

Corrigé-type du TD 3

Exercice 1

Considérer un ensemble de 5 documents $D = \{d1, d2, \dots, d5\}$ et un ensemble de 11 termes $T = \{t1, t2, \dots, t11\}$, un document étant un ensemble de termes. La table suivante montre le contenu des documents de D :

document	termes
d1	{t1, t2, t3, t4, t5, t6}
d2	{t2, t3, t4, t5, t6, t7}
d3	{t1, t4, t5, t8}
d4	{t1, t4, t6, t9, t10}
d5	{t2, t4, t5, t10, t11}

Pour extraire les ensembles de termes fréquents avec un support minimal de 60% :

- 1) Appliquer l'algorithme Apriori sur D
- 2) Appliquer l'algorithme FP-Growth sur D
- 3) Appliquer l'Algorithme ECLAT sur D
- 4) Que peut-on en conclure ?

Réponse

1) $\text{Minsup} = 60\% = \frac{60 \times 5}{100} = 3$ documents

Itemset	sup
{t1}	3
{t2}	3
{t3}	2
{t4}	5
{t5}	4
{t6}	3
{t7}	1
{t8}	1
{t9}	1
{t10}	2
{t11}	1

→

Itemset	sup
{t1}	3
{t2}	3
{t4}	5
{t5}	4
{t6}	3

↓

Itemset	sup
{t1,t4}	3
{t2, t4}	3
{t2, t5}	3
{t4, t5}	4
{t4, t6}	3

←

Itemset	sup
{t1, t2}	1
{t1,t4}	3
{t1, t5}	2
{t1, t6}	2
{t2, t4}	3
{t2, t5}	3
{t2, t6}	2
{t4, t5}	4
{t4, t6}	3
{t5, t6}	2

↓

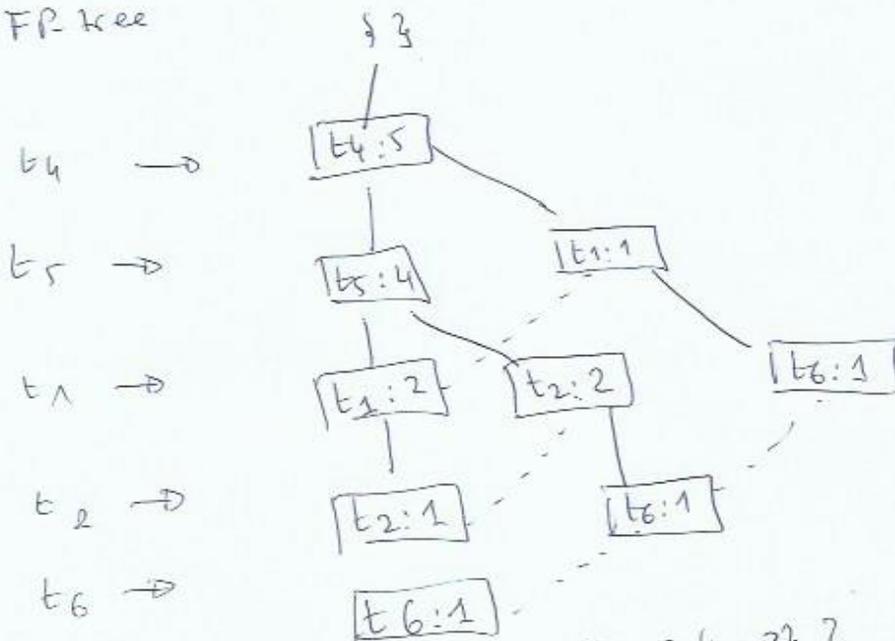
Itemset	sup
{t2,t4, t5}	3

$L = \{ \{t1\}, \{t2\}, \{t3\}, \{t4\}, \{t5\}, \{t1,t4\}, \{t2, t4\}, \{t2, t5\}, \{t4, t5\}, \{t4, t6\}, \{t2,t4, t5\} \}$

2) FP-Growth

FP List t_4, t_5, t_1, t_2, t_6

FP tree



$$L_1 = \{ \{t_4:5\}, \{t_5:4\}, \{t_1:1\}, \{t_2:2\}, \{t_6:1\} \}$$

$$L_2 = \{ \{t_4, t_5:4\}, \{t_4, t_1:3\}, \{t_5, t_2:3\}, \{t_4, t_2:3\}, \{t_4, t_6:3\} \}$$

$$L_3 = \{ \{t_5, t_4, t_2:3\} \}$$

3) ECLAT

C₁

{t ₁ }	d ₁ d ₃ d ₄
{t ₂ }	d ₁ d ₂ d ₅
{t ₃ }	/ d ₁ / d ₂ / /
{t ₄ }	d ₁ d ₂ d ₃ d ₄ d ₅
{t ₅ }	d ₁ d ₂ d ₃
{t ₆ }	d ₁ d ₂ d ₄
{t ₇ }	/ d ₂ / /
{t ₈ }	/ / d ₃ / /
{t ₉ }	/ / / d ₄ / /
{t ₁₀ }	/ / / d ₄ / d ₅ /
{t ₁₁ }	/ / / / d ₅ / /

L₁

{t ₁ }	d ₁ d ₃ d ₄
{t ₂ }	d ₁ d ₂ d ₅
{t ₄ }	d ₁ d ₂ d ₃ d ₄ d₅
{t ₅ }	d ₁ d ₂ d ₃ d ₅
{t ₆ }	d ₁ d ₂ d ₄

C₂

{t ₁ , t ₂ }	/ d ₁ / /
{t ₁ , t ₄ }	d ₁ d ₃ d ₄
{t ₁ , t ₅ }	/ d ₁ / d ₃ /
{t ₂ , t ₆ }	/ d ₁ / d ₄ /
{t ₂ , t ₄ }	d ₁ d ₂ d ₅
{t ₂ , t ₅ }	d ₁ d ₂ d ₅
{t ₂ , t ₆ }	/ d ₁ / d ₂ /
{t ₄ , t ₅ }	d ₁ d ₂ d ₃ d ₅
{t ₄ , t ₆ }	d ₁ d ₂ d ₄
{t ₅ , t ₆ }	/ d ₁ / d ₂ /

L₂

{t ₁ , t ₄ }	d ₁ d ₃ d ₄
{t ₂ , t ₄ }	d ₁ d ₂ d ₅
{t ₂ , t ₅ }	d ₁ d ₂ d ₅
{t ₄ , t ₅ }	d ₁ d ₂ d ₃ d ₅
{t ₄ , t ₆ }	d ₁ d ₂ d ₄

C₃

{t ₁ , t ₂ , t ₄ }	/ d ₁ / /
{t ₂ , t ₄ , t ₅ }	d ₁ d ₂ d ₅

L₃

{t ₂ , t ₄ , t ₅ }	d ₁ d ₂ d ₅
---	--

Exercice 2

Considérer la base suivante de cinq transactions. Supposer que $minsup = 60\%$ et $minconf = 80\%$.

TID	Articles achetés
T100	{M, O, N, K, E, Y}
T200	{D, O, N, K, E, Y}
T300	{M, A, K, E}
T400	{M, U, C, K, Y}
T500	{C, O, O, K, I, E}

- 1) Ecrire l'algorithme FP-Growth étudié en cours. Quelle est sa complexité ?
- 2) Déterminer tous les itemsets fréquents en appliquant FP-Growth
- 3) Ecrire l'algorithme ECLAT étudié en cours. Quelle est sa complexité ?
- 4) Appliquer l'algorithme ECLAT sur la table des transactions ci-dessus.
- 5) Comparer l'efficacité des deux méthodes.

Réponse

- 1) Ecrire l'algorithme FP-Growth étudié en cours. Quelle est sa complexité ?

Algorithme FP-Growth

Entrée : base de transactions, support minimum

Sortie : itemsets

début calculer F-list ; $O(n*m)$
 construire FP-tree ; $O(n*m)$
 pour chaque nœud de l'arbre **faire**
 Calculer itemsets li{nœud}
 L = U Li

fin ;

La complexité est $O(n * m)$ où n est le nombre de transactions et m le nombre d'items.

- (1) Déterminer tous les itemsets fréquents en appliquant l'algorithme Apriori

support = $60\% \times 5$ transactions = 3 transactions

C1

Itemset	support
{M}	3
{O}	3
{N}	2
{K}	5
{E}	4
{Y}	3
{D}	1
{A}	1
{U}	1
{C}	2
{I}	1

L1

Itemset	support
{M}	3
{O}	3
{K}	5
{E}	4
{Y}	3

C2

Itemset	support
{M,O}	1
{M,K}	3
{M,E}	2
{M,Y}	2
{O,K}	3
{O,E}	3
{O,Y}	2
{K,E}	4
{K,Y}	3
{E,Y}	2

L2

Itemset	support
{M,K}	3
{O,K}	3
{O,E}	3
{K,E}	4
{K,Y}	3

C3

Itemset	support
{O,K,E}	3

L3

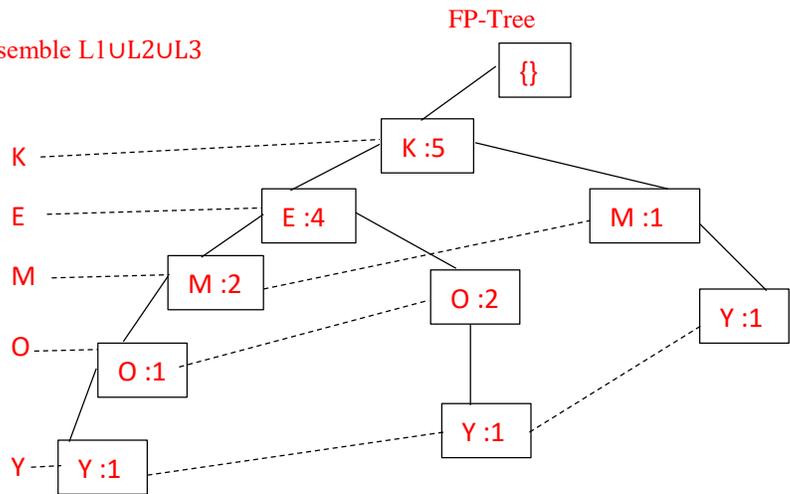
Itemset	support
{O,K,E}	3

Les itemsets fréquents sont renfermés dans l'ensemble L1UL2UL3

a. puis l'algorithme FP-growth.

F-list = K-E-M-O-Y

Item	fréquence
K	5
E	4
M	3
O	3
Y	3



Les itemsets fréquents sont obtenus en traversant l'arbre de la racine vers les feuilles:

- cardinalité égale à 1 : {K}, {E}, {M}, {O} et {Y} (* tous ceux qui existent après la racine avec un support=3 *)
- cardinalité égale à 2 : {K, E}, {K, M}, {K, O}, {K, Y} et {E, O} (* tous ceux qui existent après la racine et le préfixe avec un support=3 *)
- cardinalité égale à 3 : {K, E, O} (* tous ceux qui existent après la racine et les préfixes avec un support=3 *)

2) Ecrire l'algorithme ECLAT étudié en cours. Quelle est sa complexité ?

Algorithme ECLAT

Entrée : base de transactions, support minimum

Sortie : itemsets

Début

- Pour chaque transaction faire
 - o Pour chacun de ses items faire
 - Calculer le support

- L1 = itemsets qui respectent le support minimum
- Répéter tant que possible
 - o Faire l'intersection des itemsets
 - o Ne garder que ceux qui respectent le support minimum

Fin

La complexité pour L1 est $O(n \times m)$.

La complexité pour L2 est $O((n \times m)^2)$.

3) Appliquer l'algorithme ECLAT sur la table des transactions ci-dessus.

{M}	T100, T300, T400
{O}	T100, T200, T500
{K}	T100, T200, T300, T400, T500
{E}	T100, T200, T300, T500
{Y}	T100, T200, T400

{M, K}	T100, T300, T400
{O, K}	T100, T200, T500
{O, E}	T100, T200, T500
{K, E}	T100, T200, T500
{K, Y}	T100, T200, T400

{O, K, E}	T100, T200, T500
-----------	------------------

4) Comparer l'efficacité des deux méthodes.

Les deux méthodes sont plus efficaces que A-priori. FP-Growth utilise une structure de données assez élaborée qui permet d'augmenter l'efficacité de l'algorithme. L'arbre FP-Tree résume de manière verticale les items qui apparaissent ensemble dans les transactions.

ECLAT est plus simple et son efficacité repose aussi sur une approche verticale pour le calcul des itemsets. Cependant, il nécessite plus de temps de calcul.