DATA ANALYTICS

Unit 4: Market Basket Analysis (Apriori Algorithm)

Jyothi R.

Department of Computer Science
and Engineering

Market Basket Analysis

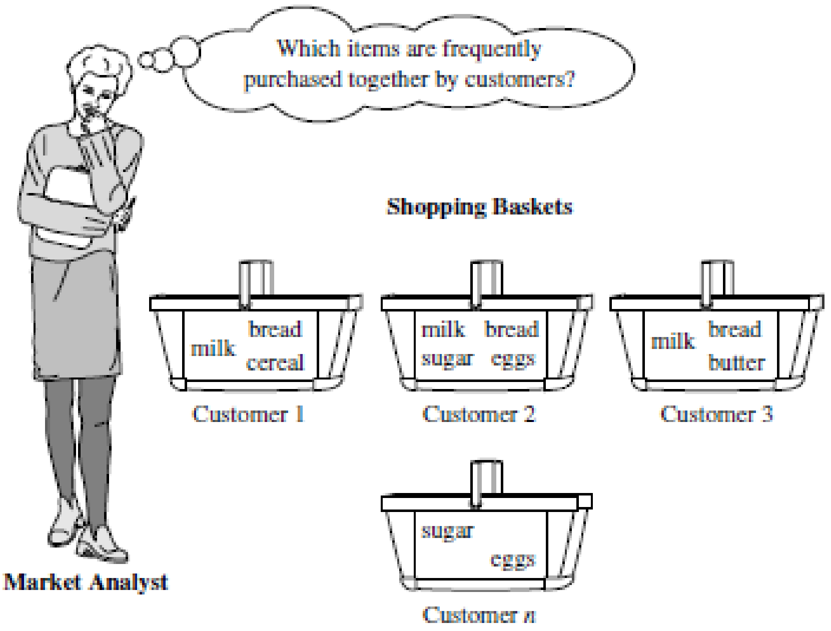
- The market-basket model of data is used to describe a common form of many- many relationship between two kinds of objects.
- On the one hand, we have items, and on the other we have baskets, sometimes called “transactions.”
- Each basket consists of a set of items (an itemset), and usually we assume that the number of items in a basket is small – much smaller than the total number of items.
- The number of baskets is usually assumed to be very large, bigger than what can fit in main memory.
- The data is assumed to be represented in a file consisting of a sequence of baskets.
- In terms of the distributed file system the baskets are the objects of the file, and each basket is of type “set of items.”

Market Basket Analysis

- Frequent itemset mining leads to the discovery of associations and correlations among items in large transactional or relational data sets.
- With massive amounts of data continuously being collected and stored, many industries are becoming interested in mining such patterns from their databases.
- The discovery of interesting correlation relationships among huge amounts of business transaction records can help in many business decision-making processes such as catalog design, cross-marketing, and customer shopping behavior analysis.

Market Basket Analysis

- A typical example of frequent itemset mining is market basket analysis.
- This process analyzes customer buying habits by finding associations between the different items that
- customers place in their “shopping baskets” as shown in Figure : **Market Basket Analysis**



Market Basket Analysis

- The discovery of these associations can help retailers develop marketing strategies by gaining insight into which items are frequently purchased together by customers.
- For instance, if customers are buying milk, how likely are they to also buy bread (and what kind of bread) on the same trip to the supermarket?
- This information can lead to increased sales by helping retailers do selective marketing and plan their shelf space.

Market Basket Analysis

- Example: Suppose, as manager of an AllElectronics branch, you would like to learn more about the buying habits of your customers.

- Specifically, you wonder, “Which groups or sets of items are customers likely to purchase on a given trip to the store?”

- To answer your question, market basket analysis may be performed on the retail data of customer transactions at your store.

- You can then use the results to plan marketing or advertising strategies, or in the design of a new catalog.

Market Basket Analysis

- Market basket analysis may help you design different store layouts. In one strategy, items that are frequently purchased together can be placed in proximity to further encourage the combined sale of such items.
- If customers who purchase computers also tend to buy antivirus software at the same time, then placing the hardware display close to the software display may help increase the sales of both items.

Market Basket Analysis

- In an alternative strategy, placing hardware and software at opposite ends of the store may entice customers who purchase such items to pick up other items along the way.

- For instance, after deciding on an expensive computer, a customer may observe security systems for sale while heading toward the software display to purchase antivirus software, and may decide to purchase a home security system as well.

- Market basket analysis can also help retailers plan which items to put on sale at reduced prices.

- If customers tend to purchase computers and printers together, then having a sale on printers may encourage the sale of printers as well as computers.

Frequent ItemSet

Itemset

- A collection of one or more items
  - Example: {Milk, Bread, Diaper}
- k-itemset
  - An itemset that contains k items

Support count (σ)

- Frequency of occurrence of an itemset
- E.g. σ({Milk, Bread,Diaper}) = 2

Support

- Fraction of transactions that contain an itemset
- E.g. s({Milk, Bread, Diaper}) = 2/5

Frequent Itemset

- An itemset whose support is greater than or equal to a *minsup* threshold

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Association Rule Mining

- **Association Rule**
  - An implication expression of the form X → Y, where X and Y are itemsets
  - Example:
    {Milk, Diaper} → {Beer}

TID	*Items*
1	**Bread, Milk**
2	**Bread, Diaper, Beer, Eggs**
3	**Milk, Diaper, Beer, Coke**
4	**Bread, Milk, Diaper, Beer**
5	**Bread, Milk, Diaper, Coke**

- **Rule Evaluation Metrics**
  - Support (s)
    - ◆ Fraction of transactions that contain both X and Y
  - Confidence (c)
    - ◆ Measures how often items in Y appear in transactions that contain X

Example:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \text{Beer}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Association Rule Mining

Two-step approach:

1. Frequent Itemset Generation
   - Generate all itemsets whose support ≥ minsup

2. Rule Generation
   - Generate high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset

Frequent itemset generation is still computationally expensive

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Apriori Principle

If an itemset is frequent, then all of its subsets must also be frequent

Apriori principle holds due to the following property of the support measure:

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

Support of an itemset never exceeds the support of its subsets
This is known as the anti-monotone property of support

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Generating frequent itemsets

Item	Count
Bread	**4**
Coke	**2**
Milk	**4**
Beer	**3**
Diaper	**4**
Eggs	**1**

Items (1-itemsets)

TID	*Items*
1	**Bread, Milk**
2	**Bread, Diaper, Beer, Eggs**
3	**Milk, Diaper, Beer, Coke**
4	**Bread, Milk, Diaper, Beer**
5	**Bread, Milk, Diaper, Coke**

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Generating frequent itemsets (given minsup)

Items (1-itemsets)

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Pairs (2-itemsets)

(No need to generate candidates involving Coke or Eggs)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Triplets (3-itemsets)

Itemset	Count
{Bread,Milk,Diaper}	2

Minimum Support = 3

If every subset is considered,

$$^6C_1 + {}^6C_2 + {}^6C_3 = 41$$

With support-based pruning,

$$6 + 6 + 1 = 13$$

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Apriori Algorithm for Frequent Itemset Generation

Method:

- Let k=1
- Generate frequent itemsets of length 1
- Repeat until no new frequent itemsets are identified
  - Generate length (k+1) candidate itemsets from length k frequent itemsets
  - Prune candidate itemsets containing subsets of length k that are infrequent
  - Count the support of each candidate by scanning the DB
  - Eliminate candidates that are infrequent, leaving only those that are frequent

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Minimum support (minsup)

- Note that the itemset support defined is sometimes referred to as *relative support*, whereas the occurrence frequency is called the **absolute support**.
- If the relative support of an itemset *I* satisfies a prespecified **minimum support threshold** (i.e., the absolute support of *I* satisfies the corresponding **minimum support count threshold**), then *I* is a **frequent** itemset.
- The set of frequent *k*-itemsets is commonly denoted by L_k

Applying multiple minimum support

How to apply multiple minimum support?

MS(i): minimum support for item i

e.g.: MS(Milk)=5%, MS(Coke) = 3%,
MS(Broccoli)=0.1%, MS(Salmon)=0.5%

MS({Milk, Broccoli}) = min (MS(Milk), MS(Broccoli))
= 0.1%

Challenge: Support is no longer anti-monotone

Suppose: Support(Milk, Coke) = 1.5% and
Support(Milk, Coke, Broccoli) = 0.5%

{Milk,Coke} is infrequent but {Milk,Coke,Broccoli} is frequent

Order the items according to their minimum support (in ascending order)

e.g.: MS(Milk)=5%, MS(Coke) = 3%,
MS(Broccoli)=0.1%, MS(Salmon)=0.5%

Ordering: Broccoli, Salmon, Coke, Milk

Need to modify Apriori such that:

L_1 : set of frequent items

F_1 : set of items whose support is $\geq$ MS(1)
where MS(1) is $\min_i$(MS(i))

C_2 : candidate itemsets of size 2 is generated from F_1 instead of L_1

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Multiple minimum support and modified Apriori

Order the items according to their minimum support (in ascending order)

e.g.: MS(Milk)=5%, MS(Coke) = 3%,
MS(Broccoli)=0.1%, MS(Salmon)=0.5%

Ordering: Broccoli, Salmon, Coke, Milk

Need to modify Apriori such that:

L_1 : set of frequent items

F_1 : set of items whose support is ≥ MS(1)
where MS(1) is $\min_i(MS(i))$

C_2 : candidate itemsets of size 2 is generated from F_1
instead of L_1

Modifications to Apriori: In traditional Apriori, A candidate (k+1)-itemset is generated by merging two frequent itemsets of size k

The candidate is pruned if it contains any infrequent subsets of size k

Pruning step has to be modified:

Prune only if subset contains the first item

e.g.: Candidate={Broccoli, Coke, Milk} (ordered according to minimum support)

{Broccoli, Coke} and {Broccoli, Milk} are frequent but {Coke, Milk} is infrequent

Candidate is not pruned because {Coke, Milk} does not contain the first item, i.e., Broccoli.

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Rule Generation

How to efficiently generate rules from frequent itemsets?

In general, confidence does not have an anti-monotone property

$c(ABC \rightarrow D)$ can be larger or smaller than $c(AB \rightarrow D)$

But confidence of rules generated from the same itemset has an anti-monotone property e.g., L = {A,B,C,D}:

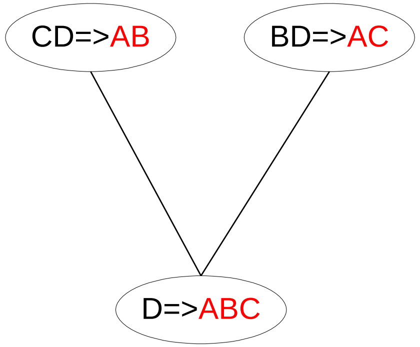
$$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

Confidence is anti-monotone w.r.t. number of items on the RHS of the rule

Candidate rule is generated by merging two rules that share the same prefix in the rule consequent

join(CD=>AB,BD=>AC)
would produce the candidate rule D => ABC

Prune rule D=>ABC if its
subset AD=>BC does not have high confidence



Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Support and Confidence

$$support(A \Rightarrow B) = P(A \cup B)$$

$$confidence(A \Rightarrow B) = P(B|A).$$

$$confidence(A \Rightarrow B) = P(B|A) = \frac{support(A \cup B)}{support(A)} = \frac{support\_count(A \cup B)}{support\_count(A)}.$$

Computing Confidence

- In confidence of rule equation A => B can be easily derived from the support counts of A and A∪B.
- That is, once the support counts of A, B, and A ∪B are found, it is straightforward to derive the corresponding association rules A =>B and B =>A and check whether they are strong.
- Thus, the problem of mining association rules can be reduced to that of mining frequent itemsets.

Evaluation of an association rule

Contingency table for X → Y

	Y	Y	
X	f_{11}	f_{10}	f_{1+}
X	f_{01}	f_{00}	f_{o+}
	f_{+1}	f_{+0}	|T|

f_{11}: support of X and Y
f_{10}: support of X and $\overline{Y}$
f_{01}: support of $\overline{X}$ and Y
f_{00}: support of $\overline{X}$ and $\overline{Y}$

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Interest = \frac{P(X,Y)}{P(X)P(Y)}$$

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi - coefficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Limitation of Confidence

	Coffee	Not Coffee	
Tea	15	5	20
Not Tea	75	5	80
	90	10	100

Association Rule: Tea → Coffee

Confidence= P(Coffee|Tea) = 0.75 (75% of those who drink tea also drink coffee)

but P(Coffee) = 0.9 (90% of the people in our sample drink coffee (most of them do!))

⇒ Although confidence is high, rule is misleading

⇒ P(Coffee|NotTea) = 0.9375 (more interesting/ meaningful that nearly 94% of those who do not drink tea, drink coffee)

⇒ One is more likely to drink coffee if they do not drink tea (than if they do drink tea)

Slide courtesy of Tan,Steinbach, Kumar, Introduction to Data Mining

Additional References

R1 Data Mining: Concepts and Techniques by Han, Kamber and Pei (Morgan Kaufman)

Introduction to Data Mining by Tan, Steinbach and Kumar (Pearson – First Edition) Chapters 6 and 7

THANK YOU

Jyothi R.
Assistant Professor,
Department of Computer Science
jyothir@pes.edu