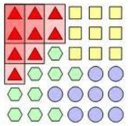
Chapter 3

Data Preprocessing and Data Mining

Instructor: Suresh Pokharel
ME in ICT (Asian Institute of Technology, Thailand)
BE in Computer (NCIT, Pokhara University)

Data Mining Tasks



1. **Classification:** learning a function that maps an item into one of a set of predefined classes
2. **Regression:** learning a function that maps an item to a real value
3. **Clustering:** identify a set of groups of similar items
4. **Dependencies and associations:** identify significant dependencies between data attributes
5. **Summarization:** find a compact description of the dataset or a subset of the dataset

Data Mining Methods

1. Decision Tree Classifiers:

   Used for modeling, classification

2. Association Rules:

   Used to find associations between sets of attributes

3. Sequential patterns:

   Used to find temporal associations in time series

4. Hierarchical clustering:

   Used to group customers, web users, etc

Data Mining Task Primitives

Data mining primitives define a data mining task, which can be specified in the form of a data mining query.

These primitives allow the user to inter-actively communicate with the data mining system during discovery in order to direct the mining process, or examine the findings from different angles or depths.

- Task Relevant Data
- Kinds of knowledge to be mined
- Background knowledge
- Interestingness measure
- Presentation and visualization of discovered patterns

Task relevant data

- Data portion to be investigated.
- Attributes of interest (relevant attributes) can be specified.

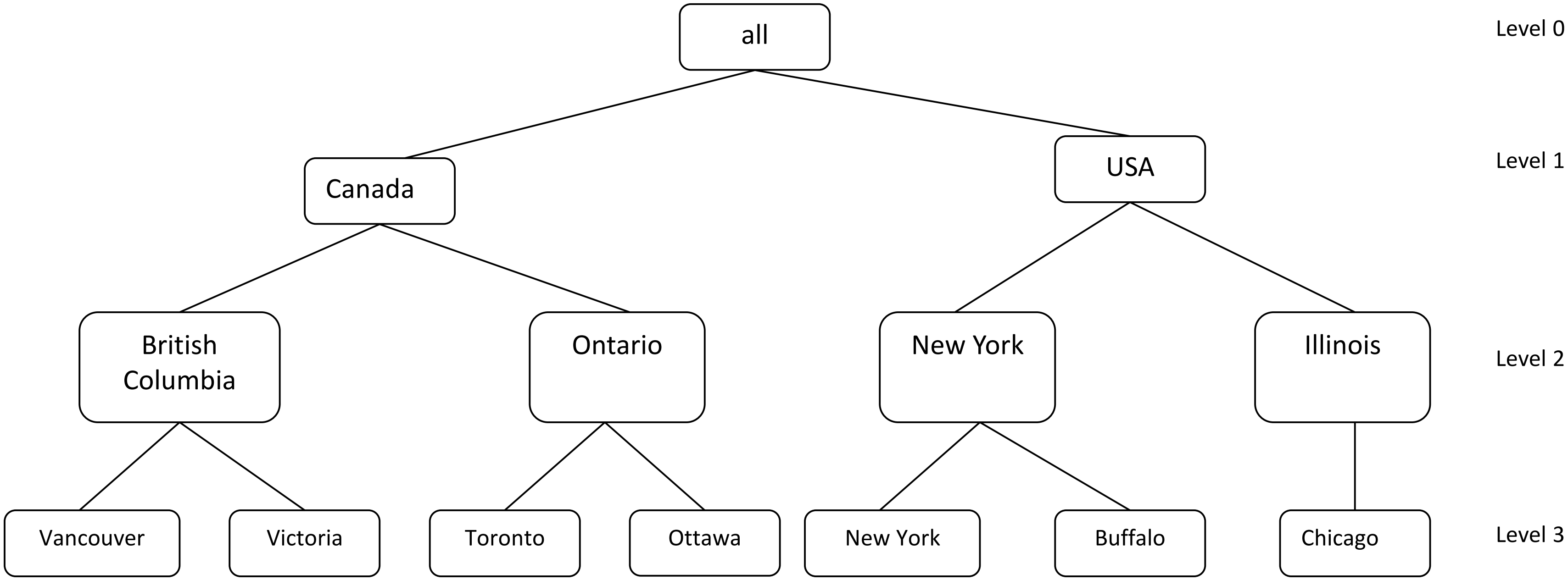
Kind of knowledge to be mined

- It is important to specify the knowledge to be mined, as this determines the data mining function to be performed.
- Kinds of knowledge include concept description, association, classification, prediction and clustering.

Background knowledge

- It is the information about the domain to be mined
- Concept hierarchy: is a powerful form of background knowledge.
- Major types of concept hierarchies:
  schema hierarchies
  set-grouping hierarchies
  rule-based hierarchies

Example



all
Level 0
Canada
USA
Level 1
British Columbia
Ontario
New York
Illinois
Level 2
Vancouver
Victoria
Toronto
Ottawa
New York
Buffalo
Chicago
Level 3

Concept hierarchies

- Rolling Up - Generalization of data
  Allows to view data at more meaningful and explicit abstractions.
  Makes it easier to understand
  Compresses the data
  Would require fewer input/output operations
- Drilling Down - Specialization of data
  Concept values replaced by lower level concepts

Schema hierarchies

- May formally express existing semantic relationships between attributes.

- Example: location hierarchy
  street < *city* < *province/state* < *country*

Set-grouping hierarchies

- Organizes values for a given attribute into groups or sets or range of values.

- Example: Set-grouping hierarchy for age
  {young, middle_aged, senior} $\subset$ *all (age)*
  {20….29} $\subset$ *young*
  {40….59} $\subset$ *middle_aged*
  {60….89} $\subset$ *senior*

Rule-based hierarchies

- Example: Following rules are used to categorize items as *low_profit, medium_profit* and *high_profit_margin*.

  $low\_profit\_margin(X) <= price(X,P1) \wedge cost(X,P2) \wedge ((P1-P2)<50)$

  $medium\_profit\_margin(X) <= price(X,P1) \wedge cost(X,P2) \wedge ((P1-P2) \geq 50) \wedge ((P1-P2) \leq 250)$

  $high\_profit\_margin(X) <= price(X,P1) \wedge cost(X,P2) \wedge ((P1-P2)>250)$

Interestingness measure

- Used to confine the number of uninteresting patterns returned by the process.
- Based on the structure of patterns and statistics underlying them.
- Associate a threshold which can be controlled by the user.
- Patterns not meeting the threshold are not presented to the user.
- Objective measures of pattern interestingness:
  certainty (confidence)
  utility (support)
  novelty

Presentation and visualization

- For data mining to be effective, data mining systems should be able to display the discovered patterns in multiple forms, such as rules, tables, crosstabs (cross-tabulations), pie or bar charts, decision trees, cubes, or other visual representations.

- User must be able to specify the forms of presentation to be used for displaying the discovered patterns.

What Is Frequent Pattern Analysis?

- Frequent pattern: a pattern (a set of items, subsequences, substructures, etc.) that occurs frequently in a data set
- First proposed by Agrawal, Imielinski, and Swami [AIS93] in the context of frequent itemsets and association rule mining
- Motivation: Finding inherent regularities in data
  - What products were often purchased together?— Beer and diapers?!
  - What are the subsequent purchases after buying a PC?
  - What kinds of DNA are sensitive to this new drug?
  - Can we automatically classify web documents?
- Applications
  - Basket data analysis, cross-marketing, Web log (click stream) analysis, and DNA sequence analysis.

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

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

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 $\rightarrow$ Y, where X and Y are itemsets
  - Example:
    {Milk, Diaper} $\rightarrow$ {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:

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

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

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

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
- 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!

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:

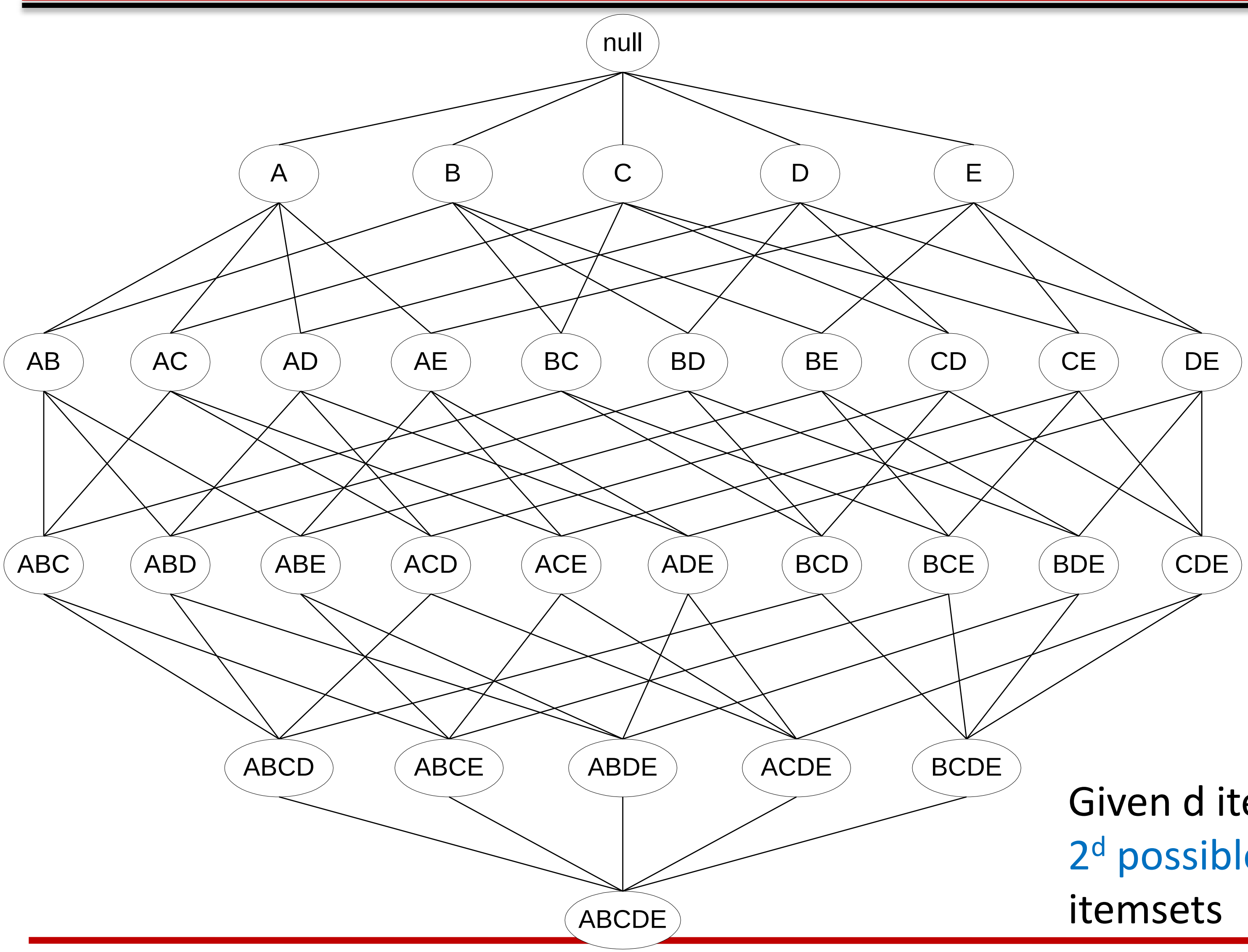
- All the above rules are binary partitions of the same itemset:
  {Milk, Diaper, Beer}
- Rules originating from the same itemset have identical support but can have different confidence
- Thus, we may decouple the support and confidence requirements

Mining Association Rules

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

Frequent Itemset Generation



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

Frequent Itemset Generation

Brute-force approach:

Each itemset in the lattice is a candidate frequent itemset

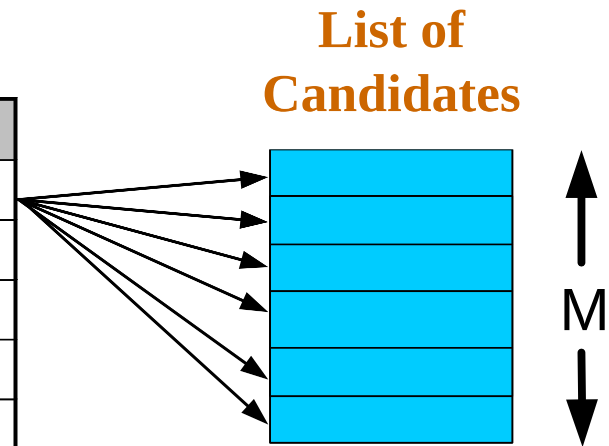
Count the support of each candidate by scanning the database

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

N

w



Match each transaction against every candidate

Complexity ~ O(NMw) => Expensive since $M = 2^d$!!!

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:

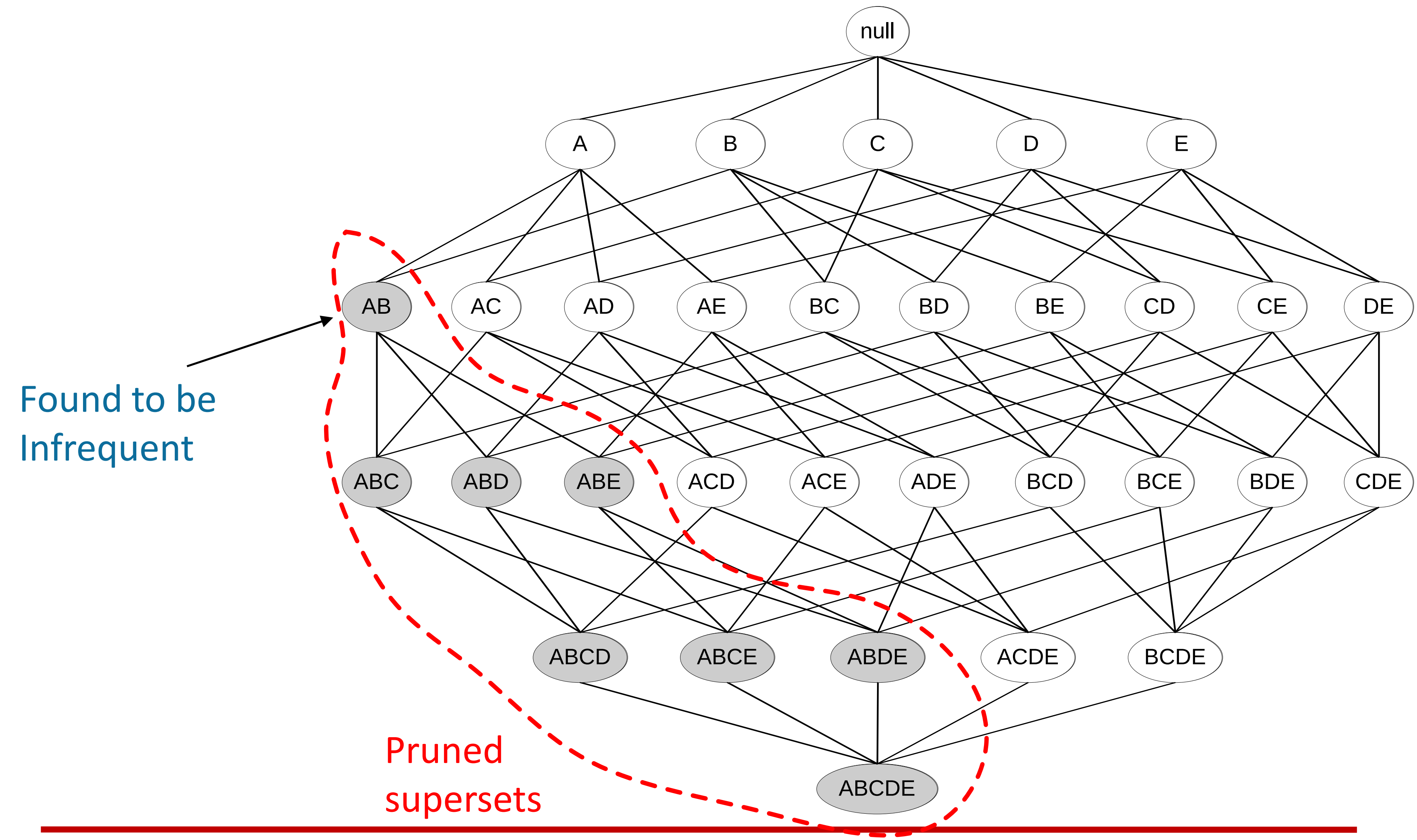
$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

Support of an itemset never exceeds the support of its subsets

This is known as the anti-monotone property of support

Anti-monotone: if a set can’t pass a test, all of its superset will fail the same test as well

Illustrating Apriori Principle



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

Items (1-itemsets)

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

Pairs (2-itemsets)

(No need to generate candidates involving Coke or Eggs)

Minimum Support = 3

Triplets (3-itemsets)

Itemset	Count
{Bread,Milk,Diaper}	3

If every subset is considered,

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

With support-based pruning,

$6 + 6 + 1 = 13$

Q: Total number of possible frequent itemsets ???

Apriori Algorithm

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(prune) candidates that are infrequent, leaving only those that are frequent

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

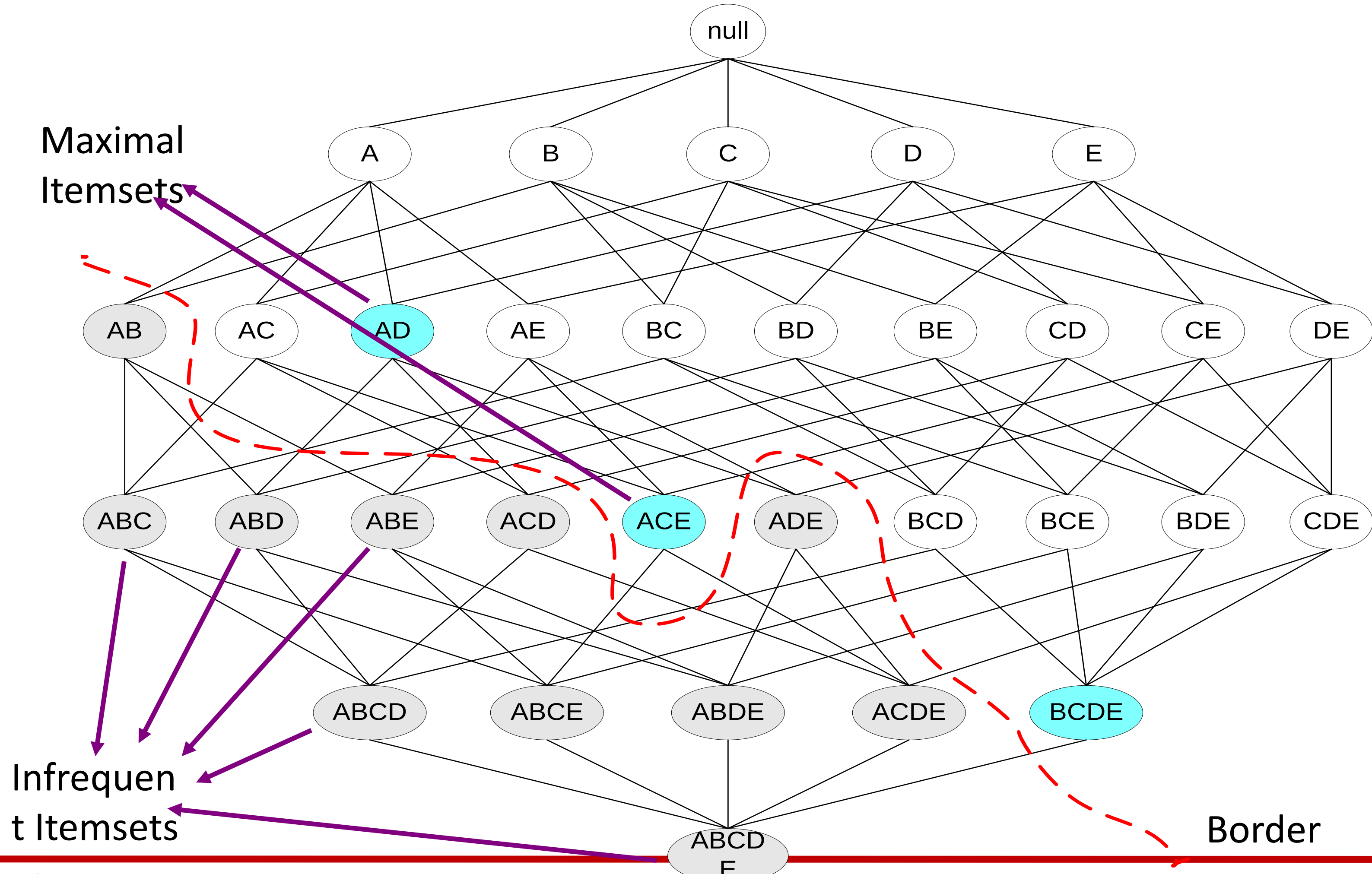
L_3

itemset	sup
{2 3 5}	2

Why {1 2 3}, {1 2 5}, {1 3 5} are not listed in C3???

Maximal Frequent Itemset

An itemset is maximal frequent if none of its immediate supersets is frequent



Closed Itemset

An itemset is closed if none of its immediate supersets has the same support as the itemset

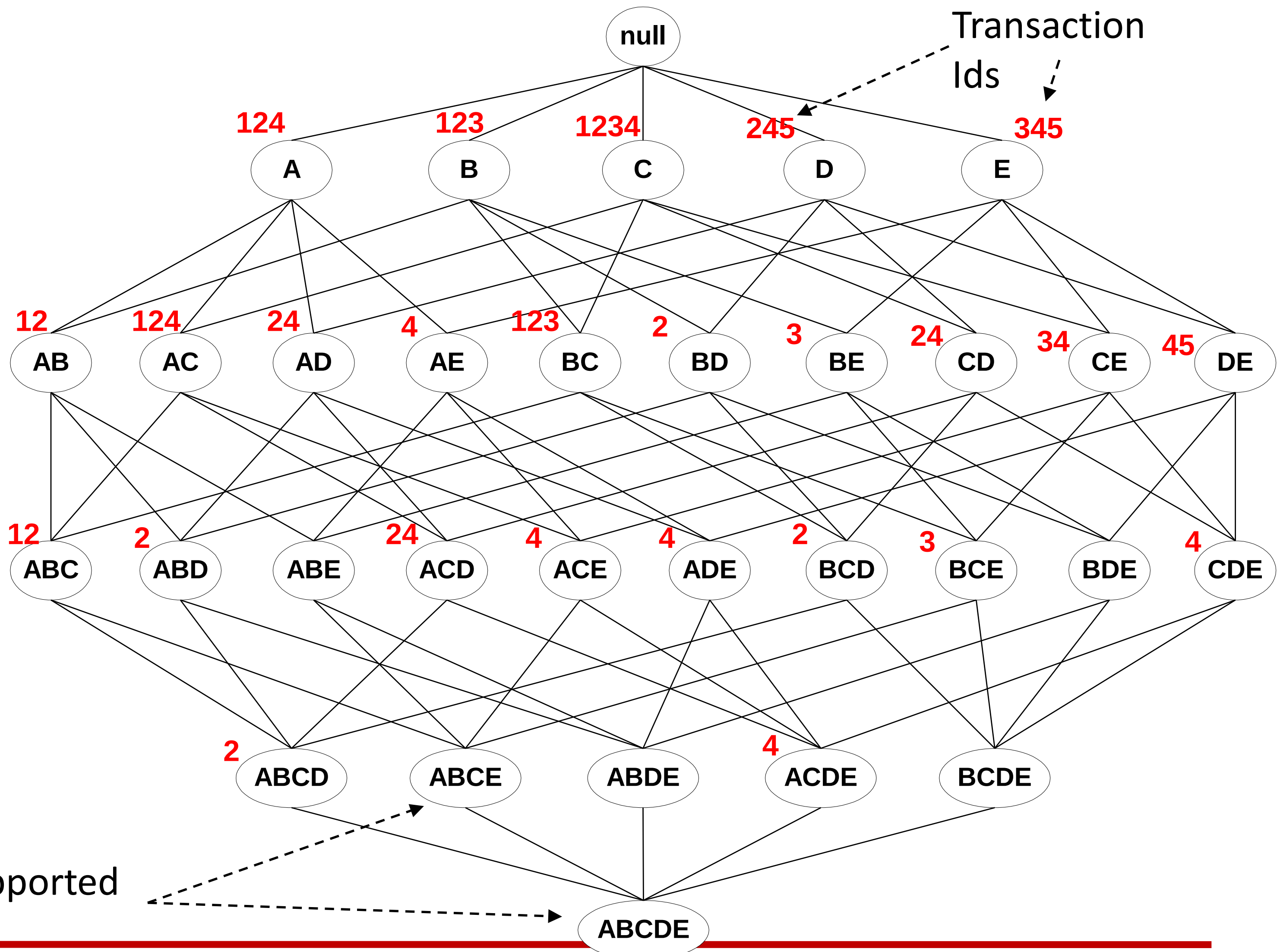
TID	Items
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

Itemset	Support
{A}	4
{B}	5
{C}	3
{D}	4
{A,B}	4
{A,C}	2
{A,D}	3
{B,C}	3
{B,D}	4
{C,D}	3

Itemset	Support
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	3
{A,B,C,D}	2

Maximal vs Closed Itemsets

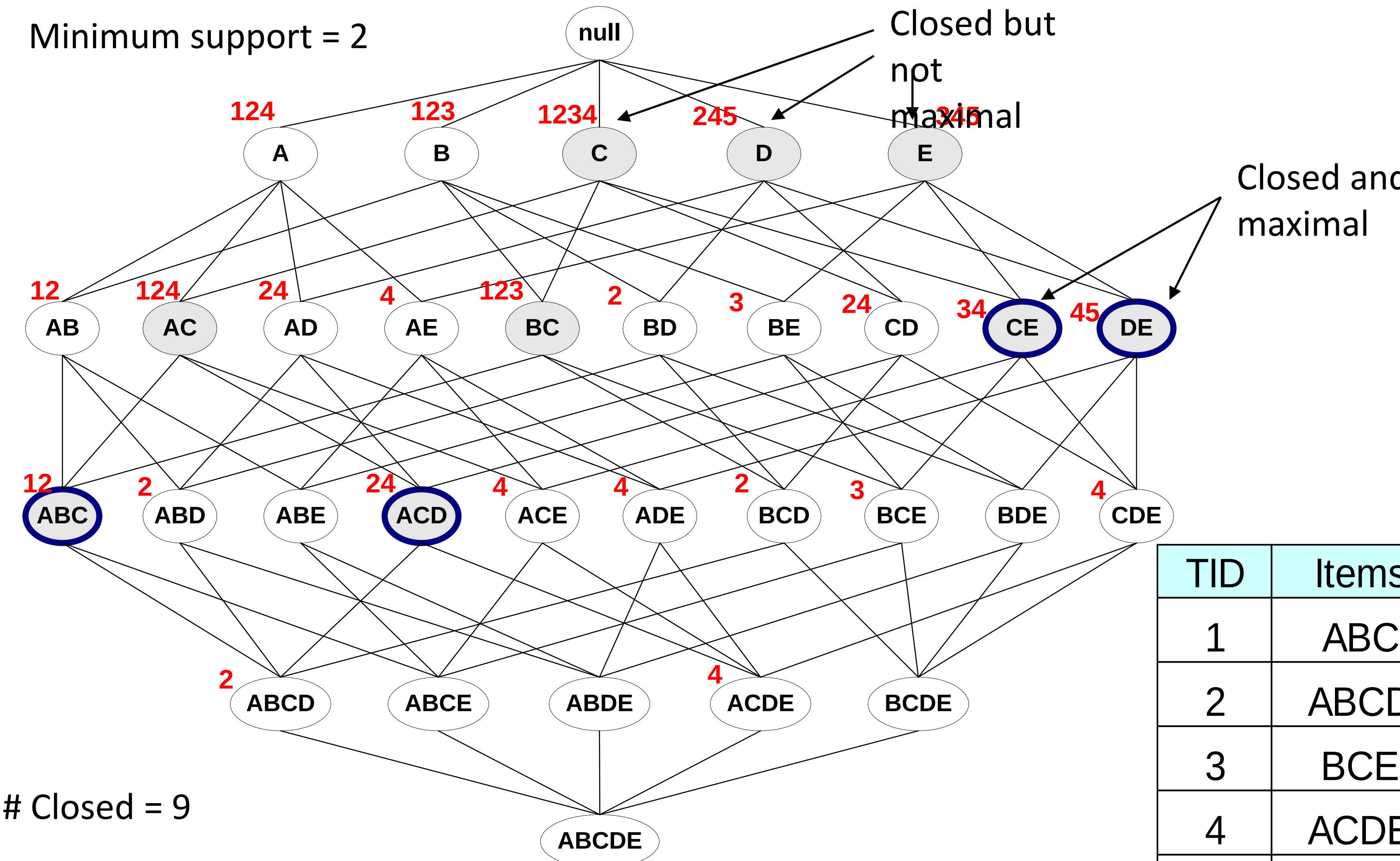
TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE



Not supported by any transactions

Maximal vs Closed Frequent Itemsets

Minimum support = 2

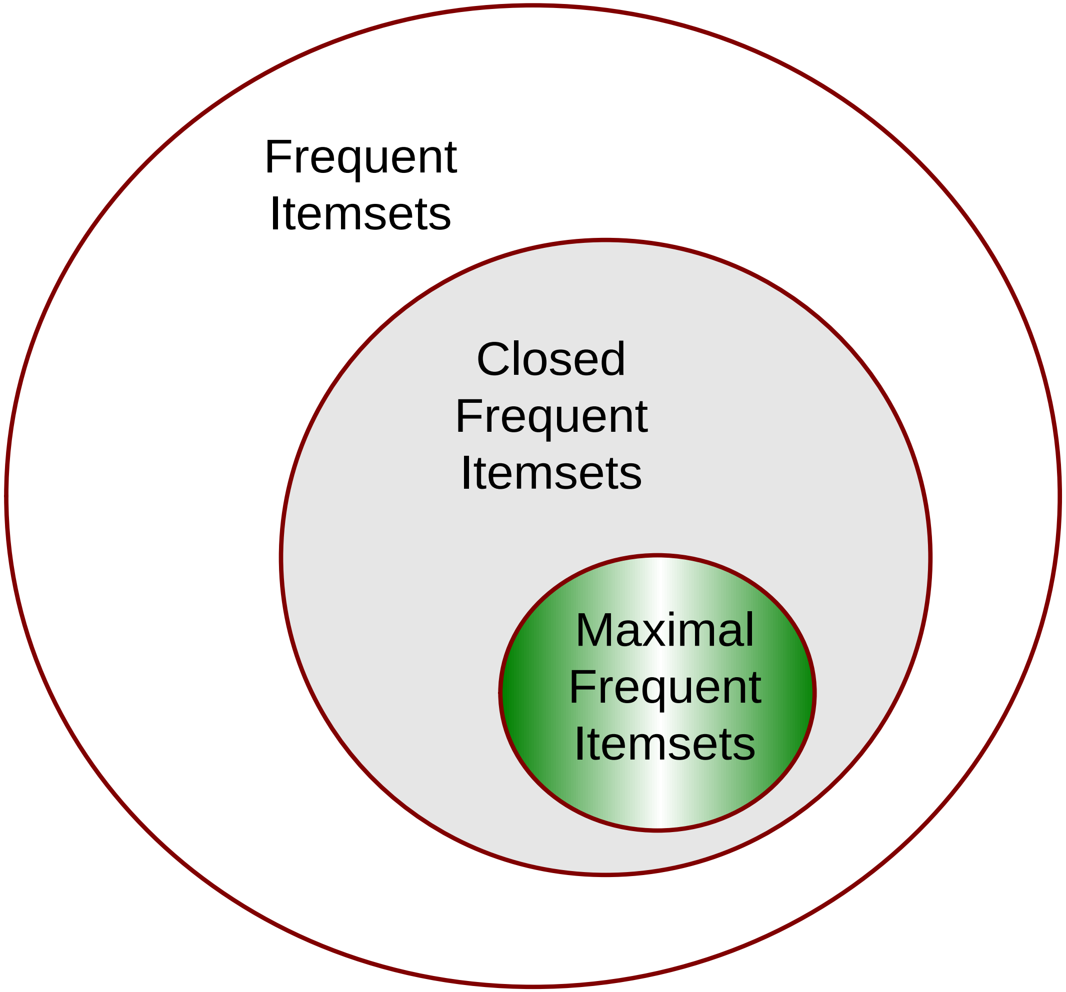


TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE

Closed = 9

Maximal = 4

Maximal vs Closed Itemsets



Frequent Pattern Tree

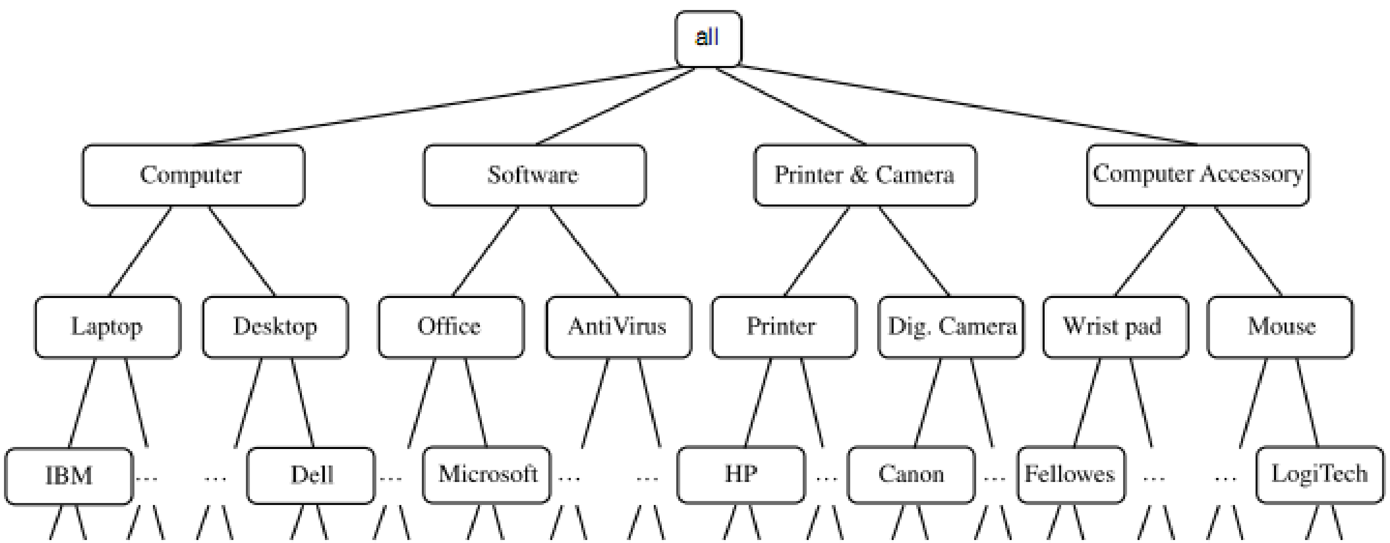
Generating Association Rule (Example)

- **Given a frequent itemset L**
  - Find all non-empty subsets F in L, such that the association rule $F \Rightarrow \{L-F\}$ satisfies the minimum confidence
  - Create the rule $F \Rightarrow \{L-F\}$

- **If L={A,B,C}**
  - The candidate itemsets are: $AB \Rightarrow C$, $AC \Rightarrow B$, $BC \Rightarrow A$, $A \Rightarrow BC$, $B \Rightarrow AC$, $C \Rightarrow AB$
  - In general, there are 2^K-2 candidate solutions, where k is the length of the itemset L

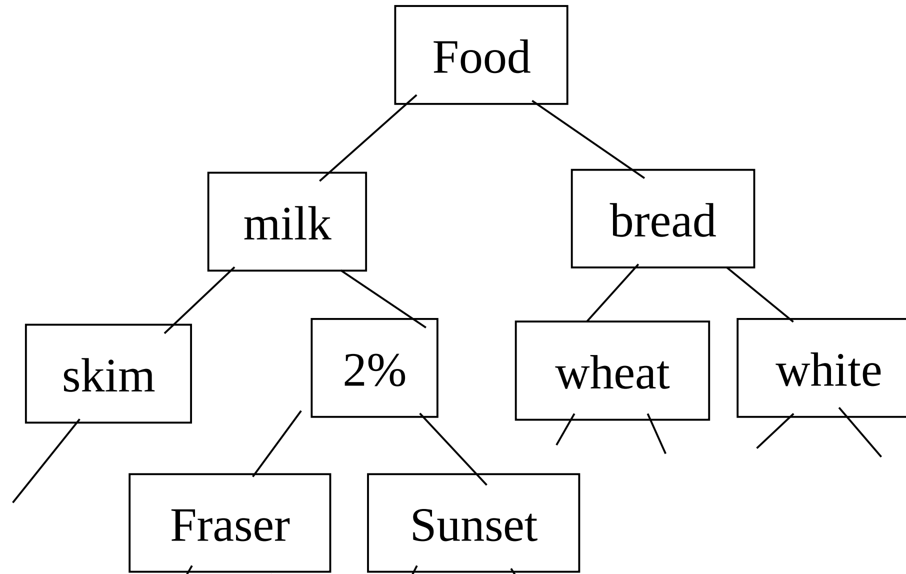
Recap : A Concept Hierarchy

TID	Items Purchased
T100	IBM-ThinkPad-T40/2373, HP-Photosmart-7660
T200	Microsoft-Office-Professional-2003, Microsoft-Plus!-Digital-Media
T300	Logitech-MX700-Cordless-Mouse, Fellowes-Wrist-Rest
T400	Dell-Dimension-XPS, Canon-PowerShot-S400
T500	IBM-ThinkPad-R40/P4M, Symantec-Norton-Antivirus-2003
...	...



Multiple-Level Association Rules

- Items often form hierarchy.
- Items at the lower level are expected to have lower support.
- Rules regarding itemsets at appropriate levels could be quite useful.
- We can explore shared multi-level mining



TID	Items
T1	{111, 121, 211, 221}
T2	{111, 211, 222, 323}
T3	{112, 122, 221, 411}
T4	{111, 121}
T5	{111, 122, 211, 221, 413}

Mining Multi-Level Associations

- A top_down, progressive deepening approach:
  - First find high-level strong rules:

    milk $\rightarrow$ bread [20%, 60%].
  - Then find their lower-level "weaker" rules:

    2% milk $\rightarrow$ wheat bread [6%, 50%].
- Variations at mining multiple-level association rules.
  - Association rules with multiple, alternative hierarchies:

    2% milk $\rightarrow$ *Wonder bread*

Multi-level Association: Uniform Support vs. Reduced Support

- Uniform Support: the same minimum support for all levels
  - **+** One minimum support threshold. No need to examine itemsets containing any item whose ancestors do not have minimum support.
  - **–** Lower level items do not occur as frequently. If support threshold
    - too high $\Rightarrow$ miss low level associations
    - too low $\Rightarrow$ generate too many high level associations
- Reduced Support: reduced minimum support at lower levels

Uniform Support

Multi-level mining with uniform support

Level 1
min_sup = 5%

Milk

[support = 10%]

Level 2
min_sup = 5%

2% Milk

[support = 6%]

Skim Milk

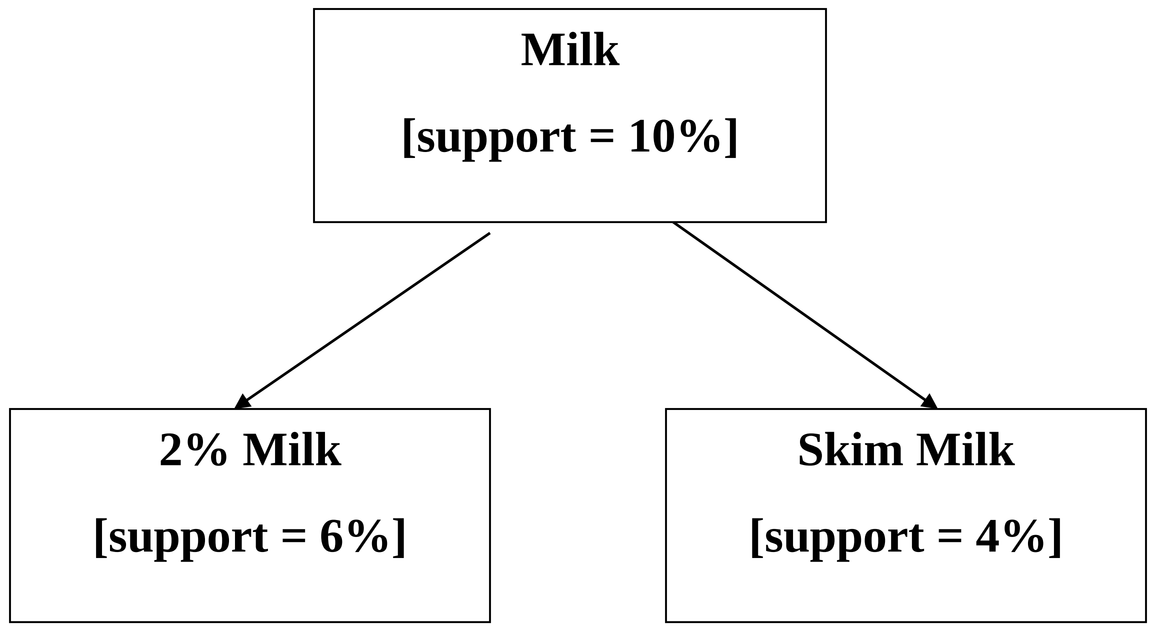
[support = 4%]

Reduced Support

Multi-level mining with reduced support

Level 1
min_sup = 5%

Level 2
min_sup = 3%



Multi-level Association: Redundancy Filtering

- Some rules may be redundant due to “ancestor” relationships between items.
- Example
  - milk $\Rightarrow$ wheat bread [support = 8%, confidence = 70%]
  - 2% milk $\Rightarrow$ wheat bread [support = 2%, confidence = 72%]
- We say the first rule is an ancestor of the second rule.

Multi-Dimensional Association: Concepts

- Single-dimensional rules:

  buys(X, "milk") $\Rightarrow$ buys(X, "bread")

- Multi-dimensional rules: ❍ 2 dimensions or predicates
  - Inter-dimension association rules (*no repeated predicates*)

    age(X,"19-25") $\wedge$ occupation(X,"student") $\Rightarrow$ buys(X,"coke")
  - hybrid-dimension association rules (*repeated predicates*)

    age(X,"19-25") $\wedge$ buys(X, "popcorn") $\Rightarrow$ buys(X, "coke")

Interestingness Measurements

- Objective measures

  Two popular measurements:

  ☆ *support;* and

  ◷ *confidence*

- Subjective measures

  A rule (pattern) is interesting if

  ☆ it is *unexpected* (surprising to the user); and/or

  ◷ *actionable* (the user can do something with it)

Mining Class Comparison

- Compare General properties of Graduate Student Vs Undergraduate Student
- Attributes : name, gender, major, birth place, birth date, residence, phone#, and GPA.
- DMQL

```
use Big_University_DB
mine comparison as "grad_vs_undergrad_students"
in relevance to name, gender, major, birth_place, birth_date, residence,
        phone#, gpa
for "graduate_students"
where status in "graduate"
versus "undergraduate_students"
where status in "undergraduate"
analyze count%
from student
```

Mining Class Comparison

Table 1: Initial working relations: the *target class* (graduate students)

name	*gender*	*major*	*birth_place*	*birth_date*	*residence*	*phone#*	*gpa*
Jim Woodman	M	CS	Vancouver, BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Vancouver	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
...	...	...	...	...	...	...	...

Table 2: Initial working relations: the *contrasting class* (undergraduate students)

name	*gender*	*major*	*birth_place*	*birth_date*	*residence*	*phone#*	*gpa*
Bob Schumann	M	Chemistry	Calgary, Alt, Canada	10-1-78	2642 Halifax St., Burnaby	294-4291	2.96
Amy Eau	F	Biology	Golden, BC, Canada	30-3-76	463 Sunset Cres., Vancouver	681-5417	3.52
...	...	...	...	...	...	...	...

Mining Class Comparison

Prime generalized relation for the *target class* (graduate students)

major	*age_range*	*gpa*	*count%*
Science	21...25	good	5.53%
Science	26...30	good	5.02%
Science	over_30	very_good	5.86%
...	...	...	...
Business	over_30	excellent	4.68%

Prime generalized relation for the *contrasting class* (undergraduate students)

major	*age_range*	*gpa*	*count%*
Science	16...20	fair	5.53%
Science	16...20	good	4.53%
...	...	...	...
Science	26...30	good	2.32%
...	...	...	...
Business	over_30	excellent	0.68%

Mining Class Comparison

Count distribution between graduate and undergraduate students for a generalized tuple.

status	*major*	*age_range*	*gpa*	count
graduate	Science	21...25	good	90
undergraduate	Science	21...25	good	210

For More.. See. Book P210


Q&A