TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA HỆ THỐNG THÔNG TIN

Tài liệu bài giảng:

KHAI THÁC DỮ LIỆU – IS252

Chương 3:

TẬP PHỔ BIẾN & LUẬT KẾT HỢP

ThS. Dương Phi Long – Email: longdp@uit.edu.vn

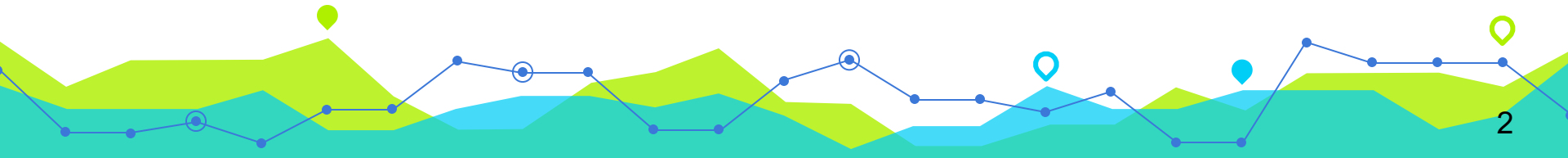
NỘI DUNG BÀI HỌC

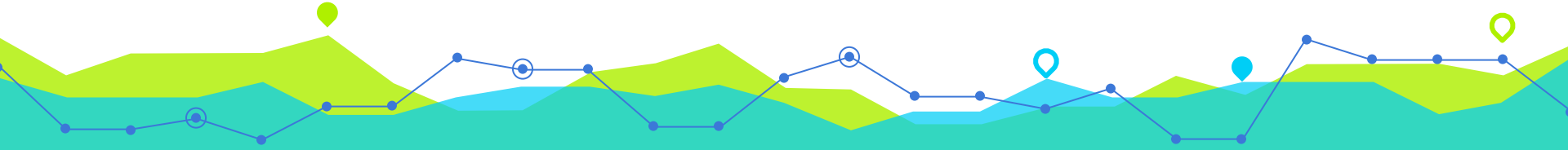
01 Các khái niệm cơ bản

02 Thuật toán Apriori

03 Thuật toán FP-Growth

04 Độ đo tính lý thú của luật kết hợp





1 Các khái niệm cơ bản

1. Mẫu phổ biến
2. CSDL giao dịch
3. Độ phổ biến và tập phổ biến
4. Tập phổ biến tối đại
5. Tập phổ biến đóng
6. Luật kết hợp
7. Bài toán khám phá Luật kết hợp

Đặt vấn đề

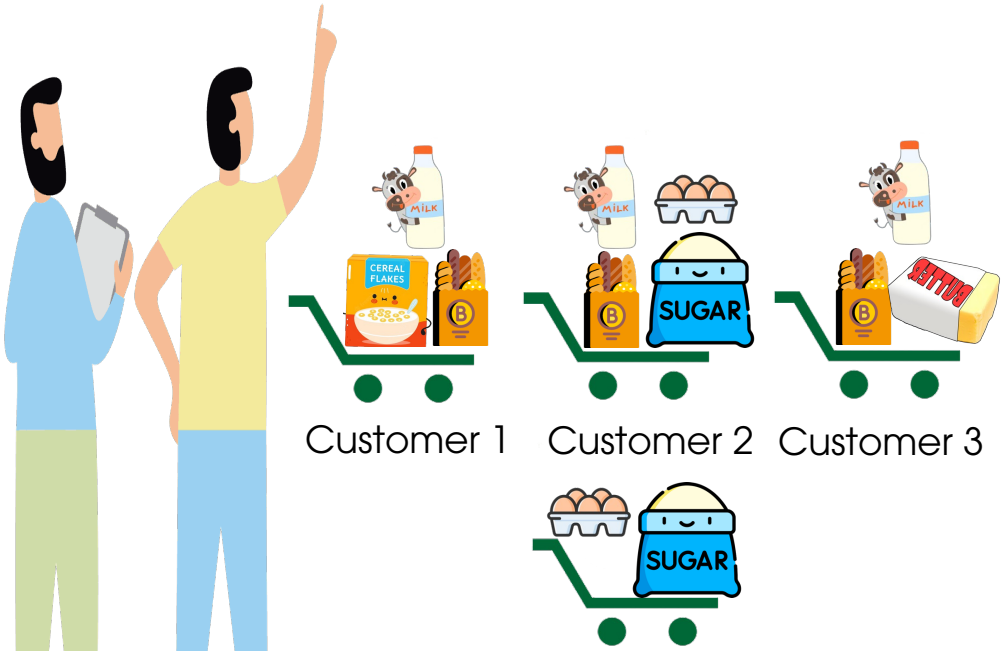
Which items are frequently purchased together by customers?

Will keeping chips and wine together increase sale of chips?

Should I keep cheese and bread side by side as well?

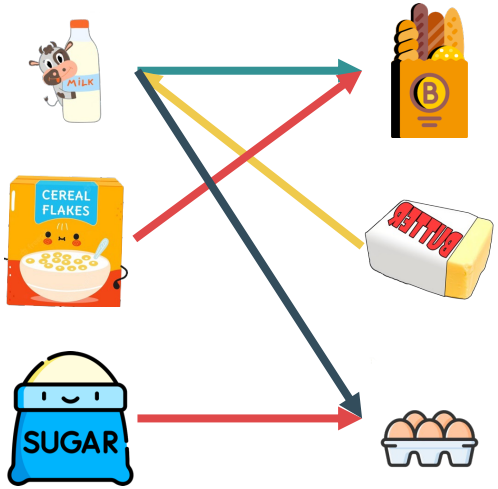
Đặt vấn đề

Which items are frequently purchased together by customers?



Customer 1 Customer 2 Customer 3

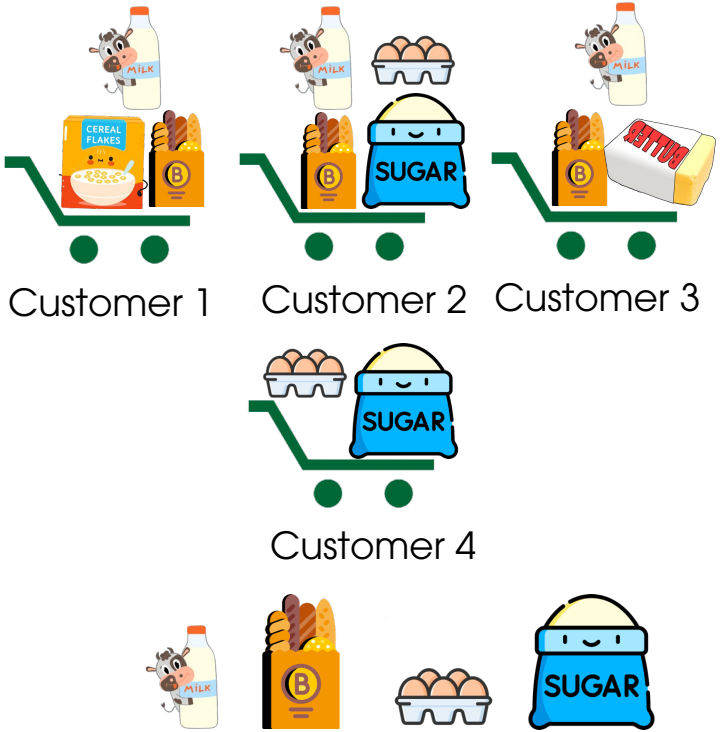
Customer 4



100% of people who purchased milk also purchased bread ???

1. Mẫu phổ biến (Frequent Pattern)

- **Mẫu phổ biến:** là mẫu (tập các item, chuỗi con, cấu trúc con, đồ thị con, ….) xuất hiện thường xuyên trong tập dữ liệu (Agrawal, 1993)
- **Mục đích:** Tìm các hiện tượng thường xuyên xảy ra trong dữ liệu
- **Ứng dụng:**
  - Phân tích CSDL bán hàng
  - Quảng cáo, phân tích chiến dịch bán hàng, Web log, chuỗi DNA, …



2. CSDL giao dịch (Transaction database)

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- Hạng mục (Item): mặt hàng, giá trị thuộc tính
- Tập các hạng mục (itemset)

$$I = \{i_1, i_2, \dots, i_m\}$$

- Tập k hạng mục (k-itemset)

$$X = \{x_1, x_2, \dots, x_k\}$$

2. CSDL giao dịch (Transaction database)

Tid	Items bought
10	Beer, Nuts, Diaper
20	Beer, Coffee, Diaper
30	Beer, Diaper, Eggs
40	Nuts, Eggs, Milk
50	Nuts, Coffee, Diaper, Eggs, Milk

- Giao dịch (transaction): tập các item mua trong 1 giỏ.
- Giao dịch t: tập các item $I_t \subseteq I$
- CSDL giao dịch: tập các t

$$D = \{t_1, t_2, \dots, t_n\}$$

$$t_i = \{item_{i1}, item_{i2}, \dots, item_{ik}\}$$

với $item_{ij} \in I$

2. CSDL giao dịch (Transaction database)

Tid	Items bought
1	Milk, Bread, Eggs
2	Bread, Sugar
3	Bread, Cereal
4	Milk, Bread, Sugar
5	Milk, Cereal
6	Bread, Cereal
7	Milk, Cereal
8	Milk, Bread, Cereal, Eggs
9	Milk, Bread, Cereal

Biến đổi CSDL về dạng nhị phân

Items:
A = Milk
B = Bread
C = Cereal
D = Sugar
E = Eggs

Tid	A	B	C	D	E
1	1	1	0	0	1
2	0	1	0	1	0
3	0	1	1	0	0
4	1	1	0	1	0
5	1	0	1	0	0
6	0	1	1	0	0
7	1	0	1	0	0
8	1	1	1	0	1
9	1	1	1	0	0

3. Độ phổ biến và tập phổ biến

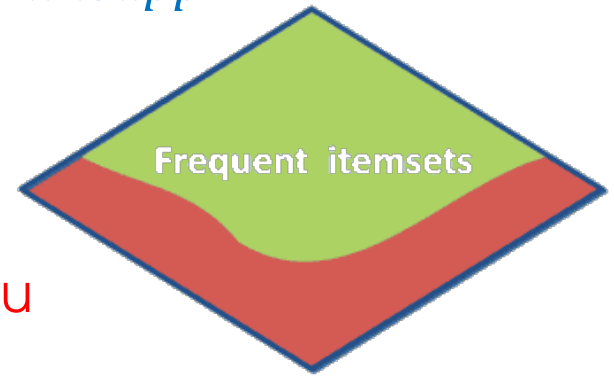
- **Độ phổ biến (supp)** của tập các item X trong CSDL D:

$$Supp(X) = \frac{Count(X)}{|D|} \quad (1)$$

$Count(X)$: số các giao dịch chứa X
$|D|$: tổng số các giao dịch có trong D

- **Tập phổ biến S:** tập các item có $supp(S) \geq minsupp$

$minsupp$: độ phổ biến tối thiểu
(do người dùng xác định)



- **Tính chất tập phổ biến:**

Tất cả các tập con của tập phổ biến đều là tập phổ biến

3. Độ phổ biến và tập phổ biến

VD1: I = {Beer, Bread, Jelly, Milk, Butter}, Minsupp = 60%

Tính độ phổ biến và xác định các tập sau có phải là tập phổ biến?

- X1 = {Bread, Butter}

  Count (X1) = 3, |D| = 5

  → supp(X) = 60% ≥ minsupp

  → X1 là tập phổ biến
- X2 = {Bread}
- X3 = {Butter}
- X4 = {Milk}
- X5 = {Milk, Bread}

Tid	Items
t1	Bread, Jelly, Butter
t2	Bread, Butter
t3	Bread, Milk, Butter
t4	Beer, Bread
t5	Beer, Jelly

4. Tập phổ biến tối đại (Max-Pattern)

- X là tập phổ biến tối đại:
  - X là tập phổ biến
  - Và không tồn tại tập phổ biến nào bao nó

- **VD2:** Minsupp = 2
  - {B, C, D, E}: tập phổ biến tối đại
  - {A, C, D}: tập phổ biến tối đại
  - {B, C, D}: không là tập phổ biến tối đại

Tid	Items
t1	A, B, C, D, E
t2	B, C, D, E
t3	A, C, D, F

(Bayardo – SIGMOD'98)

5. Tập phổ biến đóng (Close-Pattern)

- X là tập phổ biến đóng:
  - X là tập phổ biến
  - Và không tồn tại tập nào bao nó cùng độ phổ biến với nó

- **VD3:** Minsupp = 2
  - {A, B}: tập phổ biến đóng
  - {A, B, D}: tập phổ biến đóng
  - {A, B, C}: tập phổ biến đóng

Pasquier, ICDT'99

Tid	Items
t1	A, B, C
t2	A, B, C
t3	A, B, D
t4	A, B, D
t5	C, E, F

6. Luật kết hợp (Association Rules)

- Có dạng: $\boldsymbol{X} \rightarrow \boldsymbol{Y}$ với $X, Y \subset I$ và $X \cap Y = \emptyset$
- Được đánh giá dựa trên 2 độ đo:
  - **Độ phổ biến (support)**

$$supp(X \rightarrow Y) = P(X \cup Y) = supp(X \cup Y) \quad (2)$$

  - **Độ tin cậy (confidence)**

$$conf(X \rightarrow Y) = P(Y \mid X) = \frac{P(X \cup Y)}{P(X)} = \frac{supp(X \cup Y)}{supp(X)} \quad (3)$$

- VD: Bread → Butter (supp = 60%, conf = 75%)

7. Bài toán khám phá Luật kết hợp

- Cho minsupp và minconf (do người dùng xác định)
- Cho tập items $I = \{i_1, i_2, \dots, i_m\}$ và CSDL $D = \{t_1, t_2, \dots, t_n\}$ với $t_i = \{i_{i1}, i_{i2}, \dots, i_{ik}\}$ và $i_{ij} \in I$
- **Bài toán khám phá Luật kết hợp:** tìm tất cả các luật **X → Y** (X, Y ⊂ I và X ∩ Y = ∅) thỏa mãn cả 2:
  - supp (X → Y) ≥ minsupp
  - conf (X → Y) ≥ minconf

7. Bài toán khám phá Luật kết hợp

Các bước khám phá Luật kết hợp:

- **B1:** Tìm tất cả các tập phổ biến (thỏa minsupp)
- **B2:** Xây dựng luật từ các tập phổ biến
  - Với mỗi tập phổ biến S: tìm tập con khác rỗng của S
  - Với mỗi tập tập con khác rỗng A của S:

    Luật A → (S – A) là luật kết hợp khi:

$$conf\ (A\ \rightarrow\ (S - A)) = \frac{supp(S)}{supp(A)} \geq minconf$$

7. Bài toán khám phá Luật kết hợp

VD4:

Tid	Items
t1	A, B, C
t2	A, C
t3	A, D
t4	B, E, F

Minsupp = 50%, Minconf = 80%

Tập phổ biến	Supp
A	75%
B	50%
C	50%
A, C	50%

- **Luật A → C:**

  supp (A → C) = supp ({A} ∪ {C}) = 50%

  conf (A → C) = supp ({A} ∪ {C}) / supp ({A}) = 66.6% (Loại)

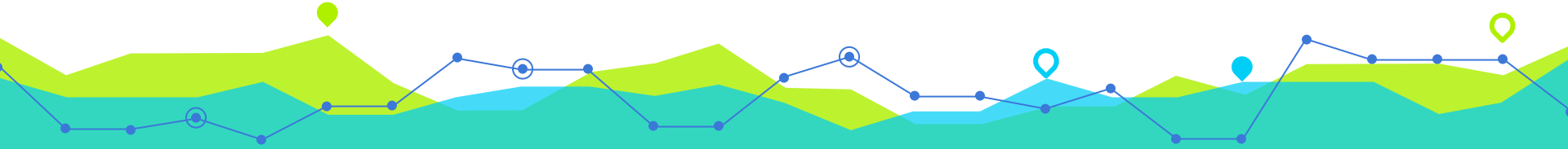
- **Luật C → A:**

  supp (C → A) = supp ({C} ∪ {A}) = 50%

  conf (C → A) = supp ({C} ∪ {A}) / supp ({C}) = 100% (Thỏa)

7. Bài toán khám phá Luật kết hợp

- Đưa về bài toán tìm tập phổ biến.
- Một số thuật toán:
  - Tìm kiếm theo chiều rộng: Thuật toán Apriori (Agrawal & Srikant @VLDB'94)
  - Phát triển mẫu: FP-Growth (Han, Pei & Yin @SIGMOD'00)
  - Tìm kiếm theo dạng dữ liệu dọc: Thuật toán Charm (Zaki & Hsiao @SDM'02) , ECLAT (Zaki et al. @KDD'97)



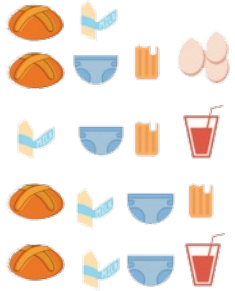
2 Thuật toán Apriori

1. Cách tiếp cận
2. Các bước thực hiện
3. Mã giả
4. Thách thức và cải tiến

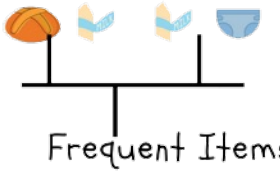
1. Apriori: Cách tiếp cận

Input

CSDL D
minsupp



Output



Các tập phổ biến

1. Apriori: Cách tiếp cận

- **Cách tiếp cận:** Tạo và thử nghiệm các tập ứng viên
- **Nguyên tắc loại bỏ Apriori:** Nếu không phải là tập phổ biến thì tập bao của nó cũng không phổ biến. (Agrawal & Srikant @VLDB'94,. @ KDD'94)

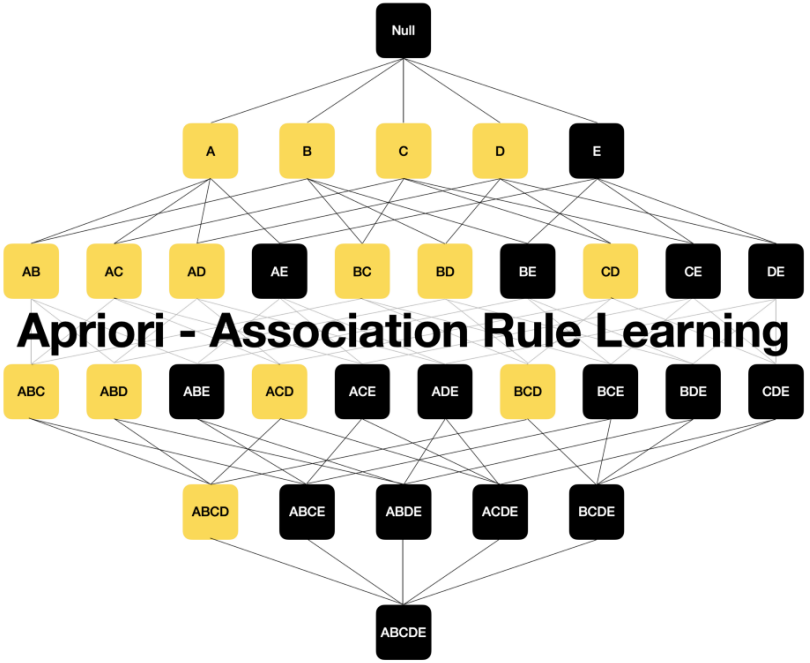
APRIORI

-An algorithm behind
"You may also like"

2. Apriori: Các bước thực hiện

Các bước thực hiện:

- Tìm tất cả các tập phổ biến 1-item
- Tạo các tập ứng viên k-item từ các tập phổ biến 1-item
- Kiểm tra độ phổ biến của các tập ứng viên. Loại các tập ứng viên không phổ biến
- Dừng khi không tạo được tập phổ biến hay tập ứng viên.



2. Apriori: Các bước thực hiện

VD5: Tạo và loại tập ứng viên

- Giả sử L3 = {{1, 2, 3}, {1, 2, 4}, {1, 3, 4}, {1, 3, 5}, {2, 3, 4}}
- Sau bước kết:

  C4 = {{1, 2, 3, 4}, {1, 3, 4, 5}}
- Sau bước loại bỏ:

  C4= {{1, 2, 3, 4}}

  Vì {1, 4, 5} $\notin$ L3 nên {1, 3, 4, 5} bị loại

Apriori: Ví dụ

VD6: Tìm tập phổ biến, minsup = 2

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

1st scan

C_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

< minsup: loại

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

Apriori: Ví dụ

VD6: Tìm tập phổ biến, minsup = 2

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

Kết hợp

C_2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

2nd scan

C_2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

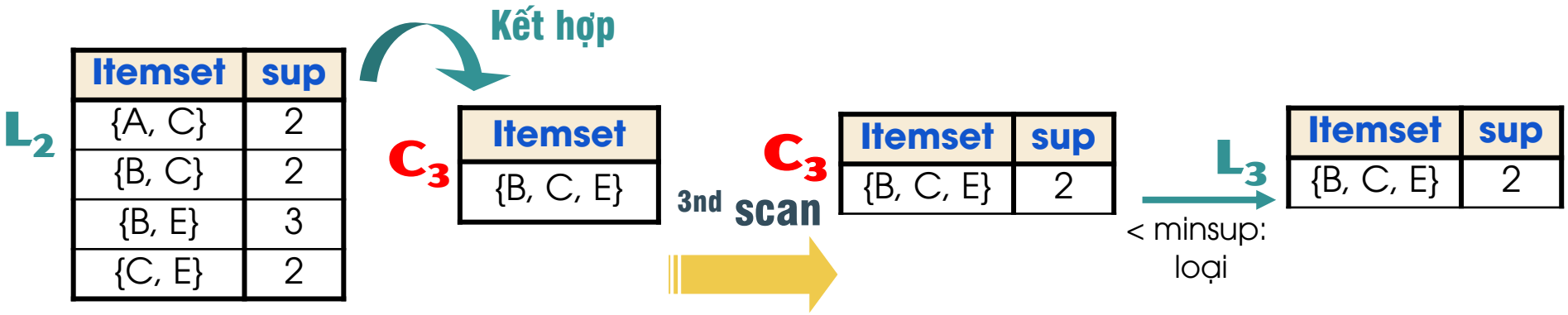
< minsup: loại

L_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

Apriori: Ví dụ

VD6: Tìm tập phổ biến, minsupp = 2



Các tập phổ biến:

$L = L_1 \cup L_2 \cup L_3$

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

3. Apriori: Mã giả

C_k: tập ứng viên k-item L_k: tập phổ biến k-item L_1: tập phổ biến 1-item

```
for (k = 1; Lk != ∅; k++) {
  Ck+1 = candidates_generate (Lk ); // tạo tập ứng viên (k+1)-item
  for each transaction t  ∈ D {  // duyệt CSDL D để tính supp
        Ct = subset(Ck+1 ,t); // lấy ra tập con của t là ứng viên
          for each candidate_set c ∈ Ct
                  c.count ++ ;}
  Lk+1  = { c ∈ Ck+1  |  c,count >= minsup}
}
return L = L ∪k Lk;
```

3. Apriori: Mã giả

candidates_generate (L_k)**:** Tạo tập ứng viên (k+1)-item

Gồm 2 bước: kết và loại bỏ

for each itemset $l_1 \in L_k$ {

for each itemset $l_2 \in L_k$

if($l_1(1) = l_2(1) \wedge \ldots \wedge l_1(k-1) = l_2(k-1) \wedge l_1(k) = l_2(k)$):

{ $c = l_1 \bowtie l_2$; // B1: Kết Lk với chính nó

if has_infrequent_subset(c,L_k):

Delete c; //B2: Xóa các ứng viên không có lợi

else Add c vào C_{k+1};

} return C_{k+1}

3. Apriori: Mã giả

has_infrequent_subset (c, L_k): kiểm tra từng tập con k-item của tập ứng viên c (k+1)-item có thuộc tập phổ biến L_k *k-item không?*

```
for each k-item subset s ∈ c
  if(s ∉ Lk):
      return True;
  return False;
```

4. Apriori: Thách thức và cải tiến

- **Thách thức:**
  - Phải duyệt CSDL nhiều lần
  - Số lượng tập ứng viên rất lớn
  - Thực hiện việc tính độ phổ biến nhiều

  VD: Tìm tập phổ biến 100 items: Số lần duyệt CSDL 100.

  Số lượng tập ứng viên 2^{100} -1
- **Cải tiến: Ý tưởng chung**
  - Giảm số lần duyệt CSDL
  - Giảm số lượng tập ứng viên
  - Tính độ phổ biến thuận tiện hơn

Một số kỹ thuật cải tiến Apriori

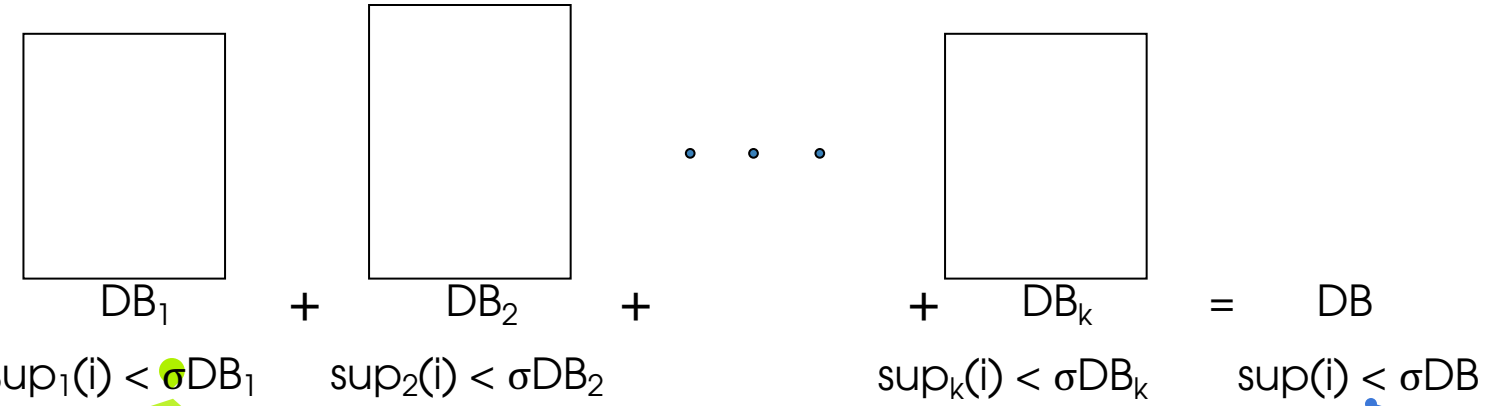
- **Chia để trị:** A. Savasere, E. Omiecinski, and S. Navathe. An efficient algorithm for mining association in large databases. VLDB'95
  - Chia CSDL thành phân hoạch
  - Tìm tập phổ biến cục bộ trong từng phân hoạch và tổ hợp
- **Hàm băm (Hashing):** J. Park, M. Chen, and P. Yu. An effective hash-based algorithm for mining association rules. SIGMOD'95
  - Băm các tập ứng viên k-item vào các giỏ
  - Tập ứng viên k-item tương ứng có độ phổ biến < minsupp sẽ bị loại

Một số kỹ thuật cải tiến Apriori

- **Lấy mẫu (Sampling):** H. Toivonen. Sampling large databases for association rules. VLDB'96
  - Chọn mẫu từ CSDL lớn và tìm tập phổ biến trên mẫu, kiểm tra bao đóng của các tập phổ biến
- **Giảm số lượng scan giao dịch:** S. Brin R. Motwani, J. Ullman, S. Tsur. Dynamic itemset counting and implication rules for market basket data, SIGMOD'97
  - Sử dụng dàn (lattice)

Partition: Scan Database Only Twice

- Any itemset that is potentially frequent in DB must be frequent in at least one of the partitions of DB
  - Scan 1: partition database and find local frequent patterns
  - Scan 2: consolidate global frequent patterns
- A. Savasere, E. Omiecinski and S. Navathe, VLDB'95



DHP: Reduce the Number of Candidates

- A *k*-itemset whose corresponding hashing bucket count is below the threshold cannot be frequent
  - Candidates: a, b, c, d, e
  - Hash entries
    - {ab, ad, ae}
    - {bd, be, de}
    - …
  - Frequent 1-itemset: a, b, d, e
  - ab is not a candidate 2-itemset if the sum of count of {ab, ad, ae} is below support threshold

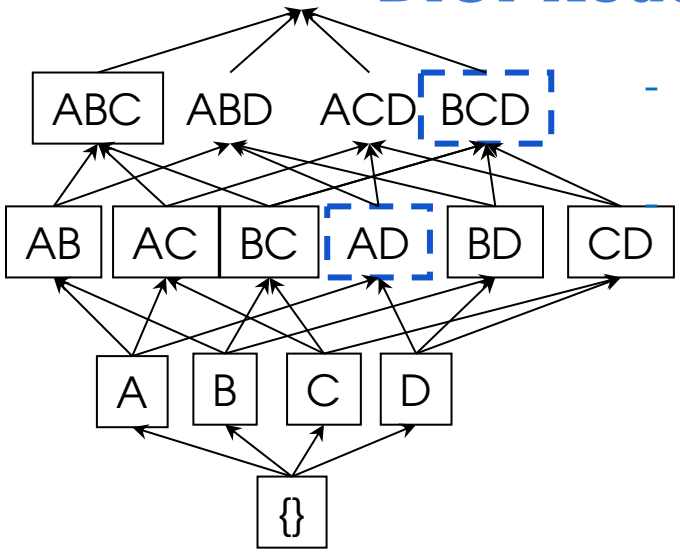
count	itemsets
35	{ab, ad, ae}
88	{bd, be, de}
. . .	. . .
102	{yz, qs, wt}

- J. Park, M. Chen, P. Yu. An effective hash-based algorithm, SIGMOD'95

Sampling for Frequent Patterns

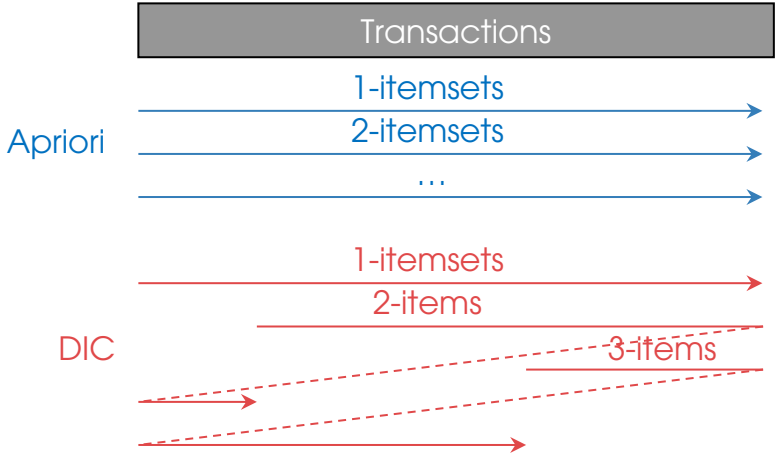
- Select a sample of original database, mine frequent patterns within sample using Apriori
- Scan database once to verify frequent itemsets found in sample, only borders of closure of frequent patterns are checked
  - Example: check abcd instead of ab, ac, …, etc.
- Scan database again to find missed frequent patterns
- H. Toivonen. Sampling large databases for association rules. In VLDB'96

DIC: Reduce Number of Scans



Itemset lattice

- Once both A and D are determined frequent, the counting of AD begins
- Once all length-2 subsets of BCD are determined frequent, the counting of BCD begins



S. Brin R. Motwani, J. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket data. *SIGMOD'97*

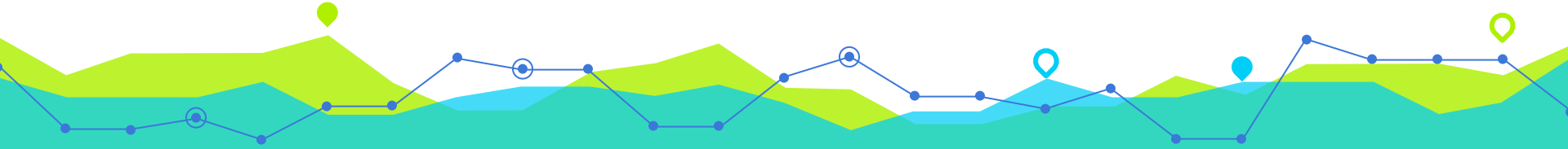
Apriori: Bài tập

BT04

Cho CSDL giao dịch:

1. Sử dụng thuật toán Apriori tìm các tập phổ biến với minsupp = 22%
2. Liệt kê các tập phổ biến tối đại và tập phổ biến đóng
3. Tìm tất cả các luật kết hợp thỏa mãn
    a. Minconf = 50%
    b. Minconf = 70%

Tid	Items
1	M1, M2, M5
2	M2, M4
3	M2, M3
4	M1, M2, M4
5	M1, M3
6	M2, M3
7	M1, M3
8	M1, M2, M3, M5
9	M1, M2, M3



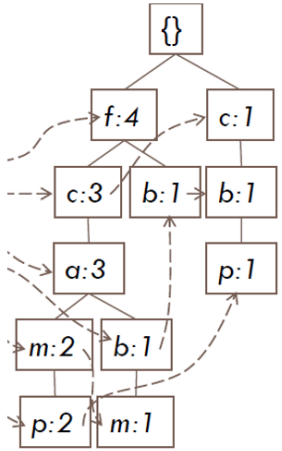
3 Thuật toán FP-Growth

1. Cách tiếp cận
2. Các bước thực hiện

1. FP-Growth: Cách tiếp cận

- Tìm kiếm theo chiều sâu
- Khai thác tập phổ biến KHÔNG sử dụng hàm tạo ứng viên
- Nén CSDL thành cấu trúc cây FP (Frequent Pattern)
- Duyệt đệ quy cây FP để tạo mẫu phổ biến

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



FP Tree

J. Han, J. Pei, and Y. Yin, @SIGMOD'00

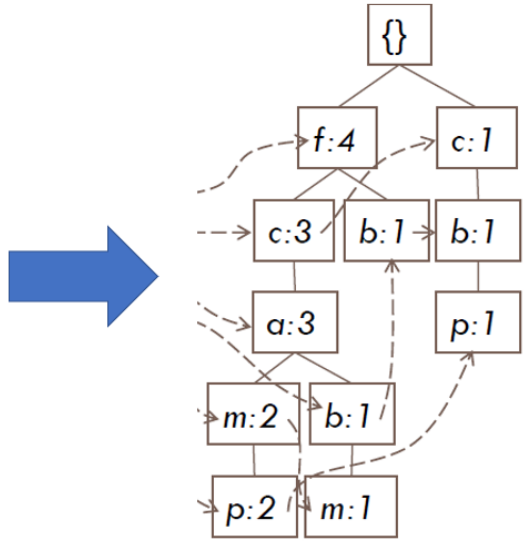
2. FP-Growth: Các bước thực hiện

- **Các bước thực hiện:**
  - **B0:** Thiết lập cây FP
  - **B1:** Thiết lập cơ sở mẫu điều kiện (conditional pattern bases) cho mỗi item phổ biến (mỗi node trên cây FP)
  - **B2:** Thiết lập cây FP điều kiện (conditional FP tree) từ mỗi cơ sở mẫu điều kiện
  - **B3:** Khai thác đệ quy cây FP điều kiện và phát triển mẫu phổ biến cho đến khi cây FP điều kiện chỉ chứa 1 đường duy nhất. Tạo ra tất cả các tập phổ biến

Bước 0. Thiết lập cây FP

- **B0.1:** Tìm tập phổ biến 1-item
- **B0.2:** Xác định F-list: sắp xếp tập phổ biến theo supp giảm dần
- **B0.3:** Sắp xếp CSDL theo F-list. Duyệt CSDL và thiết lập cây FP

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



Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.1:** Tập phổ biến 1-item thỏa minsupp

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

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

- **B0.2:** F-list = f-c-a-b-m-p

Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

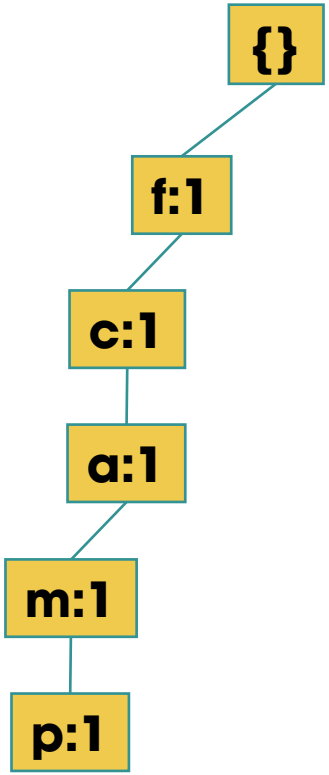
Tid	Items	Frequent-items (ordered)
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o, w}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3 Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Tid	Items	Frequent-items (orderd)
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, w}	{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}

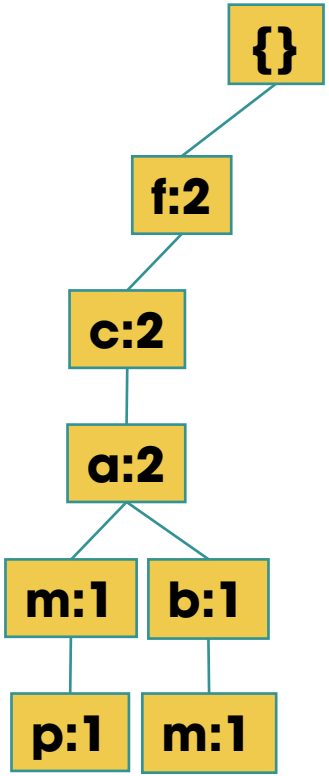


Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Tid	Items	Frequent-items (orderd)
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, w}	{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}

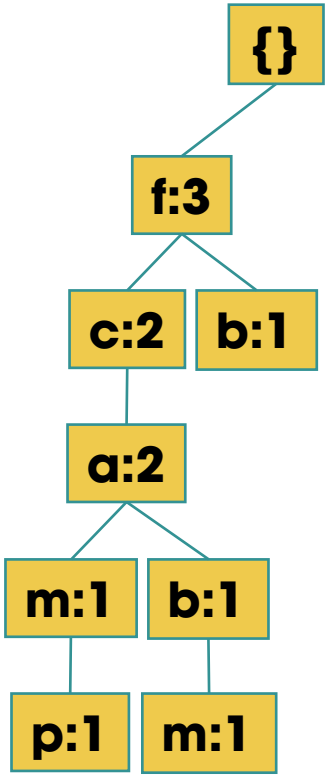


Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Tid	Items	Frequent-items (orderd)
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, w}	**{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}

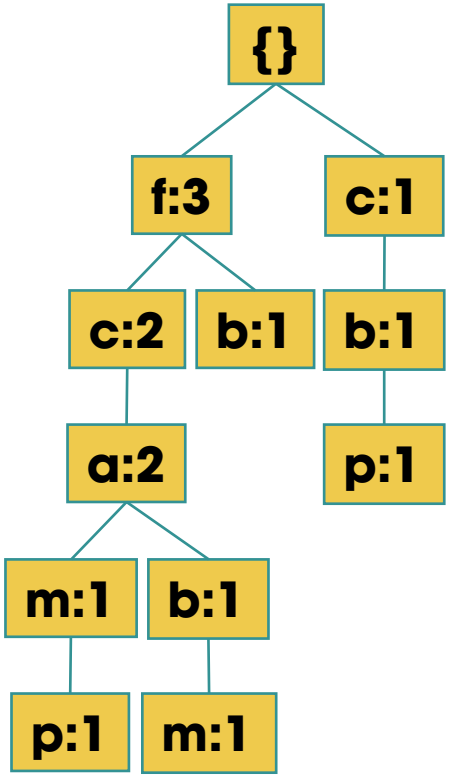


Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Tid	Items	Frequent-items (orderd)
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, w}	{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}

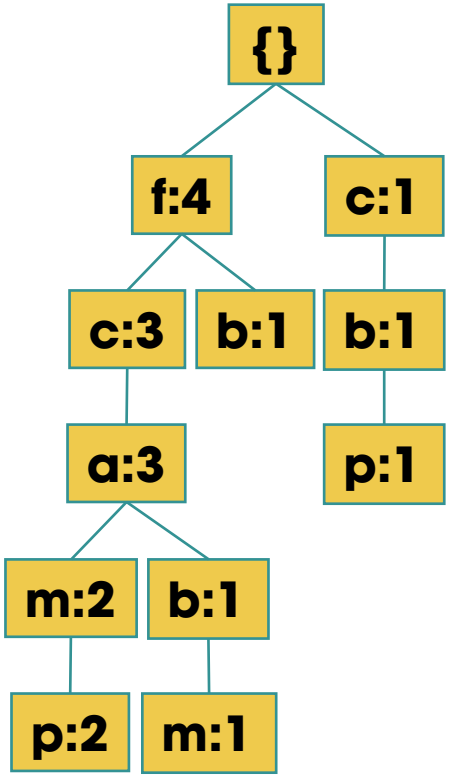


Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Tid	Items	Frequent-items (orderd)
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, w}	{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}**

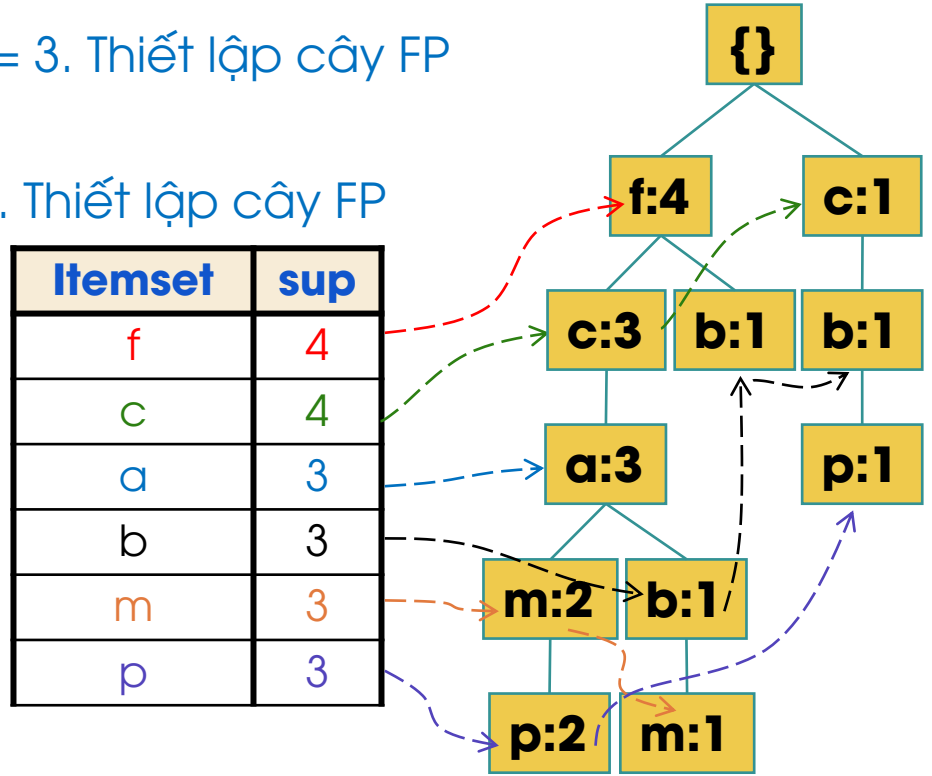


Bước 0. Thiết lập cây FP

VD7: Cho CSDL sau và minsupp = 3. Thiết lập cây FP

- **B0.2:** F-list = f-c-a-b-m-p
- **B0.3:** Sắp xếp CSDL theo F-list. Thiết lập cây FP

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3



FP-Growth: Bài tập

BT05

Tid	Items
1	B, A, K
2	K, B, C, A
3	A, D, M, B
4	D, A, B, E
5	A, K, C
6	A, B, C
7	M, B, C, E
8	B, C, D
9	B, E
10	A, E, M, K
11	A, C, E, M
12	A, D, E

Cho CSDL sau

1. Xây dựng cây FP với minsupp = 25%
2. Nếu minsupp = 40% thì cây FP sẽ thay đổi như thế nào?

Bước 1. Thiết lập Cơ sở mẫu điều kiện

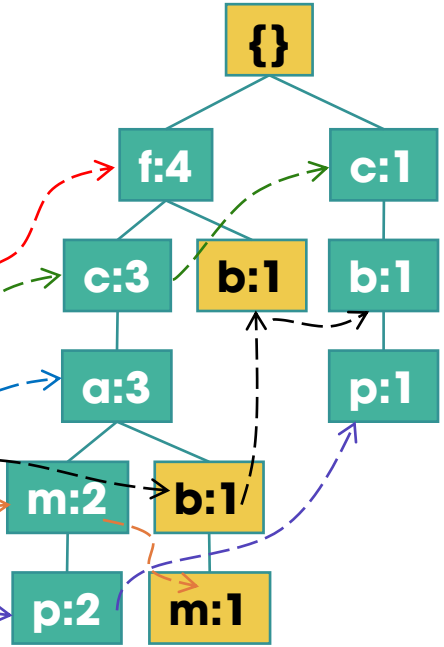
Duyệt các mẫu phổ biến bắt đầu từ mẫu phổ biến cuối cùng của cây FP cho đến mẫu trên cùng:

- **B1.1:** Duyệt cây FP theo kết nối của mỗi mẫu phổ biến
- **B1.2:** Gom tất cả đường dẫn tiền tố biến đổi của mẫu để tạo cơ sở mẫu điều kiện

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: p

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

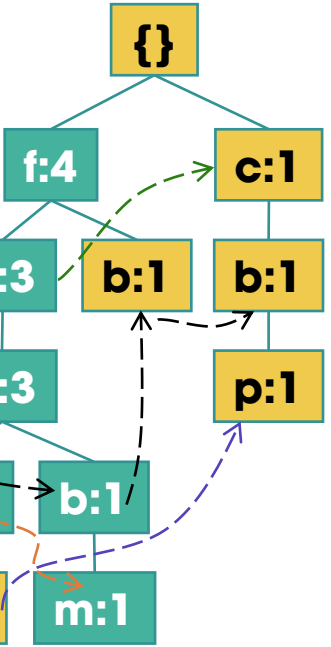


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: m

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

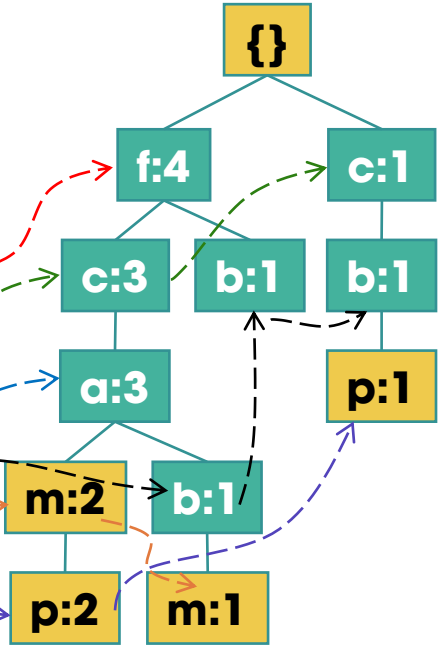


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	**{(fca: 2), (fcab: 1)}**

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: b

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

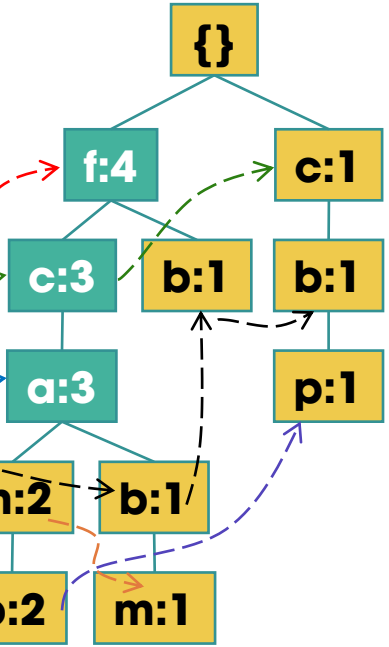


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	**{(fca: 1), (f: 1), (c: 1)}**

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: a

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

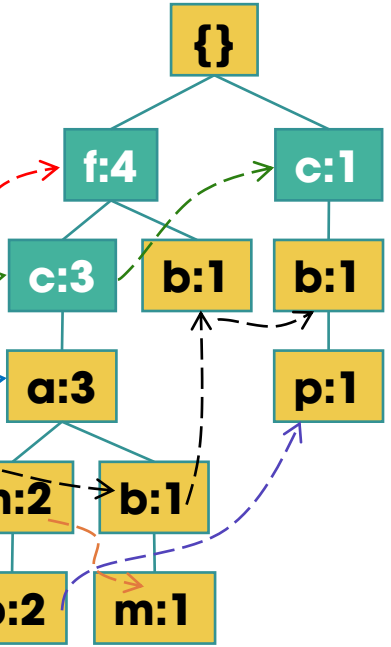


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	{(fca: 1), (f: 1), (c: 1)}
a	**{(fc: 3)}**

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: c

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

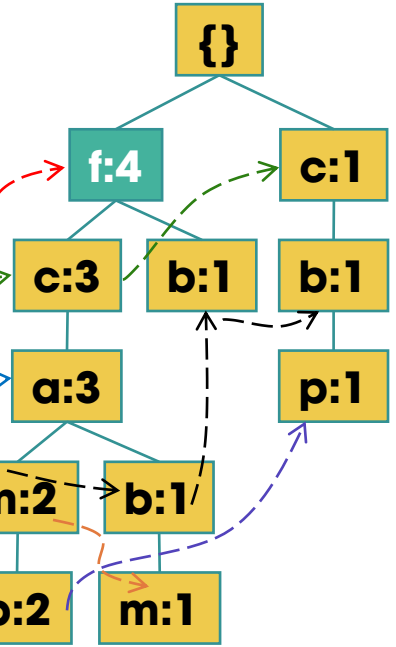


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	{(fca: 1), (f: 1), (c: 1)}
a	{(fc: 3)}
c	**{(f: 3)}**

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Bắt đầu từ mẫu phổ biến cuối của cây FP: f

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3

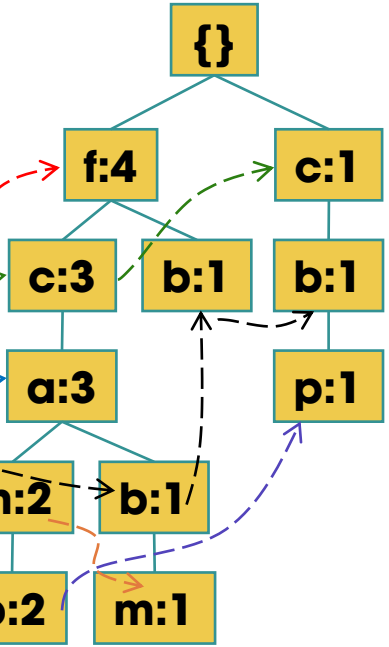


Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	{(fca: 1), (f: 1), (c: 1)}
a	{(fc: 3)}
c	{(f: 3)}
f	**{}**

Bước 1. Thiết lập Cơ sở mẫu điều kiện

VD7 (tt): Thiết lập Cơ sở mẫu điều kiện: XONG

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3



Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	{(fca: 1), (f: 1), (c: 1)}
a	{(fc: 3)}
c	{(f: 3)}
f	{}

Bước 2. Thiết lập Cây FP điều kiện

Với mỗi cơ sở mẫu điều kiện:

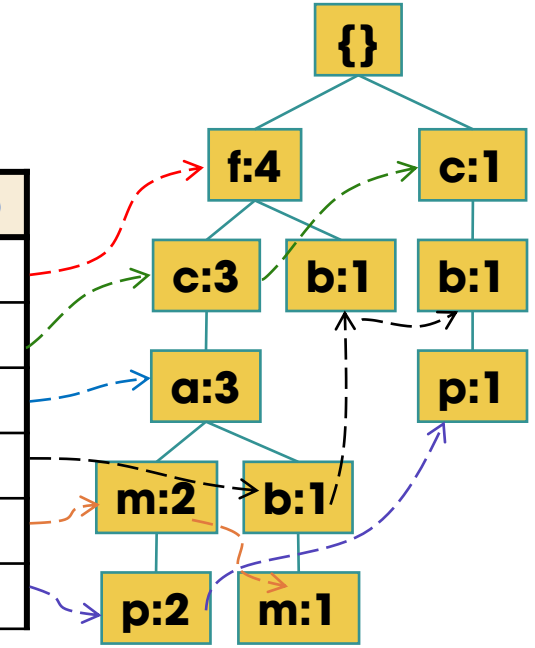
- **B2.1:** Đếm số lượng mỗi mẫu trong cơ sở mẫu. Xác định tập phổ biến của mẫu cơ sở
- **B2.2:** Xây dựng cây FP điều kiện cho tập phổ biến của mẫu cơ sở (tương tự như bước 0).

Lưu ý: Sử dụng minsupp để loại bớt các mẫu phổ biến

Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

Itemset	sup
f	4
c	4
a	3
b	3
m	3
p	3



Item	Cơ sở mẫu điều kiện
p	{(fcam: 2), (cb: 1)}
m	{(fca: 2), (fcab: 1)}
b	{(fca: 1), (f: 1), (c: 1)}
a	{(fc: 3)}
c	{(f: 3)}
f	{}

Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

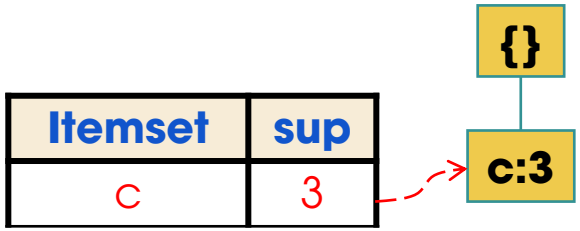
Với cơ sở mẫu điều kiện của **p: {(fcam: 2), (cb: 1)}**

- Đếm số lượng mỗi mẫu trong cơ sở mẫu:

  (f: 2, c: 3, a: 2, m: 2), (c:3, b: 1) với Minsupp = 3

  => (c:3) phổ biến trên cơ sở mẫu điều kiện của **p**

- Thiết lập cây FP cho tập phổ biến của cơ sở mẫu điều kiện của **p**



p-conditional FP-tree

Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

Với cơ sở mẫu điều kiện của **m: {(fca: 2), (fcab: 1)}**

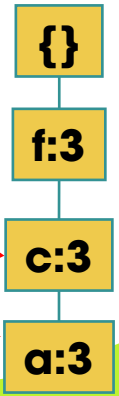
- Đếm số lượng mỗi mẫu trong cơ sở mẫu:

  (f: 3, c: 3, a: 3), (f: 3, c: 3, a: 3, b: 1) với Minsupp = 3

  => (f: 3, c: 3, a: 3) phổ biến trên cơ sở mẫu điều kiện của **m**

- Thiết lập cây FP cho tập phổ biến của cơ sở mẫu điều kiện của **m**

Itemset	sup
f	3
c	3
a	3



m-conditional FP-tree

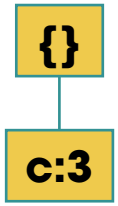
Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

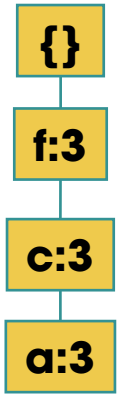
Item	Cơ sở mẫu điều kiện	Cây FP điều kiện
p	{(fcam: 2), (cb: 1)}	{(c: 3)} |p
m	{(fca: 2), (fcab: 1)}	{(f: 3, c: 3, a: 3)} | m
b	{(fca: 1), (f: 1), (c: 1)}	{ }
a	{(fc: 3)}	{(f: 3, c: 3)} | a
c	{(f: 3)}	{(f: 3)} | c
f	{}	{ }

Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

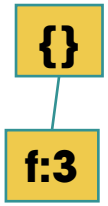


p-conditional FP-tree



m-conditional FP-tree

a-conditional FP-tree



c-conditional FP-tree

Nếu cây có nhánh thì xét tiếp cơ sở mẫu, cây FP điều kiện cho cây đó

Bước 3. Xây dựng tập phổ biến

- Dựa trên nguyên lý mở rộng mẫu phổ biến
- Dựa trên tính chất mở rộng mẫu:

  Giả sử α là tập phổ biến trong CSDL B, B là cơ sở mẫu điều kiện của α, và β là một tập các item trong B.

  $\alpha \cup \beta$ là tập phổ biến $\Leftrightarrow$ β là phổ biến trong B
- VD: "abcdef" là mẫu phổ biến khi và chỉ khi:
  - "abcde" là mẫu phổ biến, và
  - "f" là phổ biến trong các tập giao dịch chứa "abcde"

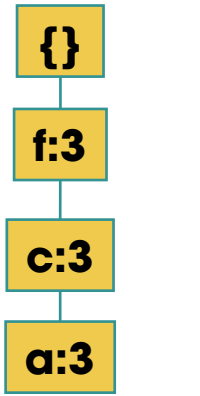
Bước 3. Xây dựng tập phổ biến

Xét từng trường hợp cây FP có điều kiện:

- **TH1: Cây chỉ có đường dẫn đơn:**

  Tập phổ biến: liệt kê tất cả các tổ hợp của đường dẫn con

VD:



m-conditional FP-tree

Tập mẫu phổ biến liên quan đến m:

- m: 3
- fm: 3, cm: 3, am: 3
- fcm: 3, fam: 3, cam: 3
- fcam: 3

Bước 3. Xây dựng tập phổ biến

Xét từng trường hợp cây FP có điều kiện:

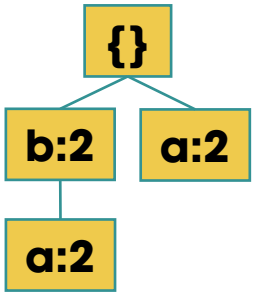
- **TH2: Cây có nhiều nhánh:**

  Thực hiện việc phân chia thành cây có đường dẫn đơn (thiết lập cơ sở mẫu có điều kiện, cây FP điều kiện).

  Xây dựng tập phổ biến cho các cây có đường dẫn đơn vừa tạo.

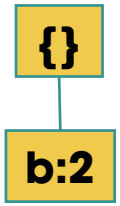
VD:

Minsupp=2



c-conditional FP-tree

Item	CSMĐK	CFPĐK
ac	{(b: 2)}	{(b: 2)} | ac
bc	{ }	{ }



ac-conditional FP-tree

Tập mẫu phổ biến liên quan đến c:
c, bc, bac, ac

Bước 2. Thiết lập Cây FP điều kiện

VD7 (tt): Thiết lập Cây FP điều kiện, minsupp = 3

Tất cả các cây PT điều kiện đều có đường dẫn đơn

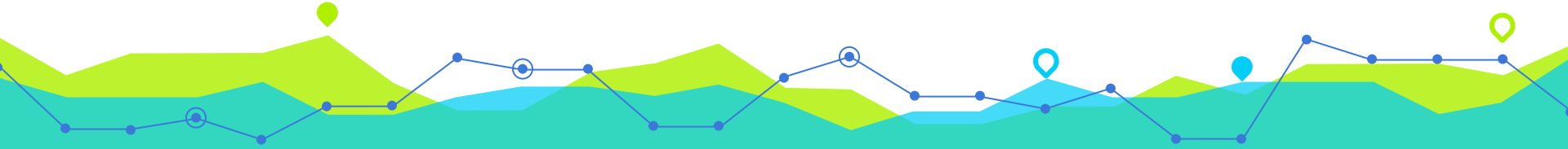
Item	sup	Cây FP điều kiện	Tập phổ biến
p	4	{(c: 3)} |p	p: 4, cp: 3
m	4	{(f: 3, c: 3, a: 3)} | m	m: 4, fm: 3, cm: 3, am: 3, fcm: 3, fam: 3, cam: 3, fcam:3
b	3	{ }	b: 3
a	3	{(f: 3, c: 3)} | a	a: 3, fa: 3, ca: 3, fca: 3
c	3	{(f: 3)} | c	c: 3, fc: 3
f	3	{ }	f: 3

Mở rộng FP-Growth

- Mining closed frequent itemsets and max-patterns
  - CLOSET (DMKD'00), FPclose, and FPMax (Grahne & Zhu, Fimi'03)
- Mining sequential patterns
  - PrefixSpan (ICDE'01), CloSpan (SDM'03), BIDE (ICDE'04)
- Mining graph patterns
  - gSpan (ICDM'02), CloseGraph (KDD'03)
- Constraint-based mining of frequent patterns
  - Convertible constraints (ICDE'01), gPrune (PAKDD'03)

Mở rộng FP-Growth

- Computing iceberg data cubes with complex measures
  - H-tree, H-cubing, and Star-cubing (SIGMOD'01, VLDB'03)
- Pattern-growth-based Clustering
  - MaPle (Pei, et al., ICDM'03)
- Pattern-Growth-Based Classification
  - Mining frequent and discriminative patterns (Cheng, et al, ICDE'07)



4 Độ đo tính lý thú của luật kết hợp

Độ đo tính lý thú

- Luật hay, luật lý thú?
  - Sinh ra quá nhiều luật, có nhiều luật không hay hoặc bị thừa
  - Cần độ đo tính lý thú để hạn chế luật
- Độ đo khách quan:
  - Độ phổ biến (supp) và độ tin cây (conf)
  - Khoảng 20 độ đo khác
- Độ đo chủ quan:
  - Luật kết hợp lý thú nếu là điều mới lạ, gây ngạc nhiên
  - Có khả năng ứng dụng

Độ đo tính lý thú

VD8:

- Trong 5,000 sinh viên:
  - 3,000 chơi bóng rổ
  - 3,750 ăn ngũ cốc
  - 2,000 chơi bóng rổ và ăn ngũ cốc

	Basketball	$\overline{\text{Basketball}}$	Sum
Cereal	2,000	1,750	3,750
$\overline{\text{Cereal}}$	1,000	250	1,250
Sum	3,000	2,000	5,000

- **Chơi bóng rổ → Ăn ngủ cốc** (40%, 66.7%): là sai lầm
  Vì tỷ lệ sv ăn ngủ cốc là 75% > 66.7%

- **Chơi bóng rổ → Không ăn ngủ cốc** (20%, 33.3%): có ý nghĩa thực tiễn hơn dù supp và conf thấp hơn

Độ đo tính lý thú

VD9:

- **Tea → Coffee:**
  - Conf = P(Coffee | Tea)=15/20 = 0.75
  - Nhưng P(Coffee) = 0.9

	Coffee	$\overline{\text{Coffee}}$	Sum
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
Sum	90	10	100

Mặc dù Conf cao nhưng luật làm cho lạc hướng

P(Coffee | $\overline{\text{Tea}}$) = 75/80 = 0.9375

Độ đo tính lý thú

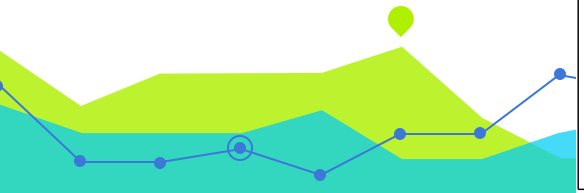
- Cần độ đo sự phụ thuộc, mối tương quan giữa các sự kiện
- Một số độ đo tính lý thú của X → Y
  - **Lift:** $$Lift = \frac{P(Y|X)}{P(Y)} = \frac{P(Y,X)}{P(Y)P(X)} \quad (4)$$
  - **Interest:** $$Interest = \frac{P(X,Y)}{P(X)P(Y)} \quad (5)$$
  - **Piatetsky-Shapiro's:** $$PS = P(X,Y) - P(X)P(Y) \quad (6)$$
  - $\phi - coneffficient$: $$\phi - coneffficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}} \quad (7)$$

- Mua quả óc chó → mua sữa (1%, 80%): là sai lầm nếu 85% khách hàng mua sữa

- Supp và Conf không tốt để chỉ ra mối tương quan

- Hơn 20 độ đo mức độ thú vị đã được đề xuất

(Tan, Kumar, Sritastava @KDD'02)



symbol	measure	range	formula
ϕ	ϕ-coefficient	-1 . . . 1	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
Q	Yule's Q	-1 . . . 1	$\frac{P(A,B)P(\overline{A},\overline{B})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{A},\overline{B})+P(A,\overline{B})P(\overline{A},B)}$
Y	Yule's Y	-1 . . . 1	$\frac{\sqrt{P(A,B)P(\overline{A},\overline{B})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{A},\overline{B})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}}$
k	Cohen's	-1 . . . 1	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
PS	Piatetsky-Shapiro's	-0.25 . . . 0.25	$P(A,B)-P(A)P(B)$
F	Certainty factor	-1 . . . 1	$\max(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)})$
AV	added value	-0.5 . . . 1	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
K	Klosgen's Q	-0.33 . . . 0.38	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$
g	Goodman-kruskal's	0 . . . 1	$\frac{\Sigma_j \max_k P(A_j,B_k)+\Sigma_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-max_k P(B_k)}$
M	Mutual Information	0 . . . 1	$\frac{\Sigma_i\Sigma_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_J)}}{\min(-\Sigma_i P(A_i)\log P(A_i)\log P(A_i), -\Sigma_i P(B_i)\log P(B_i)\log P(B_i))}$
J	J-Measure	0 . . . 1	$\max(P(A,B)\log(\frac{P(B\|A)}{P(B)})+P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}))$ $P(A,B)\log(\frac{P(A\|B)}{P(A)})+P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})$
G	Gini index	0 . . . 1	$\max(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A}[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B}[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]-P(A)^2-P(\overline{A})^2)$
s	support	0 . . . 1	$P(A,B)$
c	confidence	0 . . . 1	$max(P(B\|A), P(A\|B))$
L	Laplace	0 . . . 1	$\max(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2})$
IS	Cosine	0 . . . 1	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
γ	coherence(Jaccard)	0 . . . 1	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
α	all_confidence	0 . . . 1	$\frac{P(A,B)}{\max(P(A),P(B))}$
o	odds ratio	0 . . . ∞	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(\overline{A},B)P(A,\overline{B})}$
V	Conviction	0.5 . . . ∞	$\max(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})})$
λ	lift	0 . . . ∞	$\frac{P(A,B)}{P(A)P(B)}$
S	Collective strength	0 . . . ∞	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
χ^2	χ^2	0 . . . ∞	$\Sigma_i \frac{(P(A_i)-E_i)^2}{E_i}$

Độ đo tính lý thú

VD8: Độ đo tương quan **Interest**

- Interest <1: X, Y tương quan nghịch
- Ngược lại, tương quan thuận

	Basketball	$\overline{\text{Basketball}}$	Sum
Cereal	2,000	1,750	3,750
$\overline{\text{Cereal}}$	1,000	250	1,250
Sum	3,000	2,000	5,000

B: Basketball; C: Cereal

Chơi bóng rổ → Ăn ngủ cốc

$$Interest(B,C) = \frac{P(B,C)}{P(B)P(C)} = \frac{\frac{2{,}000}{5{,}000}}{\frac{3{,}000}{5{,}000} * \frac{3{,}750}{5{,}000}} = 0.89$$

Chơi bóng rổ → Không Ăn ngủ cốc

$$Interest(B,\neg C) = \frac{P(B,\neg C)}{P(B)P(\neg C)} = \frac{\frac{1{,}000}{5{,}000}}{\frac{3{,}000}{5{,}000} * \frac{1{,}250}{5{,}000}} = 1.33$$

Độ đo tính lý thú

VD9: Độ đo tương quan **PS**

	Coffee	$\overline{\text{Coffee}}$	Sum
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
Sum	90	10	100

- Tea → Coffee:

$$PS(Tea, Coffee)$$
$$= P(Tea, Coffee) - P(Tea)P(Coffee)$$
$$= 0.15 - 0.2 * 0.9 = -0.03$$

- ¬Tea → Coffee:

$$PS(\neg Tea, Coffee)$$
$$= P(\neg Tea, Coffee) - P(\neg Tea)P(Coffee)$$
$$= 0.75 - 0.8 * 0.9 = 0.03$$

FP-Growth: Bài tập

BT06

Cho CSDL giao dịch:

1. Sử dụng thuật toán FP-Growth tìm các tập phổ biến với minsupp = 22%
2. Tìm tất cả các luật kết hợp có dạng $X \wedge Y \rightarrow Z$ thỏa mãn Minconf = 100%
3. Tính độ lý thú PS, Lift cho các luật kết hợp đã tìm được ở câu 2

Tid	Items
1	M1, M2, M5
2	M2, M4
3	M2, M3
4	M1, M2, M4
5	M1, M3
6	M2, M3
7	M1, M3
8	M1, M2, M3, M5
9	M1, M2, M3

Tổng kết chương

Các khái niệm cơ bản

1.Mẫu phổ biến
2.CSDL giao dịch
3.Độ phổ biến và tập phổ biến
4.Tập phổ biến tối đại
5.Tập phổ biến đóng
6.Luật kết hợp
7.Bài toán khám phá Luật kết hợp

Thuật toán Apriori

1.Cách tiếp cận
2.Các bước thực hiện
3.Mã giả
4.Thách thức và cải tiến

Tổng kết chương

Thuật toán FP-Growth

1.Cách tiếp cận

2.Các bước thực hiện

Độ đo tính lý thú của luật kết hợp

Tóm tắt

- Bài toán khai thác tập phổ biến và luật kết hợp
- Thuật toán tiêu biểu tìm tập phổ biến: Apriori, FP-Growth
- Độ đo tính lý thu của luật kết hợp
- Vấn đề mở: Phân tích mối kết hợp trong các loại dữ liệu khác: không gian, hình ảnh, đa phương tiện, thời gian thực, …

Bài tập chương 3

3.1. Cho CSDL sau và minsupp=60%, minconf=100%

a. Sử dụng thuật toán Apriori tìm các tập phổ biến, tập phổ biến tối đại và tập phổ biến đóng

b. Tìm tất cả các luật kết hợp có dạng (item1 ^ item2) →item3 và thỏa điều kiện minconf.

Tid	Items
10	H, D, F, B, K
20	K, G, C, D, H
30	A, E, F, C, P, B
40	D, H, A, K, B, C
50	H, B, P, F, G

Bài tập chương 3

3.2. Cho CSDL sau và minsupp=50%, minconf=80%

a. Sử dụng thuật toán Apriori tìm các tập phổ biến, tập phổ biến tối đại và tập phổ biến đóng.

b. Liệt kê luật kết hợp thỏa mãn ngưỡng đã cho và có dạng (item1 $\wedge$ item2) $\Rightarrow$ item3 kèm theo supp, conf của nó.

Tid	Items
100	K, A, D, B, C, I
200	D, A, C, E, B
300	C, A, B, E, D
400	B, A, D, I, K

Bài tập chương 3

3.3. Cho CSDL

a. Xây dựng cây FP với minsupp = 30%

b. Xây dựng cây FP với minsupp = 50%

c. Tính độ phổ biến, độ tin cậy, độ đo interest của các luật sau

(1): A → B

(2): B → C

(3): M → E

Tid	Items
1	M, K, A, B
2	B, C, D, M
3	A, C, D, E, K
4	A, D, M, E
5	A, K, B, C
6	A, B, C, D
7	K, B, C
8	A, B, C, K, M
9	A, M, B ,D
10	B, C, E, M

Bài tập chương 3

3.4. Cho CSDL sau và minsupp=50%, minconf=80%

a. Tìm tất cả các tập phổ biến, tập phổ biến tối đại và tập phổ biến đóng sử dụng thuật toán FP-Growth.

b. So sánh kết quả và tính hiệu quả với cách sử dụng thuật toán Apriori (bài tập 3.2).

Tid	Items
100	K, A, D, B, C, I
200	D, A, C, E, B
300	C, A, B, E, D
400	B, A, D, I, K

THANKS!

Any questions?