LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Database Systems Group • Prof. Dr. Thomas Seidl

Exercise 3: Frequent Itemset Mining

Knowledge Discovery in Databases I

SS 2016

Basic terms and definitions:

- Items $I = \{i_1, \ldots, i_m\}$
- Itemset $X \subseteq I$
- Database D
- Transactions T

TID	items
100	{butter, bread, milk, sugar}
200	{butter, flour, milk, sugar}
300	{butter, eggs, milk, salt}
400	{eggs}
500	{butter, flour, milk, salt sugar}

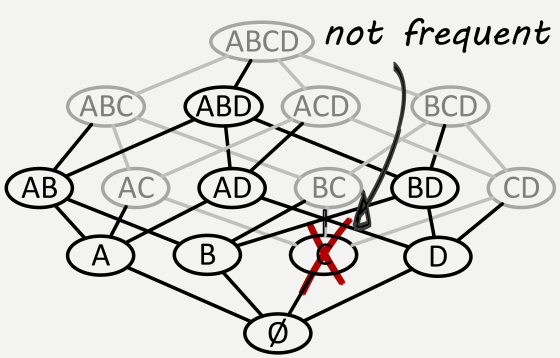
- Support: $support(X) = |\{T \in D \mid X \subseteq T\}|$
- Frequent Itemset: X freq. iff $support(X) \geq minSup$

Goal: Find all frequent itemsets in D!

Naive Algorithm: Just count the frequencies of *all possible* subsets of I in the database.

- Problem: For $|I| = m$, there are 2^m such itemsets!
- Clearly, this becomes infeasible rather quickly…

Main idea of the Apriori algorithm: Prune the exponential search space using anti-monotonicity



The Apriori algorithm makes use of prior knowledge of subset support properties. Prove the following subset properties:

a) All non-empty subsets of a frequent itemset must also be frequent.

b) The support of any non-empty subset S' of itemset S must be as great as the support of S.

a) All non-empty subsets of a frequent itemset must also be frequent:

Proof:

- Let $S \subseteq I$ be a frequent itemset, i.e. $support(S) \geq minSup$
- Let $\emptyset \neq S' \subseteq S$
- Then

$$support(S') \geq^{b)} support(S)$$
$$\geq^{S \text{ is freq.}} minSup$$

i.e. S' is a frequent itemset.

b) The support of any non-empty subset S' of itemset S must be as great as the support of S.

Proof:

- Let $\emptyset \neq S' \subseteq S \subseteq I$
- For any transaction $T \subseteq I$ in database D, we have:

$$S \subseteq T \Rightarrow S' \subseteq T$$

- Thus, it holds that

$$\{T \in D \mid S \subseteq T\} \subseteq \{T \in D \mid S' \subseteq T\}$$

and consequently

$$support(S) = |\{T \in D \mid S \subseteq T\}| \leq |\{T \in D \mid S' \subseteq T\}| = support(S')$$

Let D be a database that contains the following four transactions:

TID	items_bought
T1	{K, A, D, B}
T2	{D, A, C, E, B}
T3	{C, A, B, E}
T4	{B, A, D}

In addition let $minSup = 60\%$.

a) Find all frequent itemsets using the Apriori algorithm.

b) Find all frequent itemsets using the FP-growth algorithm.

c) Determine all closed and maximal frequent itemsets.

Exercise 3-2 (a): Apriori Algorithm

minSup=0.6

database D

TID	items
1	{K, A, D, B}
2	{D, A, C, E, B}
3	{C, A, B, E}
4	{B, A, D}

scan D

C_1

itemset	sup
{A}	100%
{B}	100%
{C}	50%
{D}	75%
{E}	50%
{K}	25%

L_1

itemset	sup
{A}	100%
{B}	100%
{D}	75%

$L_1 \bowtie L_1$

C_2

itemset
{A B}
{A D}
{B D}

prune C_2

C_2

itemset
{A B}
{A D}
{B D}

scan D

C_2

itemset	sup
{A B}	100%
{A D}	75%
{B D}	75%

L_2

itemset	sup
{A B}	100%
{A D}	75%
{B D}	75%

$L_2 \bowtie L_2$

C_3

itemset
{A B D}

prune C_3

C_3

itemset
{A B D}

scan D

C_3

itemset	sup
{A B D}	75%

L_3

itemset	sup
{A B D}	75%

$L_3 \bowtie L_3$

C_4 is empty

Bottleneck of Apriori: Candidate generation

- Huge candidate set
- Multiple scans of the database

FP-Growth: FP-mining without candidate generation

- Compress database, retain only information relevant to FP-mining: *FP-tree*
- Use efficient *Divide & Conquer* approach and *grow* frequent patterns without generating candidate sets

Exercise 3-2 (b): FP-Growth Algorithm

TID	items bought
1	{K, A, D, B}
2	{D, A, C, E, B}
3	{C, A, B, E}
4	{B, A, D}

(ordered) frequent items
{A, B, D}
{A, B, D}
{A, B}
{A, B, D}

minSup=0.6

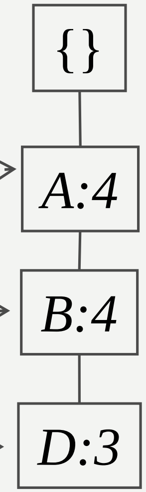
sort items in the order of descending support

header table:

item	frequency
A	4
B	4
D	3
C	2
E	2
K	1

for each transaction only keep its frequent items sorted in descending order of their frequencies

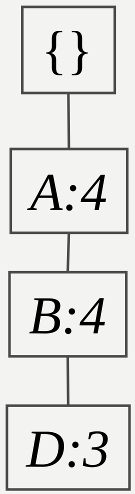
Initial FP-tree



Exercise 3-2 (b): FP-Growth Algorithm

Initial FP-tree

item	frequency
A	4
B	4
D	3
C	2
E	2
K	1

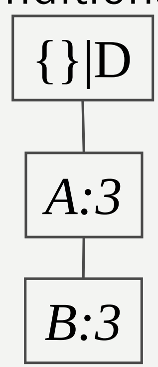


conditional pattern base:

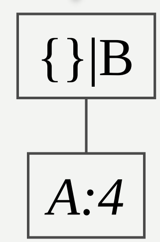
item	cond. pattern base
A	*{}*
B	*A:4*
D	*AB:3*

item	frequency
A	3
B	3

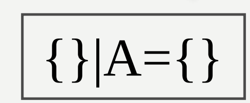
D-conditional FP-tree



{{D},{AD},{BD},{ABD}}



{{B},{AB}}



{{A}}

Exercise 3-2 (c): Closed and Maximal Frequent Itemsets

- Closed frequent itemsets:
  - $X\ closed \Leftrightarrow \nexists Y: X \subset Y \wedge support(Y) = support(X)$
  - Set of closed itemsets contains complete information
- Maximal frequent itemsets:
  - $X\ maximal \Leftrightarrow \nexists Y: X \subset Y \wedge support(Y) \geq minSup$
  - Not complete, but more compact

TID	items_bought
T1	{K, A, D, B}
T2	{D, A, C, E, B}
T3	{C, A, B, E}
T4	{B, A, D}

	frequent itemsets	support
	{A}	1
	{B}	1
	{D}	0.75
closed but not maximal →	{A,B}	1
	{A,D}	0.75
	{B,D}	0.75
closed & maximal →	{A,B,D}	0.75

Association rule:

$$X \Rightarrow Y$$

where $X, Y \subseteq I$ are two itemsets with $X \cap Y = \emptyset$.

- $support(X \Rightarrow Y) = support(X \cup Y)$
- $confidence(X \Rightarrow Y) = \frac{support(X \cup Y)}{support(X)}$
- *Strong* association rules have $support \geq minSup$ and $confidence \geq minConf$

Goal: Find all strong association rules in D!

After frequent itemset mining, association rules can be extracted as follows: For each frequent itemset X and every non-empty subset $Y \subset X$, generate a rule $Y \Rightarrow X \setminus Y$ if it fulfills the minimum confidence property.

a) Proof the following anti-monotonicity lemma for strong association rules:

Let X be a frequent itemset and $Y \subset X$. If $Y \Rightarrow X \setminus Y$ is a strong association rule, then $Y' \Rightarrow X \setminus Y'$ is also a strong association rule for every $Y \subseteq Y'$.

Let X be a frequent itemset and $Y \subset X$. If $Y \Rightarrow X \setminus Y$ is a strong association rule, then $Y' \Rightarrow X \setminus Y'$ is also a strong association rule for every $Y \subseteq Y'$.

Proof:

- $support(Y' \Rightarrow X \setminus Y') = support(X)$

  $\geq^{X \text{ is freq.}} minSup$

- $confidence(Y' \Rightarrow X \setminus Y') = \frac{support(X)}{support(Y')}$

  $\geq^{3-1(b)} \frac{support(X)}{support(Y)}$

  $= confidence(Y \Rightarrow X \setminus Y)$

  $\geq^{Y \Rightarrow X \setminus Y \text{ is strong}} minConf$

b) Extract all strong association rules from the database D provided in the previous exercise with a minimum confidence of $minConf = 80\%$. Which candidate rules can be pruned based on anti-monotonicity?

frequent itemsets	support
{A}	1
{B}	1
{D}	0.75
{A,B}	1
{A,D}	0.75
{B,D}	0.75
{A,B,D}	0.75

candidate rule	confidence	
$A \Rightarrow B$	1	✓
$B \Rightarrow A$	1	✓
$A \Rightarrow D$	0.75	✗
$D \Rightarrow A$	1	✓
$B \Rightarrow D$	0.75	✗
$D \Rightarrow B$	1	✓
$A, B \Rightarrow D$	0.75	✗
$A, D \Rightarrow B$	1	✓
$B, D \Rightarrow A$	1	✓
$D \Rightarrow A, B$	1	✓

$A \Rightarrow B, D$ and $B \Rightarrow A, D$ can be pruned!