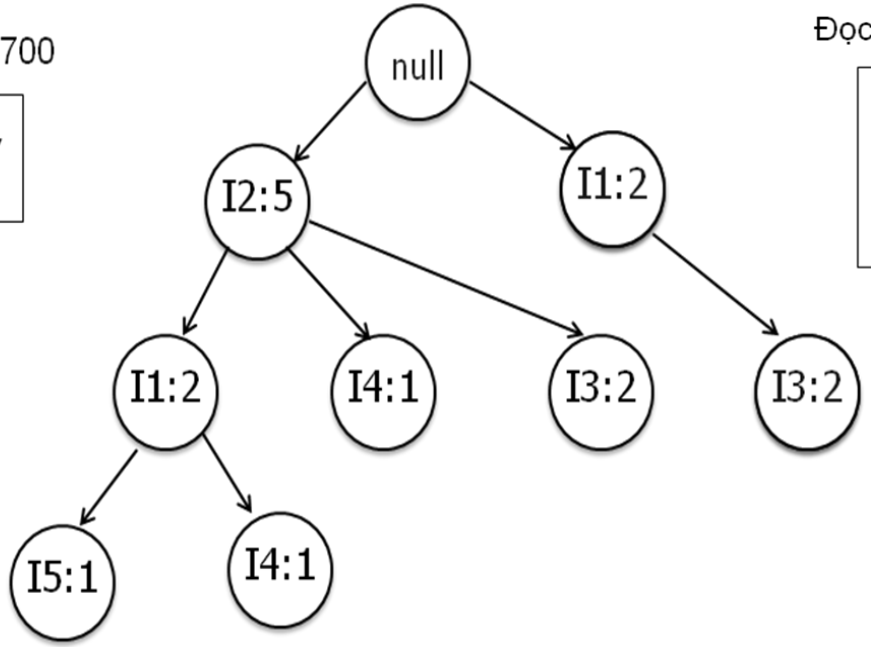
Đọc :T600

I2, I3



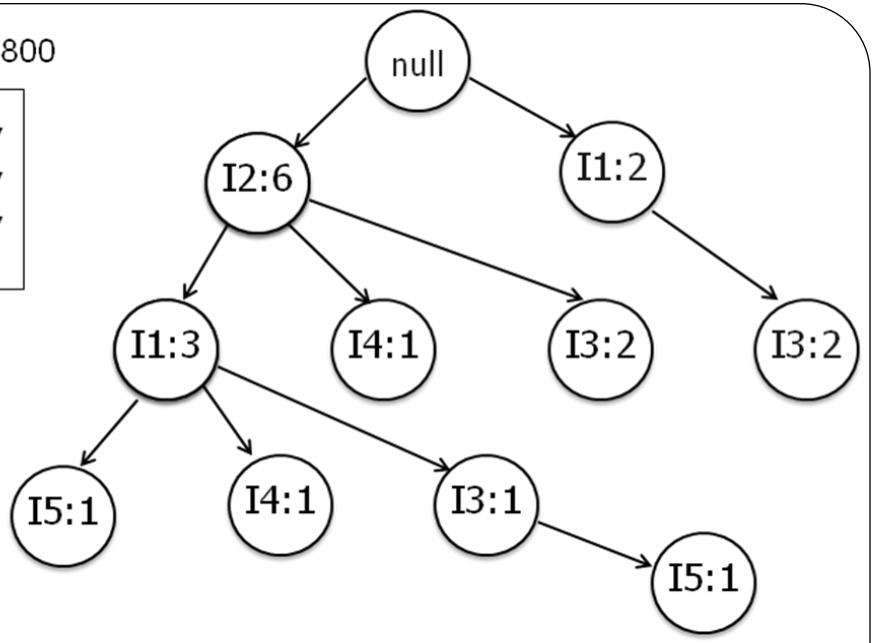
Đọc :T700

I1,
I3



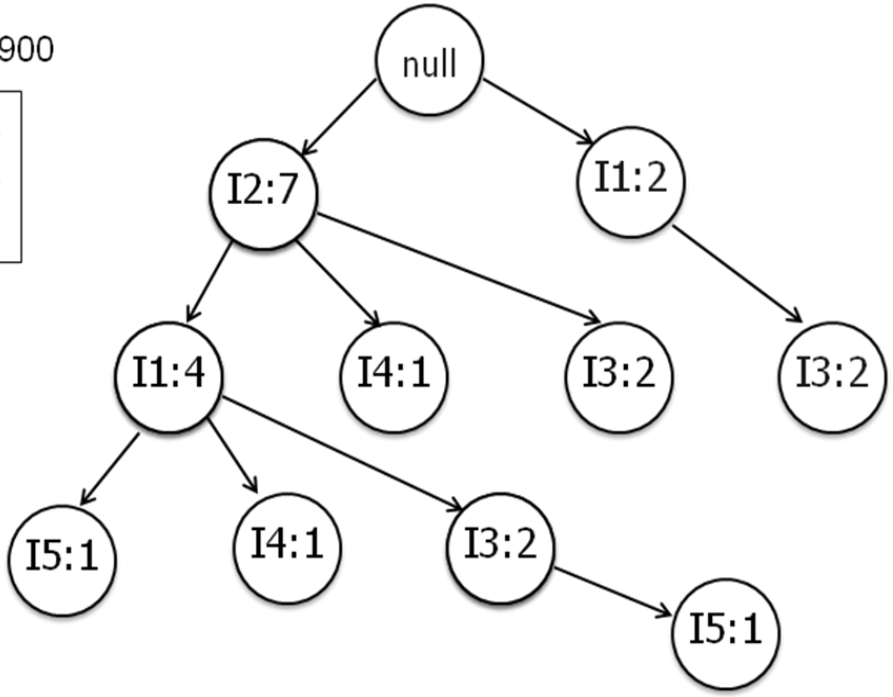
Đọc :T800

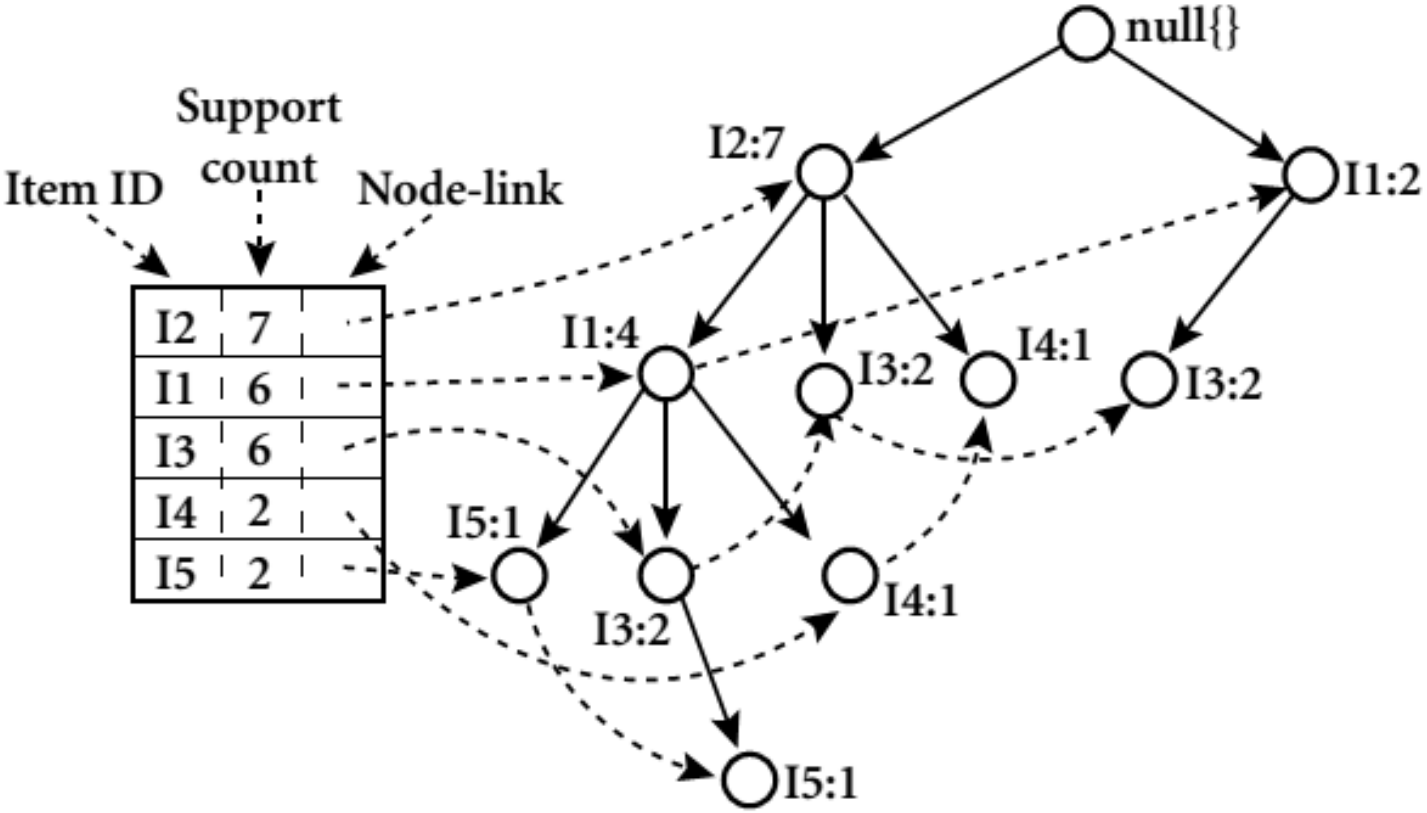
I2,
I1,
I3,
I5



Đọc :T900

I2,
I1,
I3





Item	*Conditional Pattern Base*	*Conditional FP-tree*	*Frequent Patterns Generated*
I5	{{I2, I1: 1}, {I2, I1, I3: 1}}	⟨I2: 2, I1: 2⟩	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{{I2, I1: 1}, {I2: 1}}	⟨I2: 2⟩	{I2, I4: 2}
I3	{{I2, I1: 2}, {I2: 2}, {I1: 2}}	⟨I2: 4, I1: 2⟩, ⟨I1: 2⟩	{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}
I1	{{I2: 4}}	⟨I2: 4⟩	{I2, I1: 4}

Q & A