Association Rules Outline

Goal: Provide an overview of basic Association Rule mining techniques

- Association Rules Problem Overview
  - Large itemsets
- Association Rules Algorithms
  - **Apriori**
  - Eclat
  - FP-Growth
  - Etc.

Example: Market Basket Data

- Items frequently purchased together:

  Bread ⇒PeanutButter

- Uses:
  - Placement
  - Advertising
  - Sales
  - Coupons

- Objective: increase sales and reduce costs

Association Rule Techniques

Step1: Find **Large** Frequent Itemsets.

Step 2: Generate **rules** from frequent itemsets.

Input:
- D //Database of transactions
- I //Items
- L //Large itemsets
- s //Support
- α //Confidence

Output:
- R //Association Rules satisfying s and α

ARGen Algorithm:

$R = \emptyset$;
for each $l \in L$ **do**
  for each $x \subset l$ **such that** $x \neq \emptyset$ **and** $x \neq l$ **do**
    if $\frac{support(l)}{support(x)} \geq \alpha$ **then**
      $R = R \cup \{x \Rightarrow (l - x)\}$;

Association Rule Definitions

- ***Set of items:*** $I=\{I_1,I_2,\ldots,I_m\}$
- ***Transactions:*** $D=\{t_1,t_2,\ldots,t_n\}$, $t_j \subseteq I$
- ***Itemset:*** $\{I_{i1},I_{i2},\ldots,I_{ik}\} \subseteq I$
- ***Support of an itemset:*** Percentage of transactions which contain that itemset.
- ***Large (Frequent) itemset:*** Itemset whose number of occurrences is above a threshold (Minimum Support).

Example: Support

Transaction	Items
t_1	Bread,Jelly,PeanutButter
t_2	Bread,PeanutButter
t_3	Bread,Milk,PeanutButter
t_4	Beer,Bread
t_5	Beer,Milk

I = { Beer, Bread, Jelly, Milk, PeanutButter}

Support of {Bread,PeanutButter} is 3/5= 60%

Example: Support

Transaction ID	Items Bought
1	Shoes, Shirt, Jacket
2	Shoes,Jacket
3	Shoes, Jeans
4	Shirt, Sweatshirt

I = { Shoes, Shirt, Jacket, Jeans, Sweatshirt}

Frequent Itemset	**Support**
{Shoes}	3/4 = 75%
{Shirt}	2/4 = 50%
{Jacket}	2/4 = 50%
{Shoes, Jacket}	2/4 = 50%

In the example database, the {Shoes, Jacket} itemset has a support of 2/4 = 0.5 since it occurs in 50% of all transactions (1 out of 2 transactions).

Association Rule Definitions

- ***Association Rule (AR):*** implication

  $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$;

- ***Support of AR (s) $X \Rightarrow Y$***: Percentage of transactions that contain $X \cup Y$
- ***Confidence of AR (α) $X \Rightarrow Y$:*** Ratio of number of transactions that contain

  $X \cup Y$ to the number that contain X

  (i.e., supp(X U Y)/supp(X))

Example: Confidence

- The rule {Shoes}→{Jacket} has a confidence of 0.5/0.75 = 66% in the database, which means that for 66% of the transactions containing Shoes the rule is correct (66% of the times a customer buys Shoes, Jacket is bought as well).

- The rule {Jacket}→{Shoes} has a confidence of 0.5/0.5 = 100% in the database, which means that for 100% of the transactions containing Jacket the rule is correct (100% of the times a customer buys Jacket, Shoes is bought as well).

Frequent Itemset	**Support**
{Shoes}	3/4 = 75%
{Shirt}	2/4 = 50%
{Jacket}	2/4 = 50%
{Shoes, Jacket}	2/4 = 50%

Transaction ID	Items Bought
1	Shoes, Shirt, Jacket
2	Shoes,Jacket
3	Shoes, Jeans
4	Shirt, Sweatshirt

Example: Association Rules

Transaction ID	Items Bought
1	Shoes, Shirt, Jacket
2	Shoes,Jacket
3	Shoes, Jeans
4	Shirt, Sweatshirt

If the ***minimum support*** is 50%, then {Shoes, Jacket} is the only 2- itemset that satisfies the minimum support.

Frequent Itemset	Support
{Shoes}	75%
{Shirt}	50%
{Jacket}	50%
{Shoes, Jacket}	50%

If the ***minimum confidence*** is 50%, then the only two rules generated from this 2-itemset, that have confidence greater than 50%, are:

Shoes $\Rightarrow$ Jacket Support=50%, Confidence=66%
Jacket $\Rightarrow$ Shoes Support=50%, Confidence=100%

Association Rule Problem

- Given a set of items $I=\{I_1,I_2,\ldots,I_m\}$ and a database of transactions $D=\{t_1,t_2,\ldots,t_n\}$ where $t_i=\{I_{i1},I_{i2},\ldots,I_{ik}\}$ and $I_{ij} \in I$, the ***Association Rule Problem*** is to identify all association rules $X \Rightarrow Y$ with a minimum support and confidence.

- ***NOTE:*** Support of $X \Rightarrow Y$ is same as support of $X \cup Y$.

Apriori Algorithm

Definition of Apriori Algorithm

- In computer science and data mining, **Apriori** is a classic algorithm for learning association rules.

- Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation).

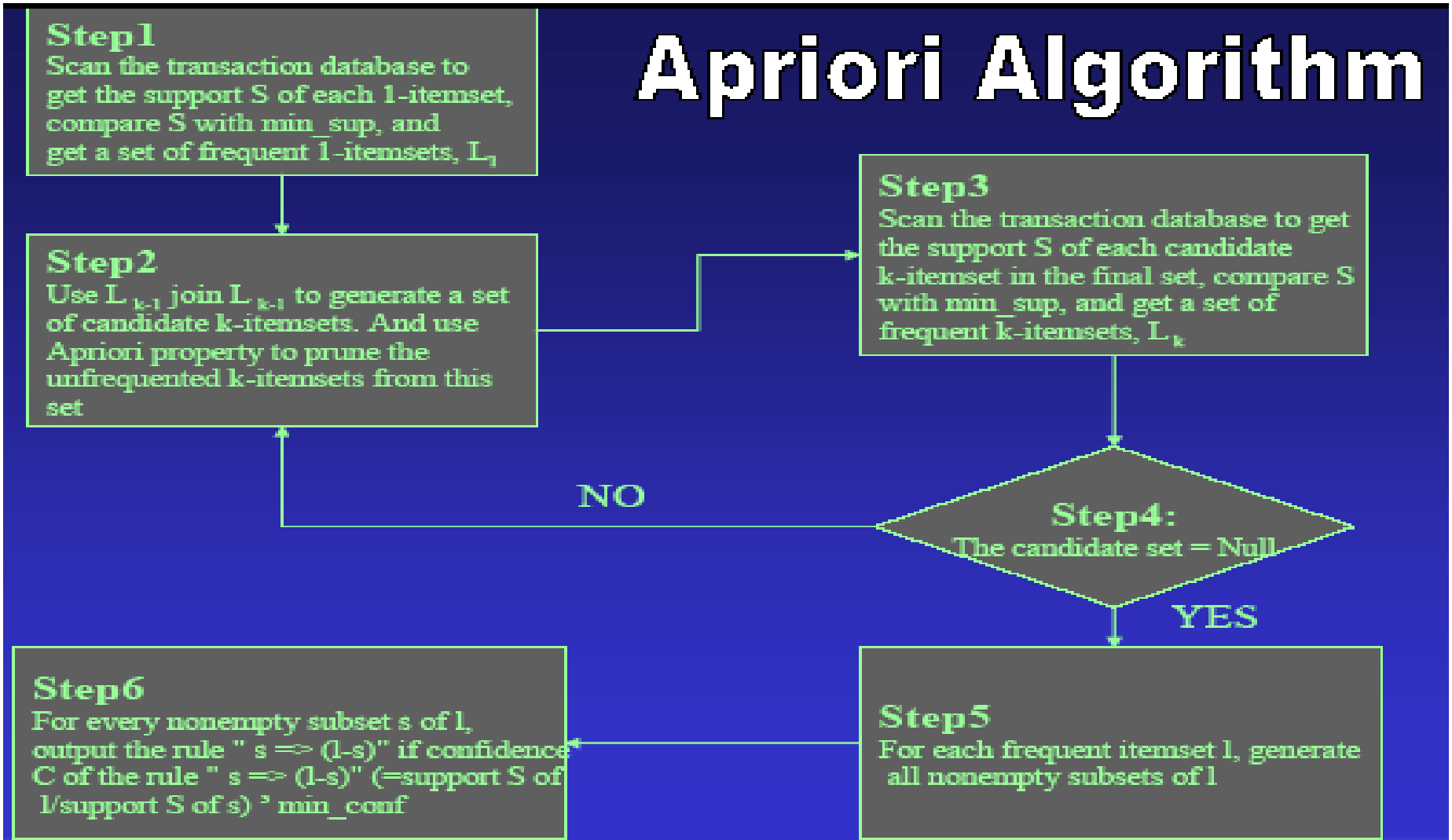
- The algorithm attempts to find subsets which are common to at least a minimum number C (the cutoff, or confidence threshold) of the itemsets.

Definition (cont’d.)

- Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as *candidate generation*, and groups of candidates are tested against the data.

- The algorithm terminates when no further successful extensions are found.

Steps to Perform Apriori Algorithm



Apriori--- Find Large Itemset

- ***Large Itemset Property:***

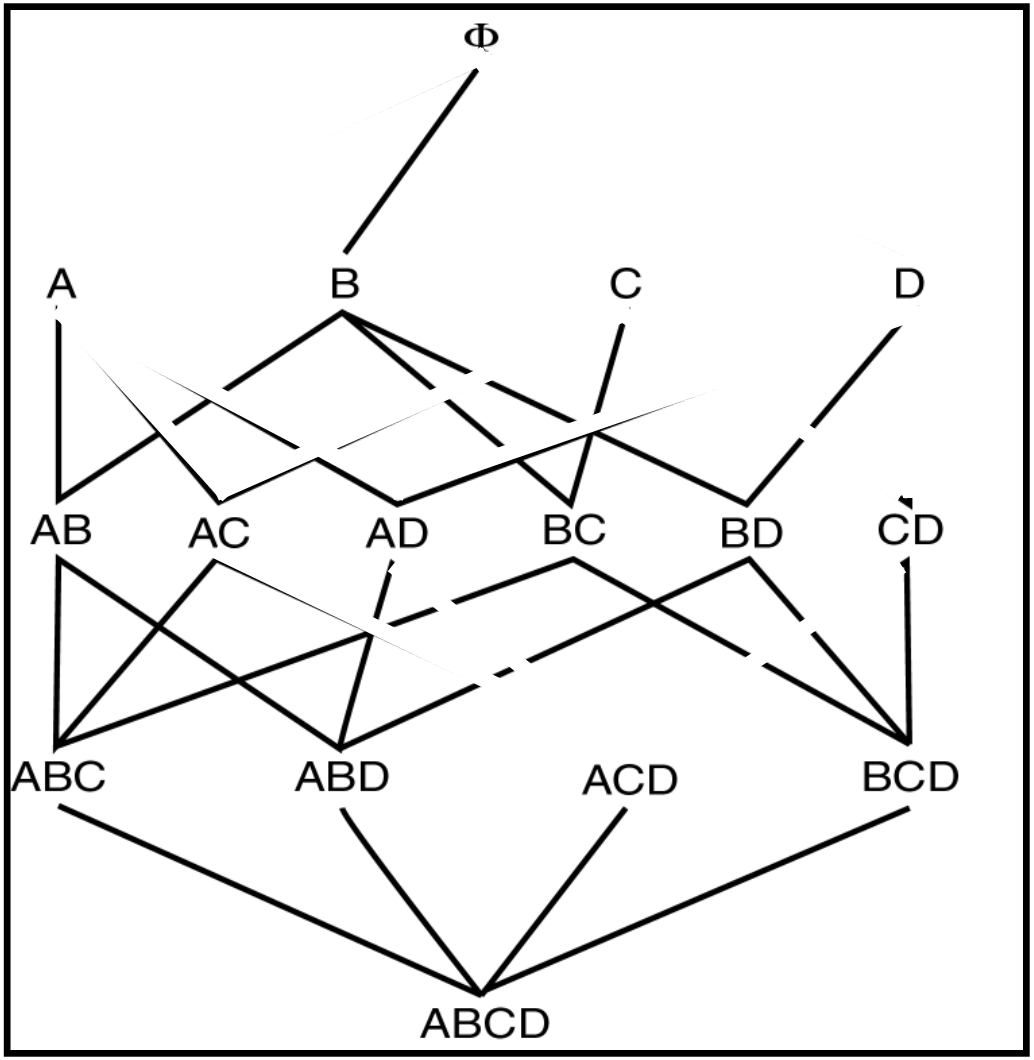
Any subset of a large itemset is large.

- Contrapositive:

If an itemset is not large,

none of its supersets are large.

Large Itemset Property

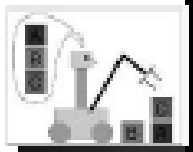


Apriori Algorithm

1. C_1 = Itemsets of size one in I;
2. Determine all large itemsets of size 1, L_1;
3. i = 1;
4. Repeat
5. i = i + 1;
6. C_i = **Apriori-Gen**(L_{i-1});
7. Count C_i to determine L_i;
8. until no more large itemsets found;

Apriori-Gen(L_{i-1})

- Generate candidates of size i+1 from large itemsets of size i.
- Approach used: join large itemsets of size i if they agree on i-1
- May also prune candidates who have subsets that are not large.



Apriori Algorithm

Pseudo code

$L_1 = \{\text{large1} - \text{itemsets}\}$ *count item frequency*

$\text{for}(K = 2; L_{k-1} \neq \{\}; k++)\text{do begin}$

$C_k = \text{apriori} - \text{gen}(L_{k-1});$ *new candidates*

$\forall \text{transactions } t \in D \text{ do begin}$

$C_t = \text{subset}(C_k, t);$ *candidates in transaction*

$\forall \text{candidates } c \in C_t \text{ do}$

$c.count++;$ *determine support*

end

$L_k = \{c \in C_k \mid c.count \geq \text{minsup}\}$ *create new set*

end

$\text{Answer} = \bigcup_k L_k;$

The Apriori Algorithm — Example

Minimum support = 2 or 50%

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan D

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

Scan D

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_3

itemset
{2 3 5}

Scan D

L_3

itemset	sup
{2 3 5}	2

Answer = L_1 U L_2 U L_3

Example: Apriori

Pass	Candidates	Large Itemsets
1	{Beer},{Bread},{Jelly}, {Milk},{PeanutButter}	{Beer},{Bread}, {Milk},{PeanutButter}
2	{Beer,Bread},{Beer,Milk}, {Beer,PeanutButter},{Bread,Milk}, {Bread,PeanutButter},{Milk,PeanutButter}	{Bread,PeanutButter}

Minimum support = 30%

Example: Apriori-Gen

Transaction	Items
t_1	Blouse
t_2	Shoes,Skirt,TShirt
t_3	Jeans,TShirt
t_4	Jeans,Shoes,TShirt
t_5	Jeans,Shorts
t_6	Shoes,TShirt
t_7	Jeans,Skirt
t_8	Jeans,Shoes,Shorts,TShirt
t_9	Jeans
t_{10}	Jeans,Shoes,TShirt
t_{11}	TShirt
t_{12}	Blouse,Jeans,Shoes,Skirt,TShirt
t_{13}	Jeans,Shoes,Shorts,TShirt
t_{14}	Shoes,Skirt,TShirt
t_{15}	Jeans,TShirt
t_{16}	Skirt,TShirt
t_{17}	Blouse,Jeans,Skirt
t_{18}	Jeans,Shoes,Shorts,TShirt
t_{19}	Jeans
t_{20}	Jeans,Shoes,Shorts,TShirt

Example: Apriori-Gen (cont'd)

Scan	Candidates	Large Itemsets
1	{Blouse},{Jeans},{Shoes}, {Shorts},{Skirt},{TShirt}	{Jeans},{Shoes},{Shorts} {Skirt},{Tshirt}
2	{Jeans,Shoes},{Jeans,Shorts},{Jeans,Skirt}, {Jeans,TShirt},{Shoes,Shorts},{Shoes,Skirt}, {Shoes,TShirt},{Shorts,Skirt},{Shorts,TShirt}, {Skirt,TShirt}	{Jeans,Shoes},{Jeans,Shorts}, {Jeans,TShirt},{Shoes,Shorts}, {Shoes,TShirt},{Shorts,TShirt}, {Skirt,TShirt}
3	{Jeans,Shoes,Shorts},{Jeans,Shoes,TShirt}, {Jeans,Shorts,TShirt},{Jeans,Skirt,TShirt}, {Shoes,Shorts,TShirt},{Shoes,Skirt,TShirt}, {Shorts,Skirt,TShirt}	{Jeans,Shoes,Shorts}, {Jeans,Shoes,TShirt}, {Jeans,Shorts,TShirt}, {Shoes,Shorts,TShirt}
4	{Jeans,Shoes,Shorts,TShirt}	{Jeans,Shoes,Shorts,TShirt}
5	$\emptyset$	$\emptyset$

Apriori Adv/Disadv

- ***Advantages:***
  - Uses large itemset property.
  - Easily parallelized
  - Easy to implement.
- ***Disadvantages:***
  - Assumes transaction database is memory resident.
  - Requires **many** database scans.

Generate Rule

Step1: Find Large Frequent Itemsets.

Step 2: Generate **rules** from frequent itemsets.

Input:
- D //Database of transactions
- I //Items
- L //Large itemsets
- s //Support
- α //Confidence

Output:
- R //Association Rules satisfying s *and* α

ARGen Algorithm:

$R = \emptyset$;
for each $l \in L$ **do**
  for each $x \subset l$ **such that** $x \neq \emptyset$ **and** $x \neq l$ **do**
    if $\frac{support(l)}{support(x)} \geq \alpha$ **then**
      $R = R \cup \{x \Rightarrow (l - x)\}$;

Rules

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Support = 75% Confidence = 100%

- 5 → 2
- 2 → 5

itemset	support
{1}	2
{2}	3
{3}	3
{5}	3

2-itemset	support
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

3-itemset	support
{2 3 5}	2

Support = 50% Confidence = 100%

- 1 → 3 3,5 → 2 2,3 → 5

Support = 50% Confidence = 67%

- 3 → 1 2 → 3 3 → 2 3 → 5 5 → 3 2 →3,5
- 5 → 2,3 2,5 → 3 3 → 2,5

Example: Association Rules

$X \Rightarrow Y$	support	confidence
Bread $\Rightarrow$ PeanutButter	60%	75%
PeanutButter $\Rightarrow$ Bread	60%	100%
Beer $\Rightarrow$ Bread	20%	50%
PeanutButter $\Rightarrow$ Jelly	20%	33.3%
Jelly $\Rightarrow$ PeanutButter	20%	100%
Jelly $\Rightarrow$ Milk	0%	0%

Exercise after finishing

Summary

- Association Rules form an very applied data mining approach.
- Association Rules are derived from frequent itemsets.
- The Apriori algorithm is an efficient algorithm for finding all frequent itemsets.
- The Apriori algorithm implements level-wise search using frequent item property.
- The Apriori algorithm can be additionally optimized.
- There are many measures for association rules.

References

- Agrawal R, Imielinski T, Swami AN. "Mining Association Rules between Sets of Items in Large Databases." *SIGMOD*. June 1993, **22**(2):207-16, pdf.
- Agrawal R, Srikant R. "Fast Algorithms for Mining Association Rules", *VLDB*. Sep 12-15 1994, Chile, 487-99, pdf, ISBN 1-55860-153-8.
- Retrieved from http://en.wikipedia.org/wiki/Apriori_algorithm
- I. H. Witten, E. Frank and M. A. Hall, “Data Mining: Practical Machine Learning Tools and Techniques”, 3rd Edition, Morgan Kaufmann.

โจทย์ กำหนดให้ห้าง ABC ขายสินค้าจำนวน 6 ชนิดคือ Chips, Coke, HotDog, Bread, Ketchup, Milk ฐานข้อมูลของห้าง ABC มีข้อมูลจำนวน 4 ทรานแซคชัน ดังนี้

Transaction ID	Item
1	Ketchup, Chips, HotDog, Coke
2	HotDog, Chips, Milk, Bread, Coke
3	Milk, Chips, Coke, Bread
4	Coke, Chips, HotDog

ให้นักศึกษาใช้ Apriori Algorithm แสดงวิธีทำเพื่อทำการหากฎที่มีค่า minimum support = 60% และค่า confidence = 80% จากฐานข้อมูลของห้าง ABC