Data Science School at DKE
Association Rules

Mirela Popa
Postdoctoral Researcher
Department of Data Science and Knowledge Engineering (DKE)

Maastricht University, June 2019

Overview

- Association Rule Problem
- Apriori Algorithm (FP-Growth Algorithm)
- Rule Generation
- Measures for Association Rules
- Relationship with data mining domain
- Applications in different domains

Association Rule Mining

- Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

Market-Basket transactions

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**

Example of Association Rules

{Diaper} $\rightarrow$ {Beer},
{Milk, Bread} $\rightarrow$ {Eggs,Coke},
{Beer, Bread} $\rightarrow$ {Milk},

Implication means co-occurrence, not causality!

Applications

- **Market Basket Analysis:** given a database of customer transactions, where each transaction is a set of items the goal is to find groups of items which are frequently purchased together.
- **Telecommunication** (each customer is a transaction containing the set of phone calls)
- **Credit Cards/ Banking Services** (each card/account is a transaction containing the set of customer’s payments)
- **Medical Treatments** (each patient is represented as a transaction containing the ordered set of diseases)
- **Basketball-Game Analysis** (each game is represented as a transaction containing the ordered set of ball passes)

Motivation

(a) discovering patterns from a large database can be computationally expensive,

(b) some of the discovered patterns can be spurious, or even for non-spurious ones, some can be more interesting/valuable from a semantic point of view.

Definition: 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**

Definition: Association Rule

- **Association Rule**
  - An implication expression of the form X → Y, where X and Y are disjoint itemsets (X∩Y=∅)
  - 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

$$\text{Support, } s(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{N};$$

$$\text{Confidence, } c(X \longrightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}.$$

Example:

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

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

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

Association Rule Mining Task

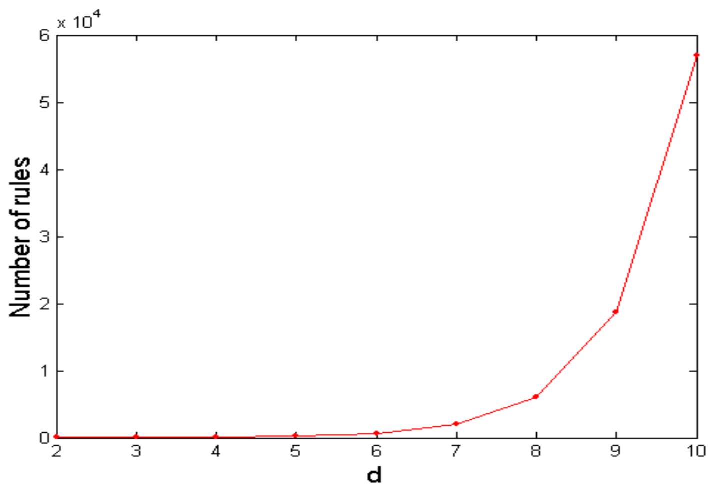
- Given a set of transactions T, the goal of association rule mining is to find all rules having
  - support ≥ *minsup* threshold
  - confidence ≥ *minconf* threshold

Association Rule Mining Task

- Brute-force approach:
  - List all possible association rules
  - Compute the support and confidence for each rule
  - Prune rules that fail the *minsup* and *minconf* thresholds

  $\Rightarrow$ Computationally prohibitive!

- Note that given d unique items:
  - Total number of itemsets = 2^d
  - Total number of possible association rules:



$$R = \sum_{k=1}^{d-1}\left[\binom{d}{k} \times \sum_{j=1}^{d-k}\binom{d-k}{j}\right] = 3^d - 2^{d+1} + 1$$

If d=6, R = 602 rules

How to make Efficient Mining Association Rules

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**

Example of Rules:

{Milk,Diaper} $\rightarrow$ {Beer} (s=0.4, c=0.67)
{Milk,Beer} $\rightarrow$ {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} $\rightarrow$ {Milk} (s=0.4, c=0.67)
{Beer} $\rightarrow$ {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} $\rightarrow$ {Milk,Beer} (s=0.4, c=0.5)
{Milk} $\rightarrow$ {Diaper,Beer} (s=0.4, c=0.5)

Observations:

- All the above rules are binary partitions of the same itemset: {Milk, Diaper, Beer}
- Rules originating from the same itemset have identical support but can have different confidence
- ***Thus, we may decouple the support and confidence requirements!***

Mining Association Rules: Problem Decomposition

- Two-step approach:
  1. Frequent Itemset Generation
     – Generate all itemsets whose support $\geq$ 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

Mining Association Rules: Problem Decomposition

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.

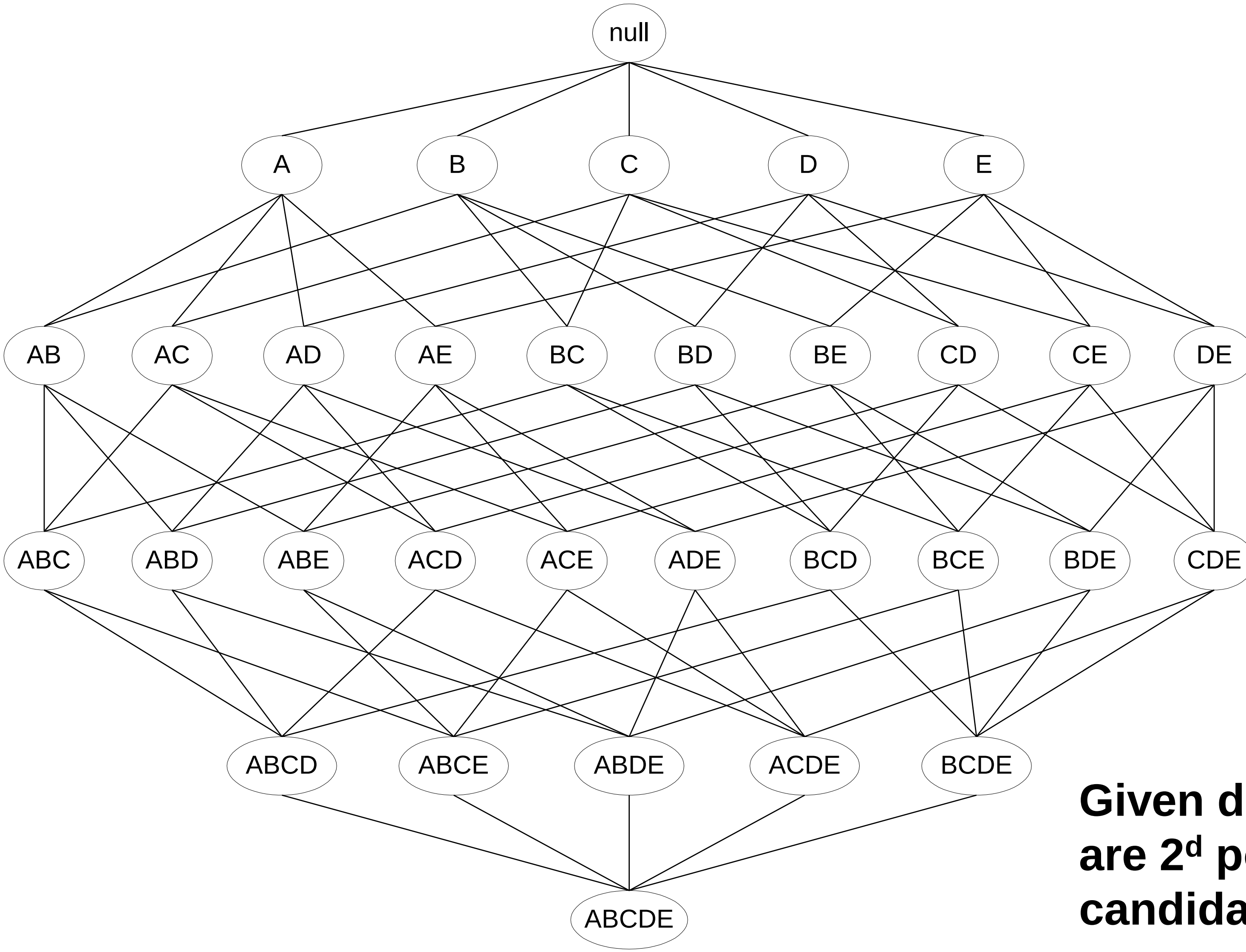
Frequent Itemset	Support
{Shoes}	75%
{Shirt}	50%
{Jacket}	50%
{Shoes, Jacket}	50%

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%

Frequent Itemset Generation

Frequent Itemset Generation: Complexity



Given d items, there are 2^d possible candidate itemsets

Frequent Itemset Generation: Complexity

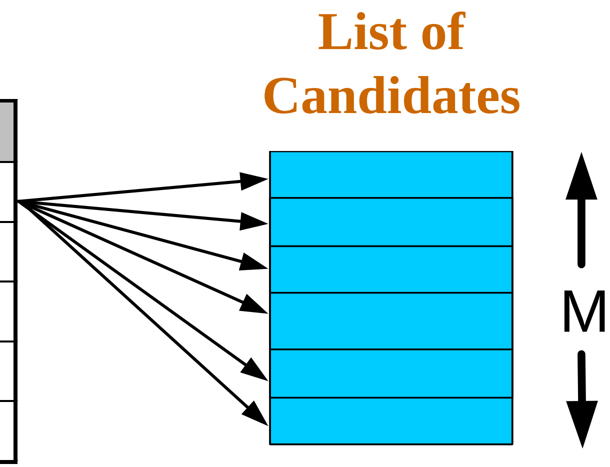
- Brute-force approach:
  - Each itemset in the lattice is a candidate frequent itemset
  - Count the support of each candidate by scanning the database

Transactions

N

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

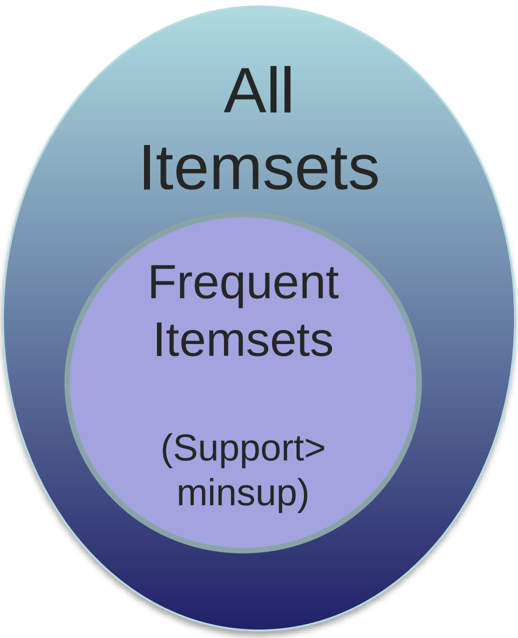
w



  - Match each transaction against every candidate
  - Complexity ~ O(NMw) => Expensive since $M = 2^d$!!!

Frequent Itemset Generation Strategies

- Reduce the number of candidates(M)
  - Complete search: $M=2^d$
  - Use pruning techniques to reduce M
- Reduce the number of transactions(N)
  - Reduce size of N as the size of itemset increases
  - Used by vertical-based mining algorithms



Reducing Number of Candidates

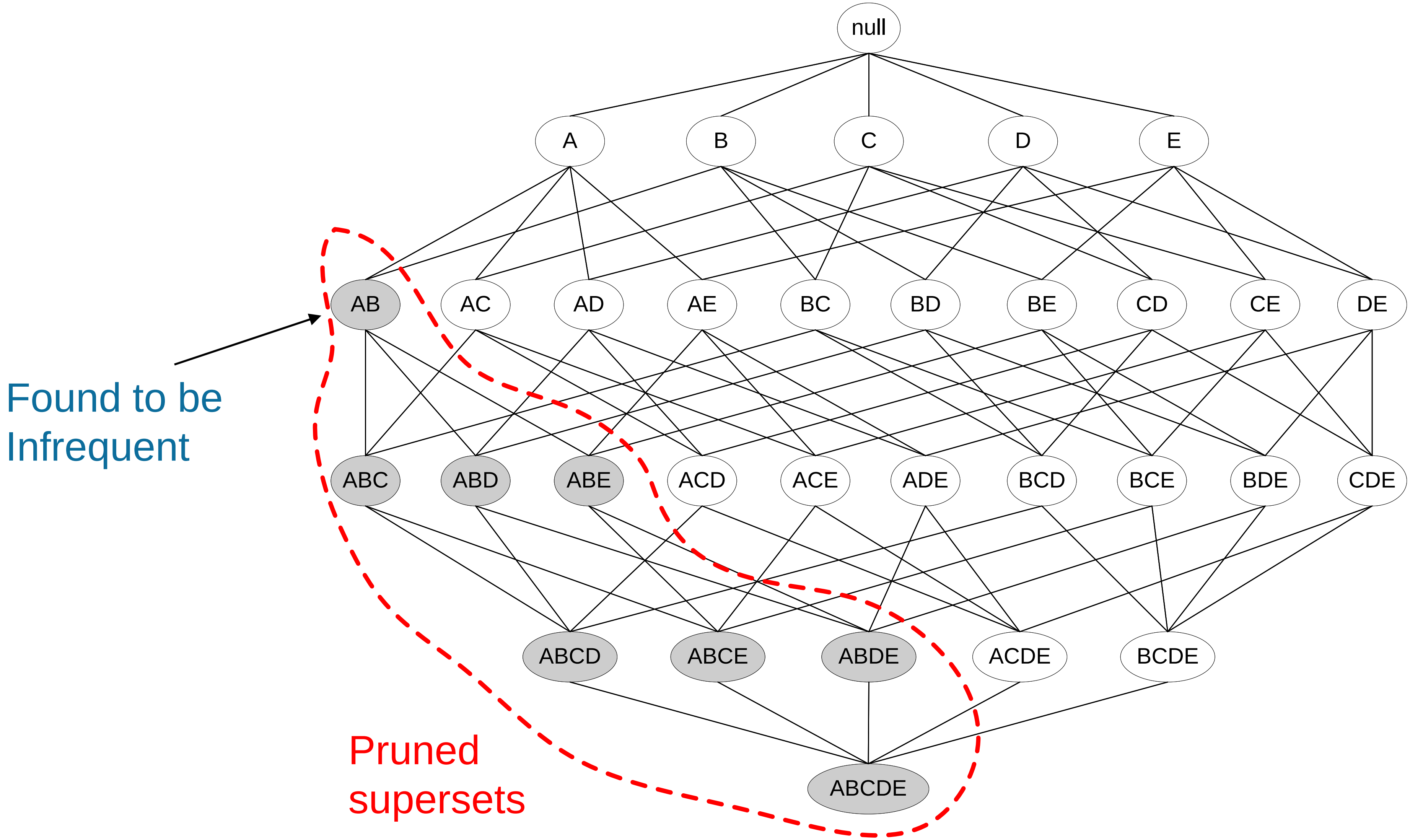
- Apriori principle:
  - If an itemset is frequent, then all of its subsets must also be frequent

- 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

Illustrating Apriori Principle



Apriori Algorithm

- 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

The Apriori Algorithm — Example

Min support =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

Candidate Generation

- An efficient generation procedure must be complete and non-redundant and should avoid generating too many unnecessary candidates.

- Methods: brute-force method, $\mathbf{L_{k-1} \times L_1}$ method (combine frequent k-1 itemsets with frequent 1-itemset), $\mathbf{L_{k-1} \times L_{k-1}}$ method – avoid generating duplicate itemsets, by sorting the items in their lexicographic order).

How to Generate Candidates (L_{k-1} x L_{k-1}) method

Input: L_{i-1} : set of frequent itemsets of size i-1

Output: C_i : set of candidate itemsets of size i

C_i = empty set;

for each itemset J in L_{i-1} **do**

  for each itemset K in L_{i-1} s.t. K<> J **do**

    if i-2 of the elements in J and K are equal **then**

      if all subsets of $\{K \cup J\}$ are in L_{i-1} **then**

        $C_i = C_i \cup \{K \cup J\}$

return C_i;

Example of Generating Candidates

- L_3={*abc, abd, acd, ace, bcd*}
- Generating C_4 from L_3
  - *abcd* from *abc* and *abd*
  - *acde* from *acd* and *ace*
- Pruning:
  - *acde* is removed because *ade* is not in L_3
- C_4={*abcd*}

Support Counting

- Comparing each transaction against every candidate itemset is computationally expensive, an alternative approach is to enumerate the itemsets contained in each transaction.

- In the next example, all the 3-itemsets contained in t are obtained using a systematic approach.

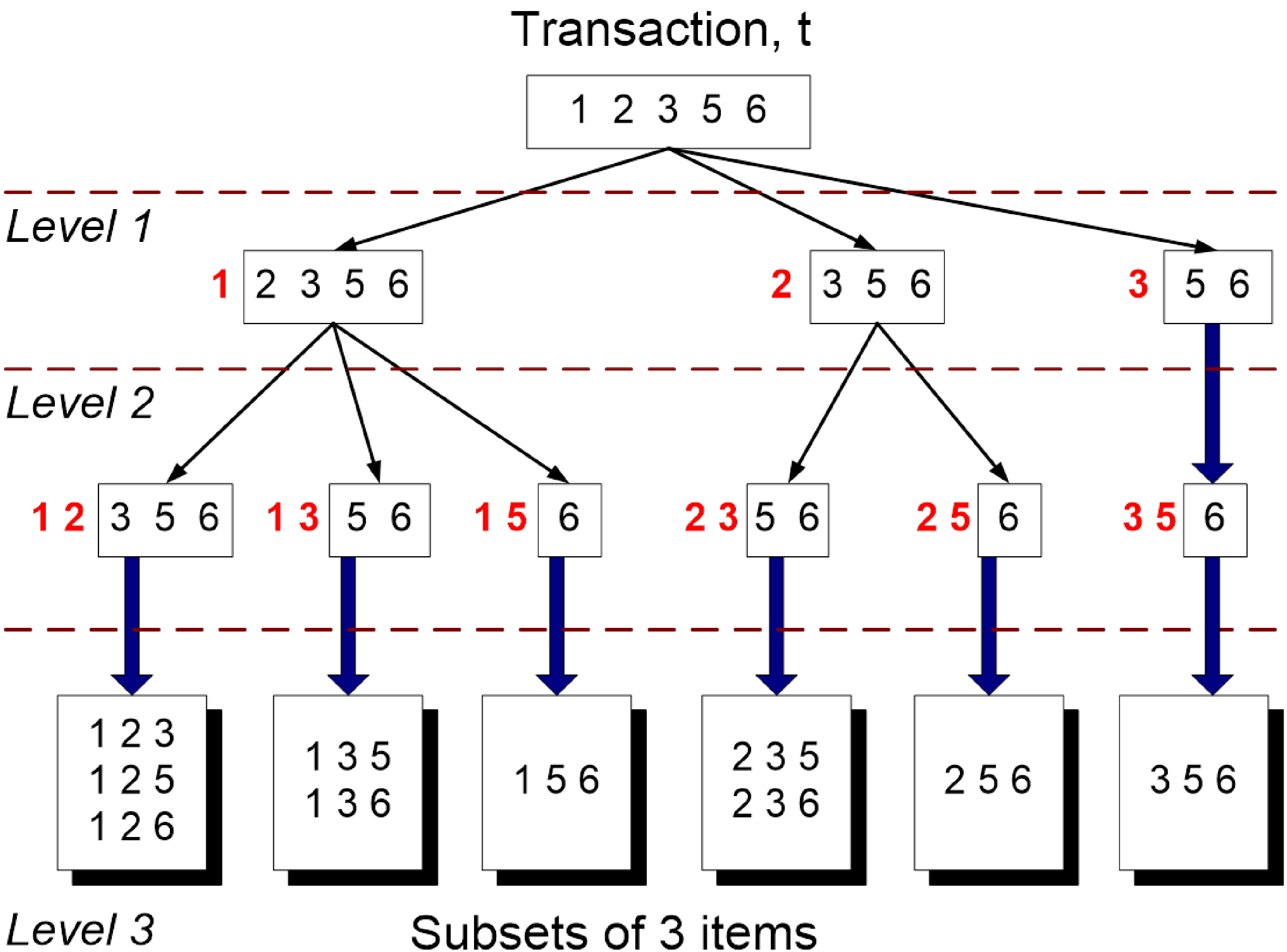
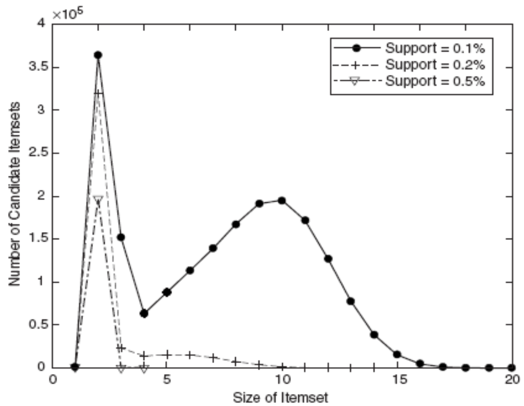
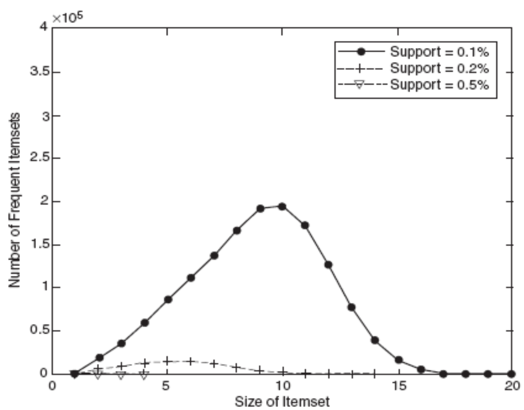


Image from [1], Chapter 5 Association Analysis

Experiment Results

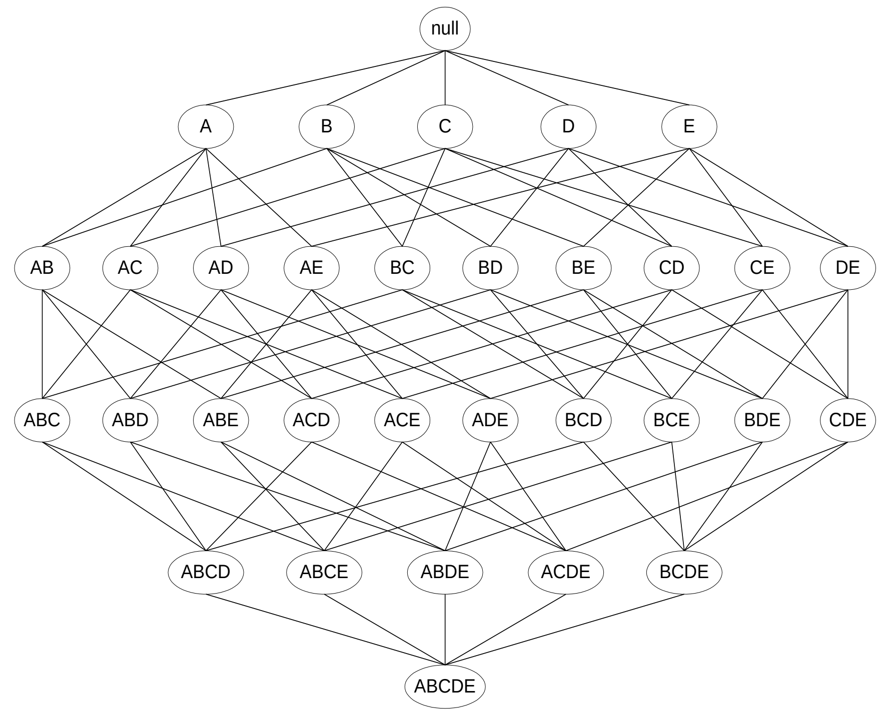


(a) Number of candidate itemsets.



(b) Number of frequent itemsets.

Figure 6.13. Effect of support threshold on the number of candidate and frequent itemsets.



Rule Generation

- Given a frequent itemset L, find all non-empty subsets $f \subset L$ such that $f \rightarrow L - f$ satisfies the minimum confidence requirement
  - If {A,B,C,D} is a frequent itemset, candidate rules:

    $ABC \rightarrow D$, $ABD \rightarrow C$, $ACD \rightarrow B$, $BCD \rightarrow A$,
    $A \rightarrow BCD$, $B \rightarrow ACD$, $C \rightarrow ABD$, $D \rightarrow ABC$
    $AB \rightarrow CD$, $AC \rightarrow BD$, $AD \rightarrow BC$, $BC \rightarrow AD$,
    $BD \rightarrow AC$, $CD \rightarrow AB$,

- If $|L| = k$, then there are $2^k - 2$ candidate association rules (ignoring $L \rightarrow \emptyset$ and $\emptyset \rightarrow L$)

Rule Generation: Brute Force Approach

for each frequent itemset I **do**
  for each subset C of I **do**
    if (support(I) / support(I - C) >= minconf) **then**
      output the rule (I - C) $\Rightarrow$ C,
      with confidence = support(I) / support (I - C)
        and support = support(I)

Rule Generation Example: Brute Force Approach

TID	**List of Item_IDs**
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Let use consider the 3-itemset {I1, I2, I5} with support of 0.22(2)%. Let generate all the association rules from this itemset:

$I1 \wedge I2 \Rightarrow I5$ *confidence* = 2/4 = 50%

$I1 \wedge I5 \Rightarrow I2$ *confidence* = 2/2 = 100%

$I2 \wedge I5 \Rightarrow I1$ *confidence* = 2/2 = 100%

$I1 \Rightarrow I2 \wedge I5$ *confidence* = 2/6 = 33%

$I2 \Rightarrow I1 \wedge I5$ *confidence* = 2/7 = 29%

$I5 \Rightarrow I1 \wedge I2$ *confidence* = 2/2 = 100%

Efficient Rule Generation

- How to efficiently generate rules from frequent itemsets?
  - The 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

Efficient Rule Generation

Theorem. Consider a non-empty itemset Y and a non-empty itemset $X \subseteq Y$. Then:

$$c(X \rightarrow Y \setminus X) \geq c(X' \rightarrow Y \setminus X')$$

$$where \qquad X' \subseteq X.$$

Proof:

$$c(X \rightarrow Y \setminus X) = \frac{\sigma(Y)}{\sigma(X)} \qquad and$$

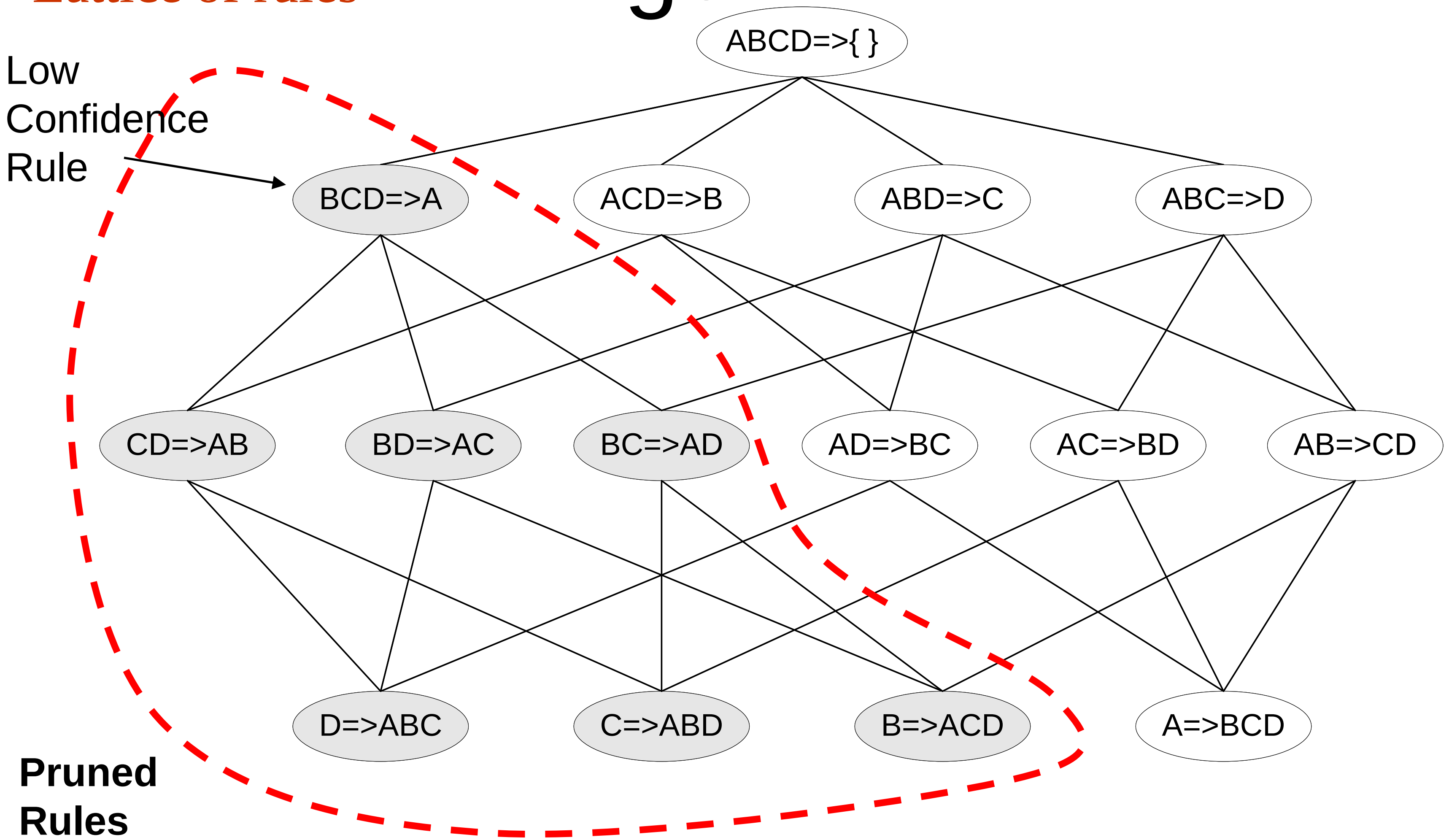
$$c(X' \rightarrow Y \setminus X') = \frac{\sigma(Y)}{\sigma(X')}.$$

$$But, \qquad \sigma(X) \leq \sigma(X').Thus,$$

$$c(X \rightarrow Y \setminus X) \geq c(X' \rightarrow Y \setminus X').$$

Rule Generation for Apriori Algorithm

Lattice of rules



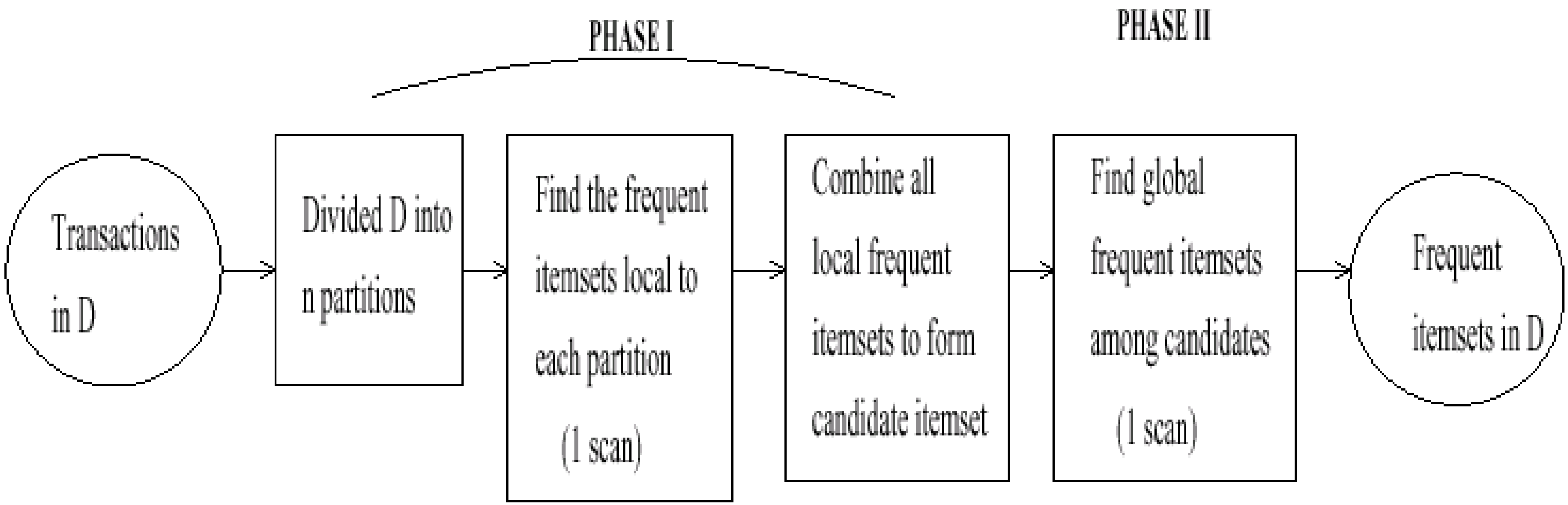
Factors Affecting Complexity

- Choice of minimum support threshold
  - lowering support threshold results in more frequent itemsets
  - this may increase number of candidates and max length of frequent itemsets
- Dimensionality (number of items) of the data set
  - more space is needed to store support count
  - if number of frequent items also increases, both computation and I/O costs may also increase
- Size of database
  - since Apriori makes multiple passes, run time of algorithm may increase with number of transactions
- Average transaction width increases max length of frequent itemsets

Further Improvement of the Apriori Method

- Major computational challenges
  - Multiple scans of transaction database
  - Huge number of candidates
  - Tedious workload of support counting for candidates
- Improving Apriori: general ideas
  - Reduce passes of transaction database scans
  - Shrink number of candidates
  - Reduce data size

Partitioning



Transaction reduction

A transaction that does not contain any frequent *k*-itemset will not contain frequent l-itemset for $l > k$! Thus, it is useless in subsequent scans!

Sampling

Mining on a subset of given data, lower support threshold + a method to determine the completeness

Alternative methods for the Apriori Algorithm

- *General-to-Specific* versus *Specific-to-General* (the Apriori alg. uses a general-to-specific search strategy, while a specific-to-general strategy is useful at discovering maximal frequent itemsets in dense transactions), or a combination of the two approaches which can help to rapidly identify the frequent itemset border.

Frequent Itemset Search

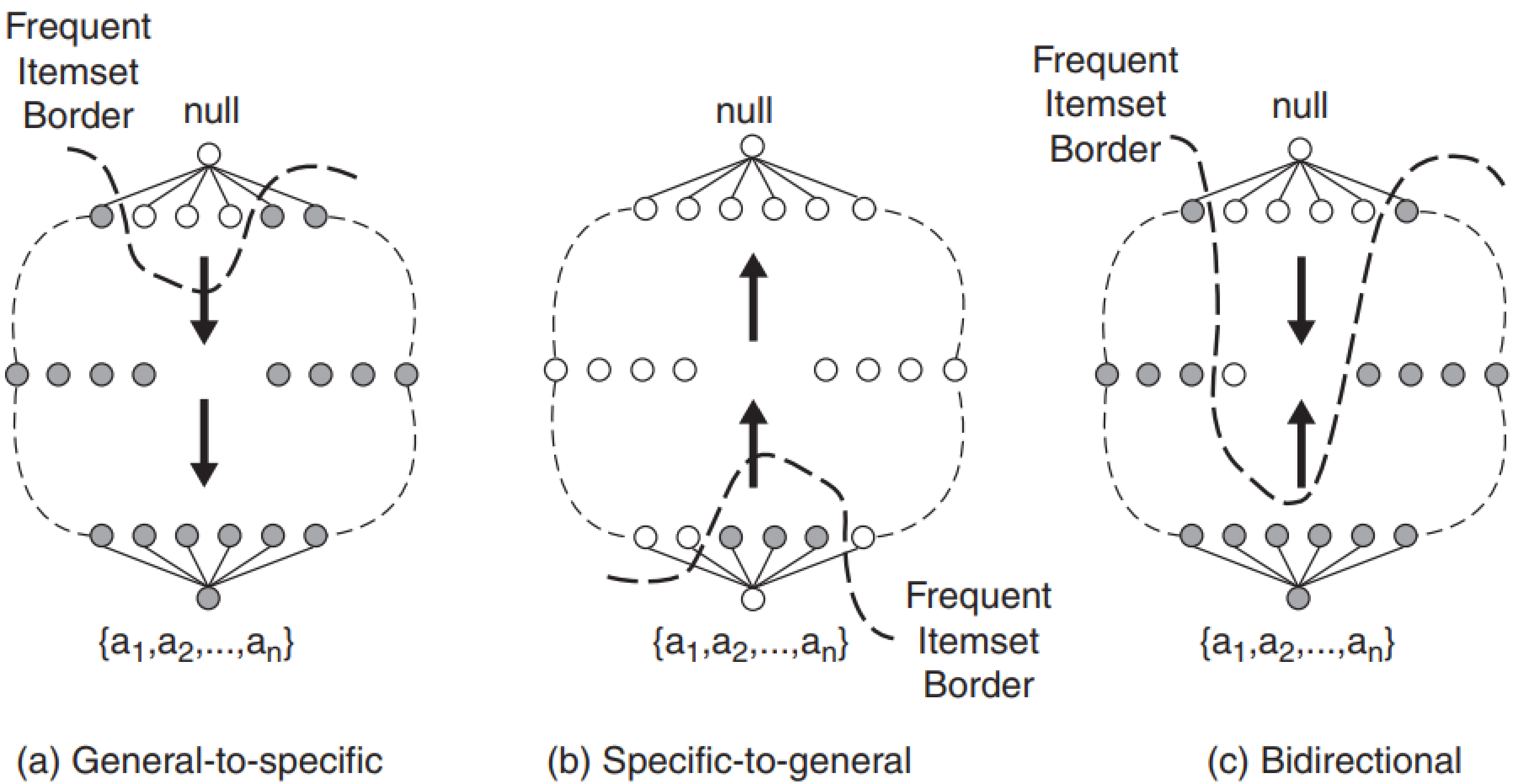


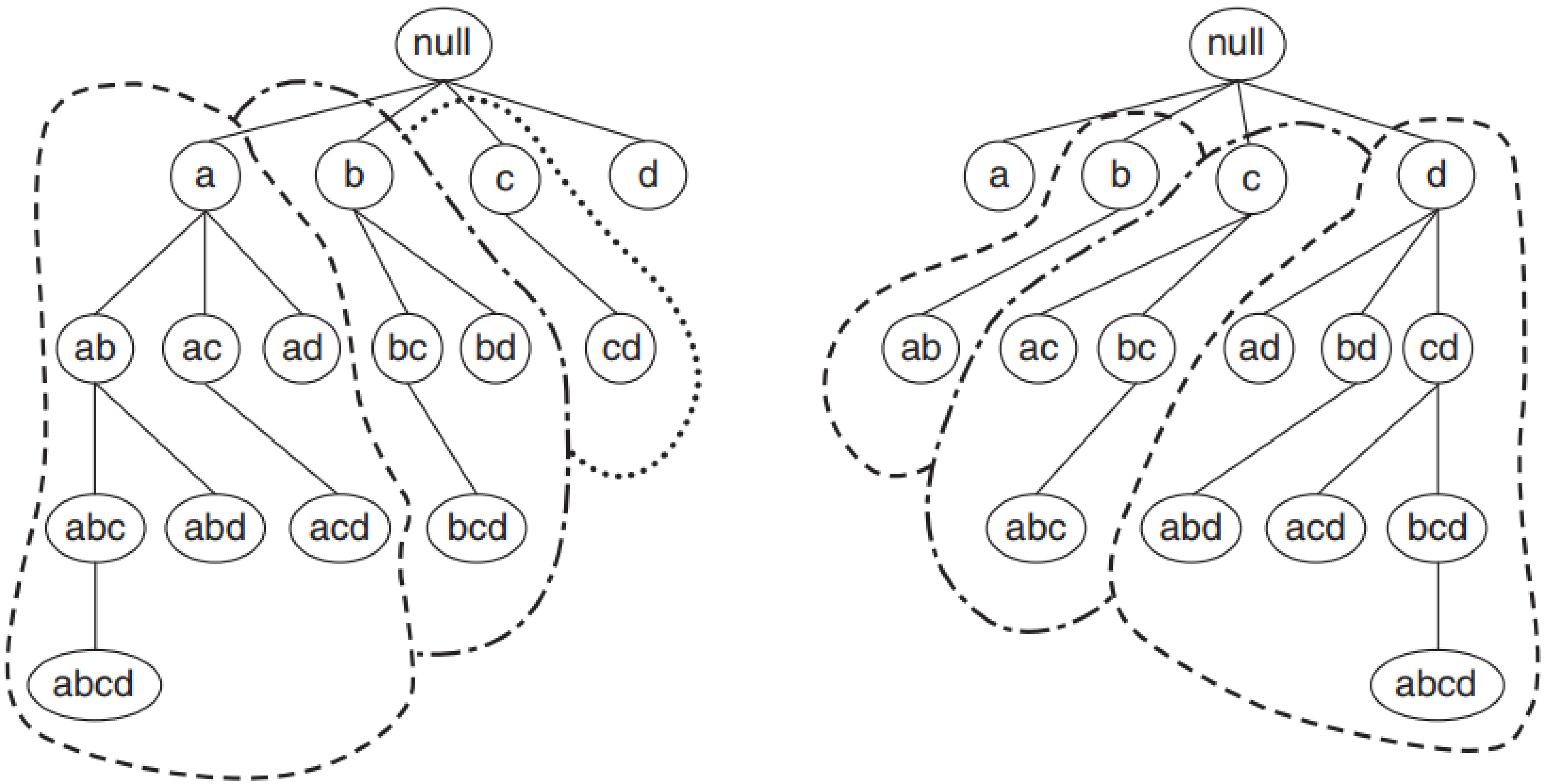
Image from [1], Chapter 5 Association Analysis

Alternative methods for the Apriori Algorithm

- *General-to-Specific* versus *Specific-to-General* (the Apriori alg. uses a general-to-specific search strategy, while a specific-to-general strategy is useful at discovering maximal frequent itemsets in dense transactions), or a combination of the two approaches which can help to rapidly identify the frequent itemset border.

- *Equivalent classes* – first partition the lattice into disjoint group of nodes and perform the search in each of them

Equivalence Classes Example



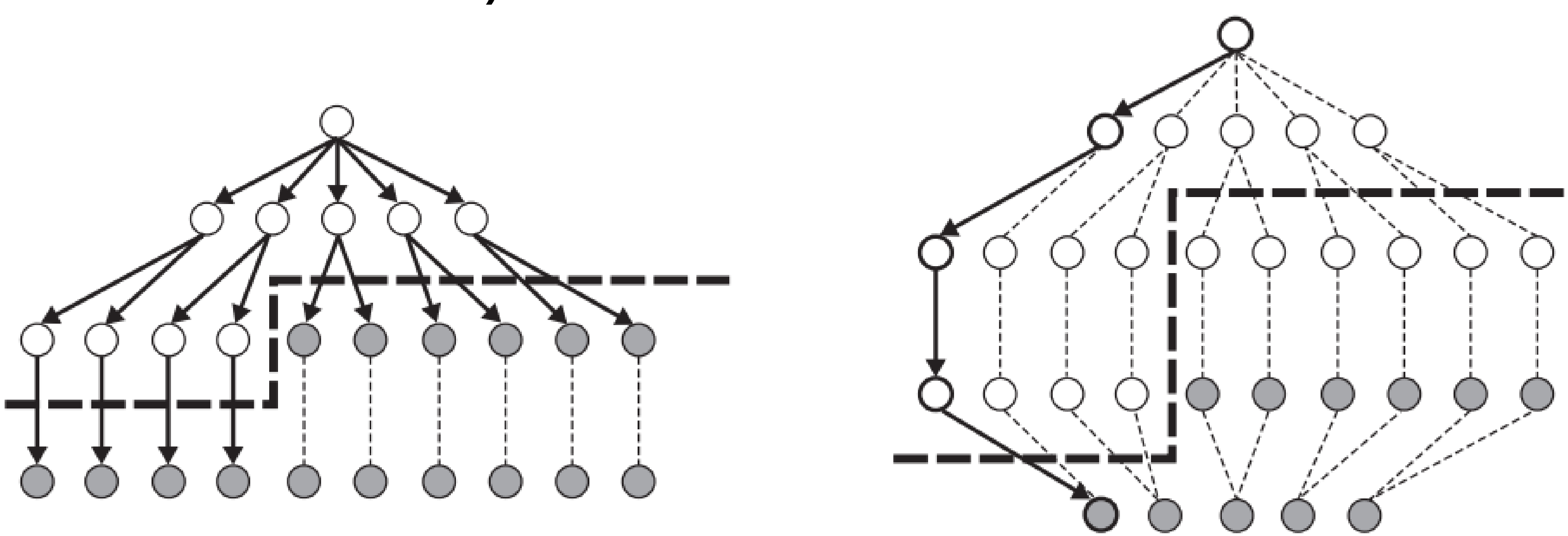
(a) Prefix tree.

(b) Suffix tree.

Image from [1], Chapter 5 Association Analysis

Alternative methods for the Apriori Algorithm

- *Breadth-First versus Depth-First* (the Apriori alg. uses a breadth-first manner, while a depth-first approach enables a faster detection of the frequent itemset border).



(a) Breadth first

(b) Depth first

Dataset Representation

- The transactions in a dataset can use a horizontal or a vertical data layout.

Horizontal
Data Layout

TID	Items
1	a,b,e
2	b,c,d
3	c,e
4	a,c,d
5	a,b,c,d
6	a,e
7	a,b
8	a,b,c
9	a,c,d
10	b

Vertical Data Layout

a	b	c	d	e
1	1	2	2	1
4	2	3	4	3
5	5	4	5	6
6	7	8	9	
7	8	9		
8	10			
9				

FP-Growth Algorithm

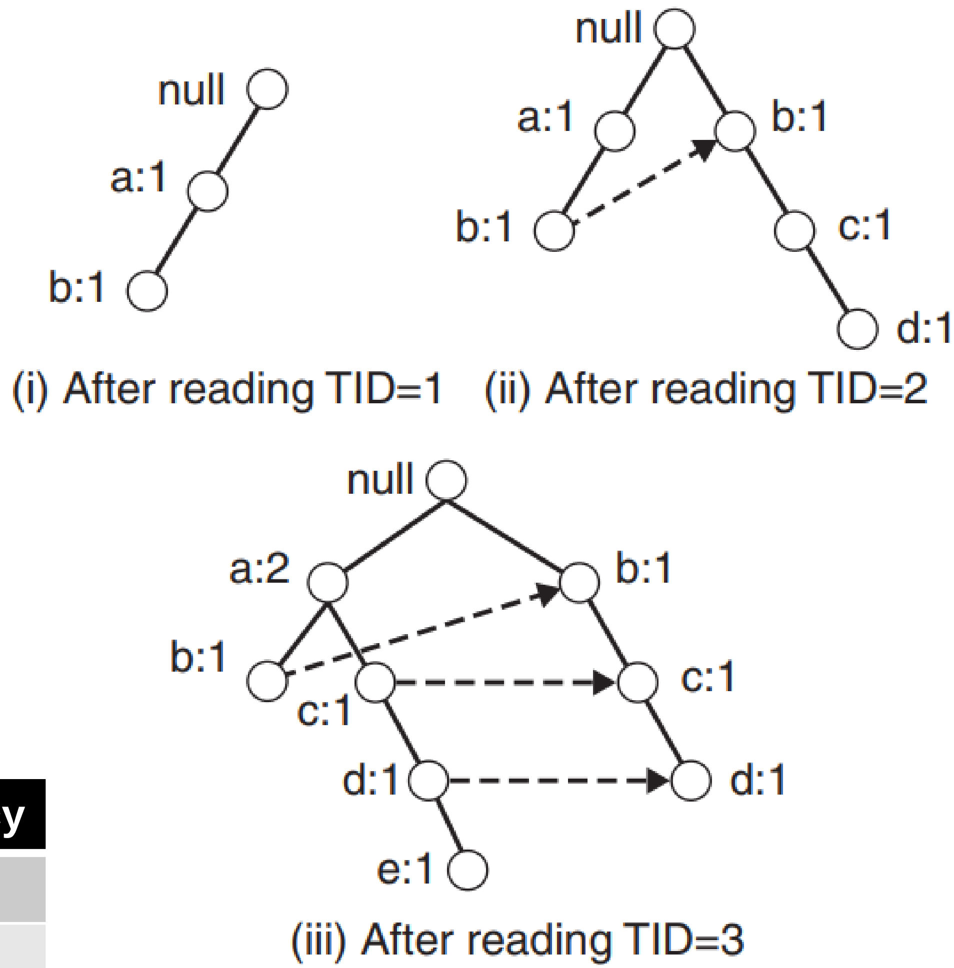
- An alternative approach of discovering frequent itemsets. It encodes the data using a FP-tree data structure from which it extracts the frequent itemsets.
- The FP-tree is constructed by:
  a. Scan DB once, find frequent 1-itemset
  b. Sort frequent items in frequency descending order
  c. Scan DB again and construct the FP-tree
- The more paths overlaps, the better compression can be achieved.

FP-tree representation

Transaction Data Set

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

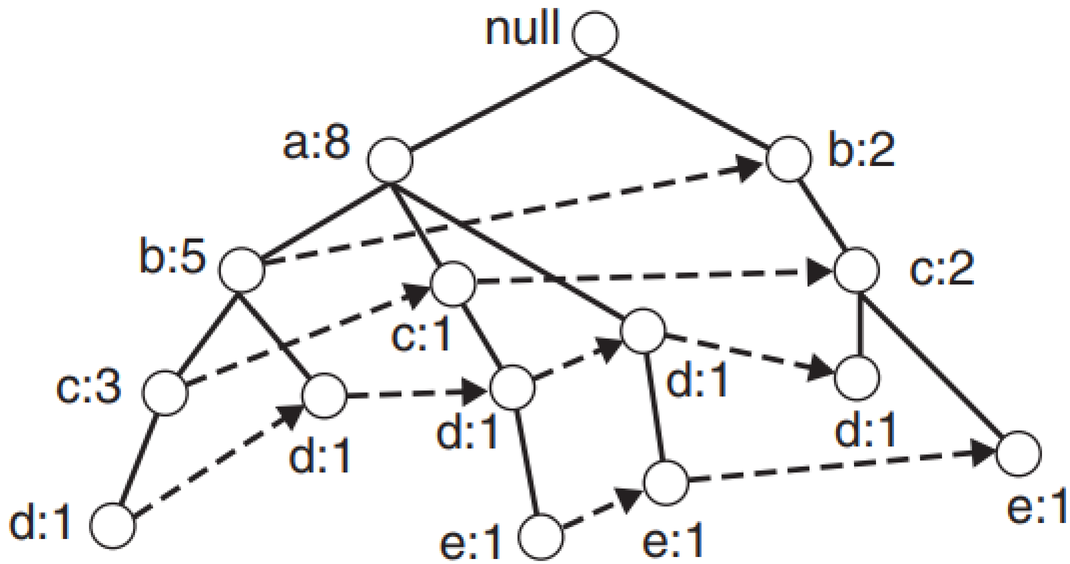
Item	Frequency
a	8
b	7
c	6
d	5
e	3



FP-tree representation

Transaction
Data Set

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}



(iv) After reading TID=10

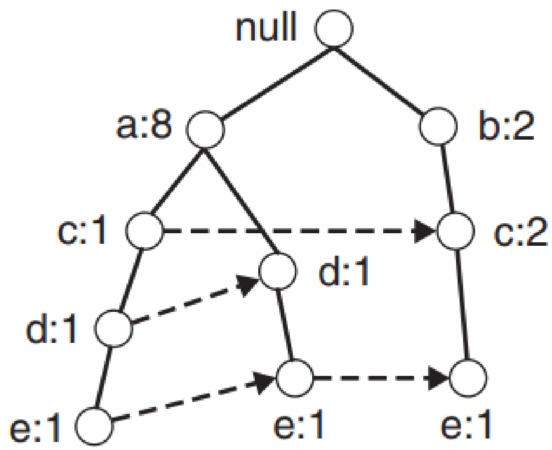
Frequent Itemset Generation in FP-growth algorithm

- It uses a bottom-up method, looking for frequent itemsets ending in e, then d, c, b, and finally a, by examining the corresponding paths.

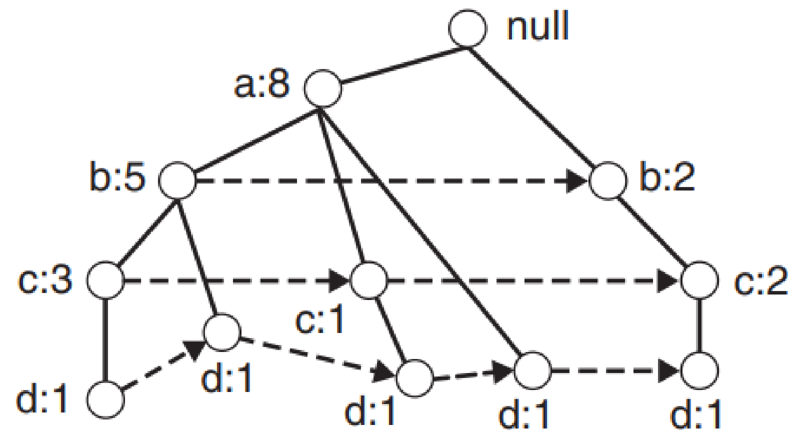
- This strategy (divide-and-conquer) is similar to the suffix-based approach.

- The advantage of FP-tree representation is given by the rapid access to each path, using associated pointers and reduced memory usage due to the compact representation, resulting in improved performance.

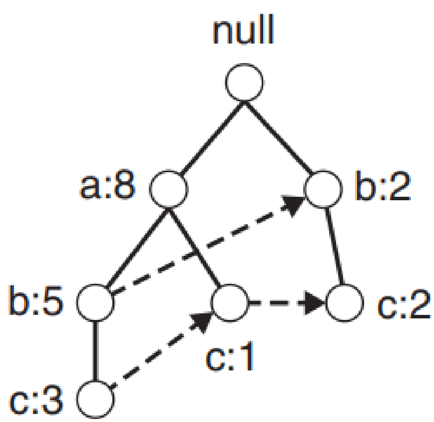
Finding Frequent Itemsets



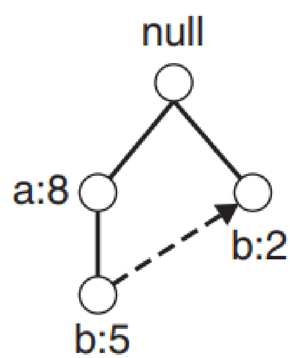
(a) Paths containing node e



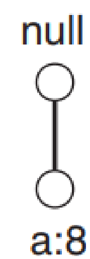
(b) Paths containing node d



(c) Paths containing node c



(d) Paths containing node b

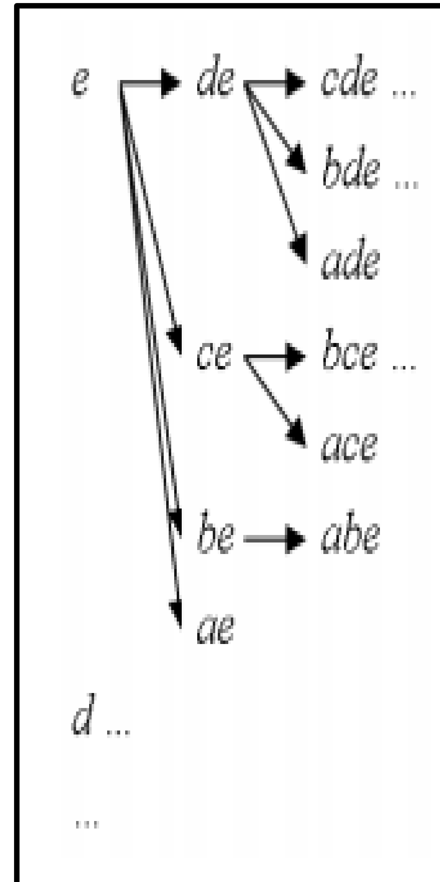


(e) Paths containing node a

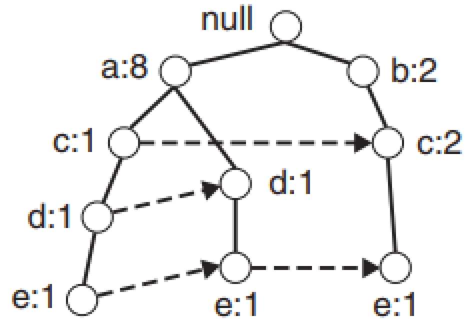
Finding Frequent Itemsets

- Each prefix path sub-tree is processed recursively to extract the frequent itemsets. Solutions are then merged.
- Build a Conditional FP-tree on each node (consider only the transactions containing a particular itemset – and then removing that itemset from all transactions).

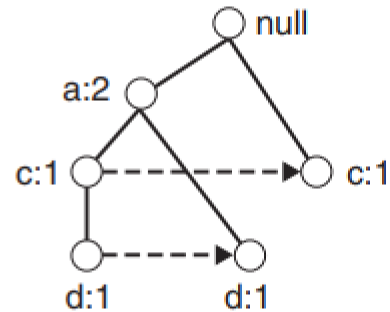
TID	Items
~~1~~	~~{a,b}~~
~~2~~	~~{b,c,d}~~
3	{a,c,d,~~e~~}
4	{a,d,~~e~~}
~~5~~	~~{a,b,c}~~
~~6~~	~~{a,b,c,d}~~
~~7~~	~~{a}~~
~~8~~	~~{a,b,c}~~
~~9~~	~~{a,b,d}~~
10	{b,c,~~e~~}



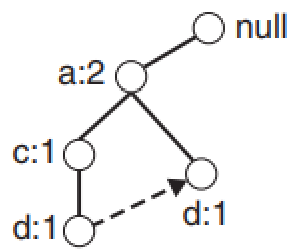
Conditional FP-tree



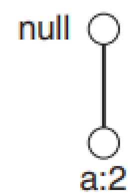
(a) Prefix paths ending in e



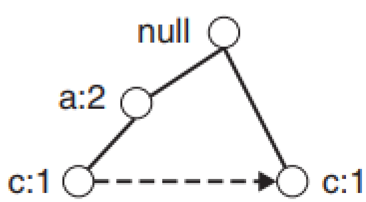
(b) Conditional FP-tree for e



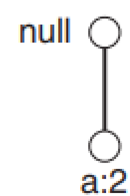
(c) Prefix paths ending in de



(d) Conditional FP-tree for de



(e) Prefix paths ending in ce



(f) Prefix paths ending in ae

Obtained frequent Itemsets

Transaction Data Set

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

Suffix	Frequent Itemsets
a	{a}
b	{b}, {a,b}
c	{c}, {b,c}, {a,b,c},{a,c}
d	{d}, {c,d},{b,c,d},{a,c,d},{b,d},{a,b,d},{a,d}
e	{e},{d,e},{a,d,e},{c,e},{a,e}

Evaluation of Association Patterns

- Establish criteria for evaluating the quality of the association patterns:
  - Data-driven approach - objective interestingness measures for ranking the discovered patterns, using statistical criteria (e.g. support, confidence, correlation)
  - Subjective arguments, require domain knowledge

Objective Measures of Interestingness

- Limitations of the Support-Confidence Framework

	$Coffee$	$\overline{Coffee}$	
Tea	150	50	200
$\overline{Tea}$	650	150	800
	800	200	1000

c(Tea → Coffee) = 150/200 = 75%
s(Coffee) = 800/1000 = 80%

	$Honey$	$\overline{Honey}$	
Tea	100	100	200
$\overline{Tea}$	20	780	800
	120	880	1000

c(Tea → Honey) = 100/200 = 50%
s(Honey) = 120/1000 = 12%
c(¬Tea →Honey) = 20/800 = 2.5%

Alternative Measures for Association Rules

- The **confidence** of $X \Rightarrow Y$ in database D is the ratio of the number of transactions containing $X \cup Y$ to the number of transactions that contain X. In other words it is:

$$conf(X \rightarrow Y) = \frac{\frac{\sigma(X \cup Y)}{|D|}}{\frac{\sigma(X)}{|D|}} = \frac{p(X \wedge Y)}{p(X)} = p(Y \mid X)$$

- But, when Y is independent of X: $p(Y) = p(Y \mid X)$. In this case if $p(Y)$ is high we'll have a rule with high confidence that associate independent itemsets! For example, if p(*"buy milk"*) = 80% and *"buy milk"* is independent from *"buy salmon"*, then the rule *"buy salmon"* $\Rightarrow$ *"buy milk"* will have confidence 80%!

Objective Measures of Interestingness

- Limitations of the Support-Confidence Framework – the support of two variables X,Y occurring together is not considering the case of independence between them, which could support better patterns discovery

Alternative Measures for Association Rules

- The **lift** measure indicates the departure from independence of *X* and *Y*. The lift of $X \Rightarrow Y$ is :

$$lift(X \to Y) = \frac{conf(X \to Y)}{p(Y)} = \frac{\frac{p(X \wedge Y)}{p(X)}}{p(Y)} = \frac{p(X \wedge Y)}{p(X)\,p(Y)}$$

- But, the lift measure is symmetric; i.e., it does not take into account the direction of implications!
- If lift is greater than 1, then *X* and *Y* are ***positively*** correlated; i.e., the occurrence of *X* (*Y*) imply occurrence of *Y*(*X*).
- If lift is smaller than 1, then *X* and *Y* are ***negatively*** corre-lated; i.e., the occurrence of *X* (*Y*) imply absence of *Y*(*X*).

Piatesky-Shapiro (PS) Measure

$$PS = s(X,Y) - s(X) \times s(Y)$$

PS = 0, if X and Y are mutually independent

PS>0, for a positive relationship between (X,Y)

PS<0, for a negative relationship between (X,Y)

Correlation Analysis

- For continuous variables, can be used the Pearson's correlation coefficient
- For binary variables, the ɵ-coefficient (a normalized version of the PS measure),
- 0 – no relationship,
- 1 – a perfect positive relationship
- -1 – a perfect negative relationship

$$ɵ = \frac{s(X,Y) - s(X)\cdot s(Y)}{\sqrt{s(X)\cdot\big(1 - s(X)\big)\cdot s(Y)\cdot(1 - s(Y))}}$$

Alternative Measures for Association Rules

- The **conviction** measure indicates the departure from independence of *X* and *Y* taking into account the implication direction. The conviction of $X \Rightarrow Y$ is :

$$conv(X \rightarrow Y) = \frac{p(X)\,p(\neg Y)}{p(X \wedge \neg Y)}$$

- It is useful for census data, where many items are very likely to occur with or without other items.

Alternative objective measures

	B	$\overline{B}$	
A	f_{11}	f_{10}	f_{1+}
$\overline{A}$	f_{01}	f_{00}	f_{0+}
	f_{+1}	f_{+0}	N

Measure (Symbol)	Definition
Correlation (ϕ)	$\frac{Nf_{11}-f_{1+}f_{+1}}{\sqrt{f_{1+}f_{+1}f_{0+}f_{+0}}}$
Odds ratio (α)	$(f_{11}f_{00})/(f_{10}f_{01})$
Kappa (κ)	$\frac{Nf_{11}+Nf_{00}-f_{1+}f_{+1}-f_{0+}f_{+0}}{N^2-f_{1+}f_{+1}-f_{0+}f_{+0}}$
Interest (I)	$(Nf_{11})/(f_{1+}f_{+1})$
Cosine (IS)	$(f_{11})/(\sqrt{f_{1+}f_{+1}})$
Piatetsky-Shapiro (PS)	$\frac{f_{11}}{N} - \frac{f_{1+}f_{+1}}{N^2}$
Collective strength (S)	$\frac{f_{11}+f_{00}}{f_{1+}f_{+1}+f_{0+}f_{+0}} \times \frac{N-f_{1+}f_{+1}-f_{0+}f_{+0}}{N-f_{11}-f_{00}}$
Jaccard (ζ)	$f_{11}/(f_{1+}+f_{+1}-f_{11})$
All-confidence (h)	$\min\left[\frac{f_{11}}{f_{1+}}, \frac{f_{11}}{f_{+1}}\right]$

Rankings of measures

	ϕ	α	κ	I	IS	PS	S	ζ	h
E_1	1	3	1	6	2	2	1	2	2
E_2	2	1	2	7	3	5	2	3	3
E_3	3	2	4	4	5	1	3	6	8
E_4	4	8	3	3	7	3	4	7	5
E_5	5	7	6	2	9	6	6	9	9
E_6	6	9	5	5	6	4	5	5	7
E_7	7	6	7	9	1	8	7	1	1
E_8	8	10	8	8	8	7	8	8	7
E_9	9	4	9	10	4	9	9	4	4
E_{10}	10	5	10	1	10	10	10	10	10

Properties of symmetric measures

Symbol	Measure	Inversion	Null Addition	Scaling
ϕ	ϕ-coefficient	Yes	No	No
α	odds ratio	Yes	No	Yes
κ	Cohen's	Yes	No	No
I	Interest	No	No	No
IS	Cosine	No	Yes	No
PS	Piatetsky-Shapiro's	Yes	No	No
S	Collective strength	Yes	No	No
ζ	Jaccard	No	Yes	No
h	All-confidence	No	Yes	No
s	Support	No	No	No

Other factors to consider

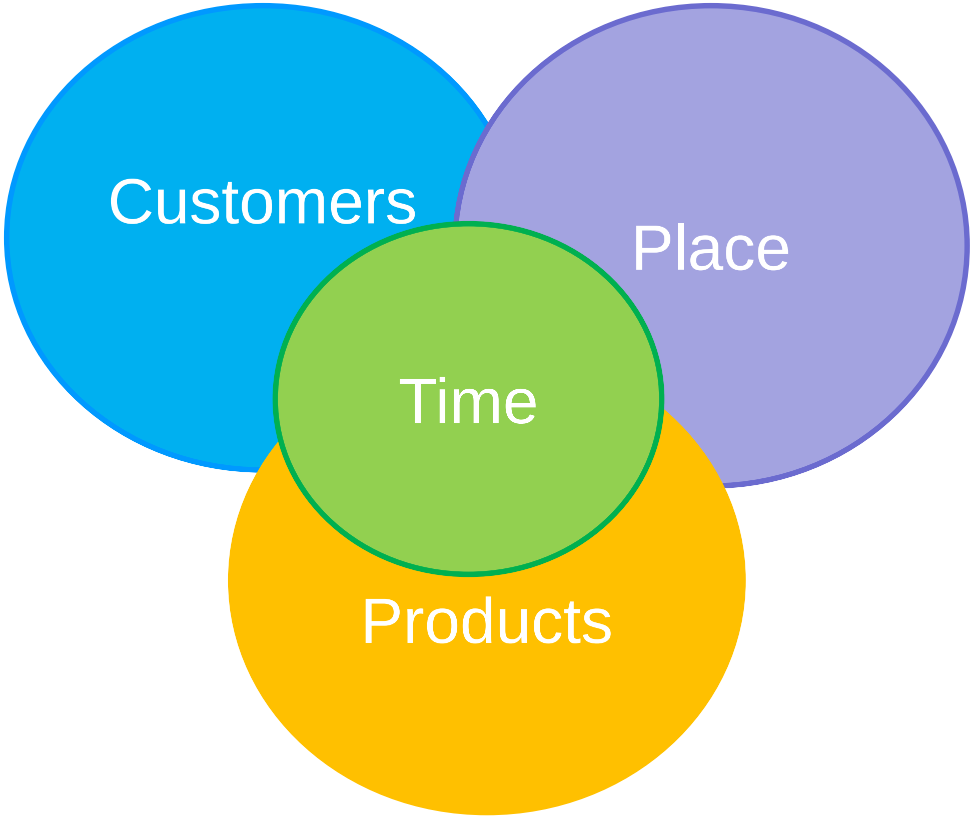
- *Simpson's Paradox* (the relationship between observed variables can be influenced by hidden variables, which can cause the relationship to disappear or to reverse its direction).

- *Effect of skewed support distribution* (most of the items have low to moderate frequencies, while a small number of them have very high frequencies).

Other rule-based patterns

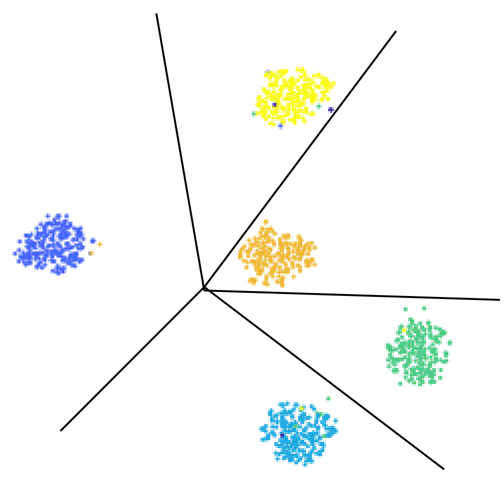
- Profile association rules
- Cyclic association rules
- Fuzzy association rules
- Exception rules
- Negative association rules
- Weighted association rules

Cyclic Association Rules



Data Mining Tasks

- Predictive models (Classification, Regression) – supervised learning
- Descriptive models (Clustering, **Association Rules**) – unsupervised learning



Applications – Consumer Behaviour

	Best rules found	conf.
1	Correct indication of products (most of all the prices)=e Quality of merchandise=e 128 ⇒ Fresh products=e 124	0.97
2	Cleanliness and store layout=e Quality of merchandise=e 136 ⇒ Fresh products=e 131	0.96
3	Quality of merchandise=e 192 ⇒ Fresh products=e 183	0.95
4	Cleanliness and store layout=e Fresh products=e 140 ⇒ Quality of merchandise=e 131	0.94
5	Cleanliness and store layout=e 152 ⇒ Fresh products=e 140	0.92
6	Correct indication of products (most of all the prices)=e Fresh products=e 135 ⇒ Quality of merchandise=e 124	0.92
7	Cleanliness and store layout=e 152 ⇒ Quality of merchandise=e 136	0.89
8	Fresh products=e 206 ⇒ Quality of merchandise=e 183	0.89
9	Easy orientation inside the store (easy to find merchandise)=e 131 ⇒ Fresh products=e 116	0.89
10	Correct indication of products (most of all the prices)=e 154 ⇒ Fresh products=e 135	0.88
11	Easy orientation inside the store (easy to find merchandise)=e 131 ⇒ Correct indication of products (most of all the prices)=e 113	0.86
12	Cleanliness and store layout=e 152⇒ Fresh products=e Quality of merchandise=e 131	0.86
13	Correct indication of products (most of all the prices)=e 154 ⇒ Quality of merchandise=e 128	0.83
14	Correct indication of products (most of all the prices)=e 154 ⇒ Fresh products=e Quality of merchandise=e 124	0.81

Image from [3], Exploring Consumer Behaviour, page 7, data from 1127 respondents

User digital behaviour

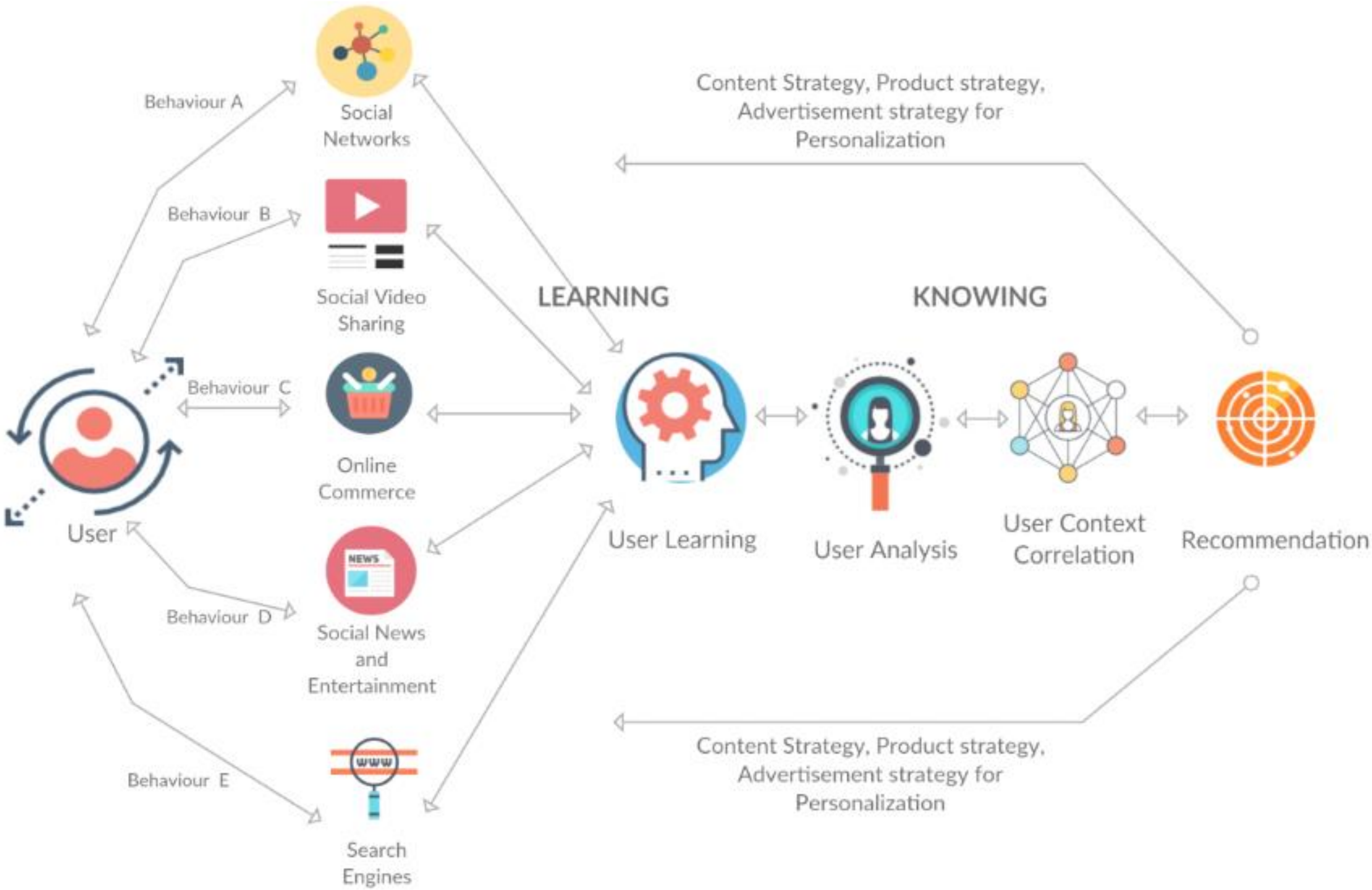


Image from: http://blog.else-corp.com/2017/04/artificial-intelligence-fuels-business-transformations-by-learning-about-and-knowing-users-and-contents-capgemeni/

Association rules in very large clustered domains

- The domain is clustered into groups with a large number of intra-group and a small number of inter-group correlations.

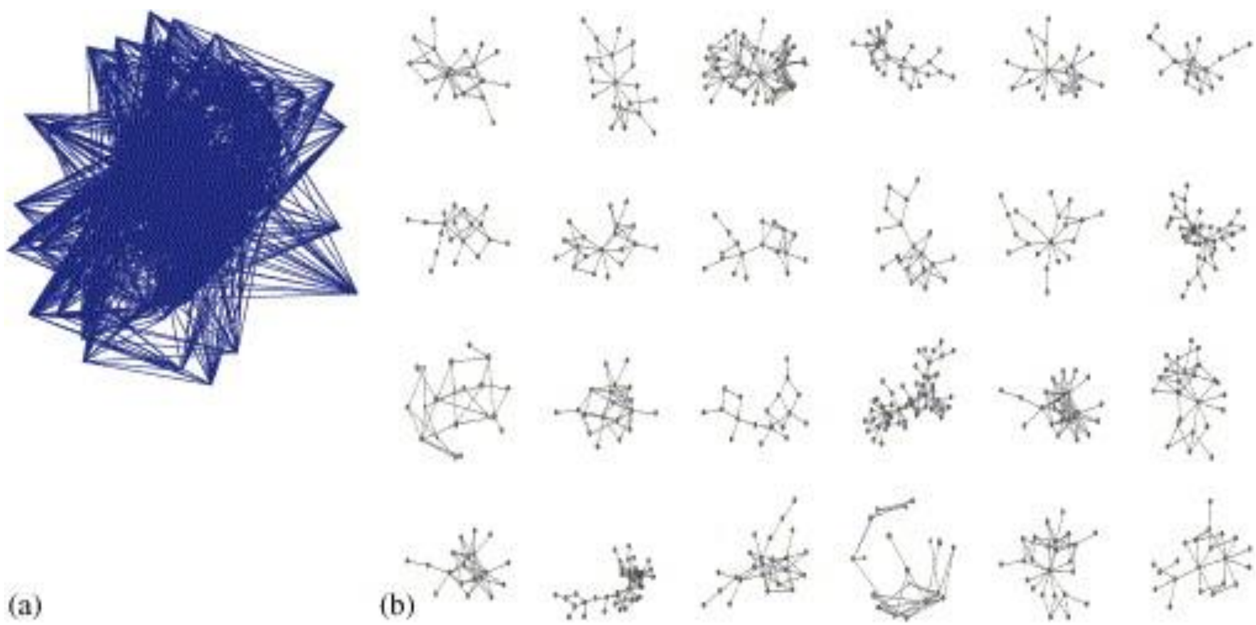


Image from [4]

Medical Diagnosis

- A technique based on relational association rules was proposed in [2] (*Medical Diagnosis using Relational Association Rules*) – determines the probability that a patient characterized by a set of symptoms suffers from a certain disease – the goal is to assist clinicians in the daily practice.

Association Rule Mining for heart disease

Confidence $= 1$:
IF $0 <= AGE < 40.0 \ \ -1.0 <= AL < 0.2 \ \ PCARSUR = n$ THEN $0 <= LAD < 50$, s=0.01 c=1.00 l=2.1
IF $0 <= AGE < 40.0 \ \ -1.0 <= AS < 0.2 \ \ PCARSUR = n$ THEN $0 <= LAD < 50$, s=0.01 c=1.00 l=2.1
IF $40.0 <= AGE < 60.0 \ \ SEX = F \ \ 0 <= CHOL < 200$ THEN $0 <= LCX < 50$, s=0.02 c=1.00 l=1.6
IF $SEX = F \ \ HTA = n \ \ 0 <= CHOL < 200$ THEN $0 <= RCA < 50$, s=0.02 c=1.00 l=1.8
Two items in the consequent:
IF $0 <= AGE < 40.0 \ \ -1.0 <= AL < 0.2$ THEN $0 <= LM < 30 \ \ 0 <= LAD < 50$, s=0.02 c=0.89 l=1.9
IF $SEX = F \ \ 0 <= CHOL < 200$ THEN $0 <= LAD < 50 \ \ 0 <= RCA < 50$, s=0.02 c=0.73 l=2.1
IF $SEX = F \ \ 0 <= CHOL < 200$ THEN $0 <= LCX < 50 \ \ 0 <= RCA < 50$, s=0.02 c=0.73 l=1.8
Confidence $>= 0.9$:
IF $40.0 <= AGE < 60.0 \ \ -1.0 <= LI < 0.2 \ \ 0 <= CHOL < 200$ THEN $0 <= LCX < 50$, s=0.03 c=0.90 l=1.5
IF $40.0 <= AGE < 60.0 \ \ -1.0 <= IL < 0.2 \ \ 0 <= CHOL < 200$ THEN $0 <= LCX < 50$, s=0.03 c=0.92 l=1.5
IF $40.0 <= AGE < 60.0 \ \ -1.0 <= IL < 0.2 \ \ SMOKE = n$ THEN $0 <= LCX < 50$, s=0.01 c=0.90 l=1.5
IF $40.0 <= AGE < 60.0 \ \ SEX = F \ \ DIAB = n$ THEN $0 <= LCX < 50$]), s=0.08 c=0.92 l=1.5
IF $HTA = n \ \ SMOKE = n \ \ 0 <= CHOL < 200$ THEN $0 <= LCX < 50$, s=0.02 c=0.92 l=1.5
Only risk factors:
IF $0 <= AGE < 40.0$ THEN $0 <= LAD < 50$, s=0.03 c=0.82 l=1.7
IF $0 <= AGE < 40.0 \ \ DIAB = n$ THEN $0 <= LAD < 50$, s=0.03 c=0.82 l=1.7
IF $40.0 <= AGE < 60.0 \ \ SEX = F \ \ DIAB = n$ THEN $0 <= LAD < 50$, s=0.07 c=0.72 l=1.5
IF $40.0 <= AGE < 60.0 \ \ SMOKE = n$ THEN $0 <= LCX < 50$, s=0.11 c=0.75 l=1.2
IF $40.0 <= AGE < 60.0 \ \ SMOKE = n$ THEN $0 <= RCA < 50$, s=0.11 c=0.76 l=1.3
Support $>= 0.2$:
IF $-1.0 <= IL < 0.2 \ \ DIAB = n$ THEN $0 <= LCX < 50$, s=0.41 c=0.72 l=1.2
IF $-1.0 <= LA < 0.2$ THEN $0 <= LCX < 50$, s=0.39 c=0.72 l=1.2
IF $SEX = F$ THEN $0 <= LCX < 50$, s=0.23 c=0.73 l=1.2
IF $40.0 <= AGE < 60.0 \ \ -1.0 <= IL < 0.2$ THEN $0 <= RCA < 50$, s=0.21 c=0.73 l=1.3

No	Name	Data Type	Medical Info	Description	Constraints Neg	g	ac
23	*PSTROKE*	C	R	Prior stroke Y/N	N	0	1
24	*PCARSUR*	C	R	Prior carotid surgery Y/N	N	0	1
25	*CHOL*	N	R	Cholesterol level	N	0	1

Table from [5]

Deep Learning Neural Networks for predicting response in cancer treatment

- Analysis of molecular profiles of 1001 cancer cell lines – for extracting cancer-specific signatures in the form of interpretable rules
- The association-rules are used as features for the DLNN framework
- Prediction if a cell-line would be sensitive or resistant to a given drug, also predict pharmacological responses to a large number of anti-cancer drugs – step towards precision medicine

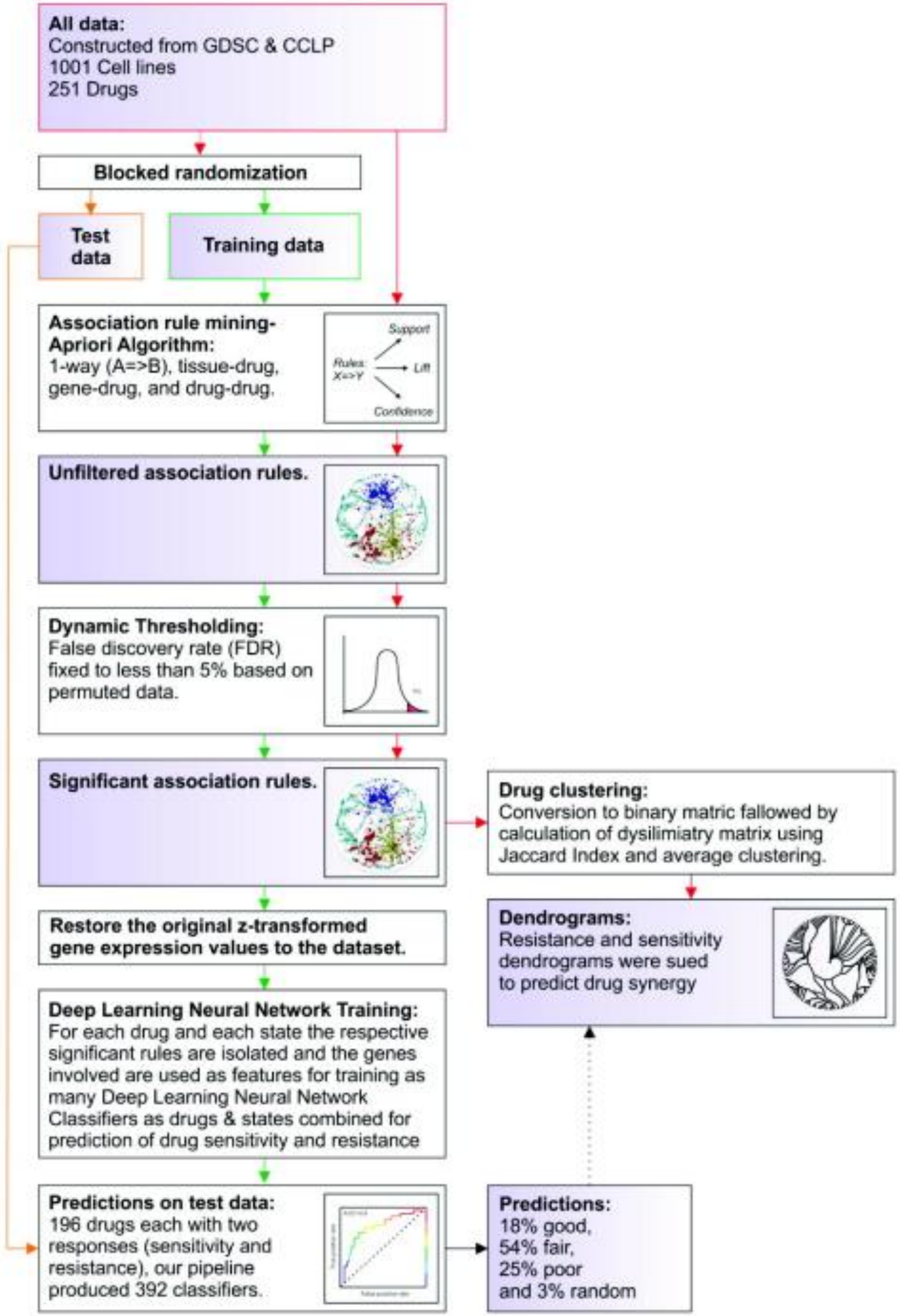

All data:
Constructed from GDSC & CCLP
1001 Cell lines
251 Drugs
Blocked randomization
Test data
Training data
Association rule mining-Apriori Algorithm:
1-way (A=>B), tissue-drug, gene-drug, and drug-drug.
Support
Rules: X=>Y
Lift
Confidence
Unfiltered association rules.
Dynamic Thresholding:
False discovery rate (FDR) fixed to less than 5% based on permuted data.
Significant association rules.
Drug clustering:
Conversion to binary matric fallowed by calculation of dysilimiatry matrix using Jaccard Index and average clustering.
Dendrograms:
Resistance and sensitivity dendrograms were sued to predict drug synergy
Restore the original z-transformed gene expression values to the dataset.
Deep Learning Neural Network Training:
For each drug and each state the respective significant rules are isolated and the genes involved are used as features for training as many Deep Learning Neural Network Classifiers as drugs & states combined for prediction of drug sensitivity and resistance
Predictions on test data:
196 drugs each with two responses (sensitivity and resistance), our pipeline produced 392 classifiers.
Predictions:
18% good,
54% fair,
25% poor
and 3% random

Image from [6]

What have we learned?

- Association Rule Problem
- Apriori Algorithm
- Rule Generation
- Measures for Association Rules

References

[1] Pang-Ning Tan, Michael Steinbach and Vipin Kumar (2018). *Introduction to Data Mining, (Second Edition), Chapter 5 Association Analysis: Basic Concepts and Algorithms*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2018.

[2] Gabriela Şerban, Istvan Czibula and Alina Campan. (2007). *Medical diagnosis prediction using relational association rules*. International Conference on Theory and Applications of Mathematics and Informatics (ICTAMI'07).

[3] Pavel Turčínek, and Jana Turčínková. (2015). *Exploring Consumer Behavior: Use of Association Rules*. Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis. 63. pp 1031-1042.

[4] Alexandros Nanopoulos, Apostolos N. Papadopoulos, and Yannis Manolopoulos. (2007). Mining association rules in very large clustered domains. Inf. Syst. 32, 5 (July 2007), 649-669.

[5] Ordonez, C., Ezquerra, N., Santana, C.A. (2006). "Constraining and summarizing association rules in medical data." International Journal of Knowledge Information System, Vol.9, Issue.3, pp.259-283.

[6] K. Vougas, M. Krochmal, T. Jackson, A. Plyzos, A. Aggelopoulos, P. Ioannis, M. Liontos, A. Varvarigou, E. Johnson, V. Georgoulias, A. Vlahou, P. Townsend, D. Thanos, J. Bartek, V. G. Gorgoulis, (2017). *Deep Learning and Association Rule Mining for Predicting Drug Response in Cancer. A Personalised Medicine Approach,* Cold Spring Harbor Laboratory, doi: 10.1101/070490.