

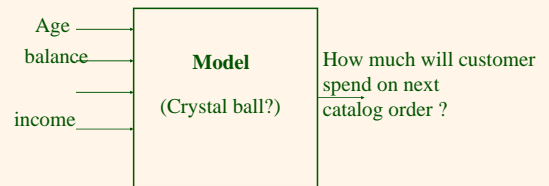


## Data Mining & Knowledge Discovery: Personalization and Profiling Technologies

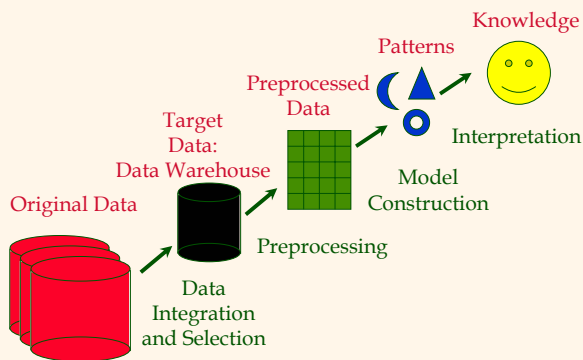
## Predictive Modeling and Knowledge Discovery via Data Mining



- ❖ A “black box” that makes predictions about the future based on information from the past and present



## Preprocessing and Mining



## Data Mining Steps and Techniques



- ❖ Data Warehouse
- ❖ On-Line Analytical Processing (OLAP)
- ❖ Mining techniques
  - Market Basket Analysis
  - Cluster Detection
  - Decision Trees and Rule Induction
  - Neural Networks
  - Genetic Algorithms

## Data Warehouse



- ❖ Warehouse is a large database
- ❖ First collect the data that you will analyze
- ❖ find, summarize, and interpret large amounts of data effectively
  - data is distributed across many different databases
- ❖ Cannot get from Transaction processing system; cannot interfere with its timing
  - Get from off-line source - "old" TP data
- ❖ integrate data from various TPS to a single storage which can then be analyzed...
- ❖ Is this integration/warehousing easy?
  - Noise, missing data, redundancy, data transformation (numeric to text etc.)

## Legal and Ethical Issues



- ❖ Privacy concerns
  - becoming more important
  - will impact the way that data can be used and analyzed
  - ownership issues
- ❖ Often data included in the data warehouse cannot legally be used in decision making process
  - Race, Gender, Age
- ❖ Data contamination will become critical

## OLAP: OnLine Analytical Processing



- ❖ Mainly a 'data presentation' concept
- ❖ Way of presenting data to facilitate understanding any patterns inside it
  - Provide for drilling down into the data starting from summarized views
- ❖ OLAP lets you look at data and manipulate interactively
- ❖ OLAP allows users to "slice and dice" data
  - Allows user to drill-down into detail data

## Relational vs Multidimensional



Product	Region	Sales
Nuts	East	50
Nuts	West	60
Nuts	Central	100
Screws	East	40
Screws	West	70
Screws	Central	80
Bolts	East	90
Bolts	West	120
Bolts	Central	140
Washers	East	20
Washers	West	10
Washers	Central	30

	East	West	Central
Nuts	50	60	100
Screws	40	70	80
Bolts	90	120	140
Washers	20	10	30

## OLAP Consolidations



	East	West	Central	Total
Nuts	50	60	100	210
Screws	40	70	80	190
Bolts	90	120	140	350
Washers	20	10	30	60
Total	200	260	350	810

## Data Mining techniques



- ❖ How do you define a “pattern” of interest ?
- ❖ Three common techniques
  - Prediction and Classification
  - Affinity Grouping
  - Clustering

## Data Mining techniques: Prediction and Classification



- ❖ Prediction and Classification
  - Directed: if A then B
  - Examples:
    - Forecast: How many units will be sold on a given day?
    - What will be the stock price on a given day?
    - Will a customer buy the product or not?
    - Who are the high paying customers of T-Mobile
    - Detect sequences:
      - balance increase → missed payment → default
      - paid in cash → one-way ticket → question the passenger

## DM techniques: Affinity Grouping



- ❖ Affinity grouping
  - Undirected
  - Which products go together naturally?
  - Market basket analysis
  - Examples:
    - Which products peak in demand simultaneously?

## DM techniques: Clustering



- ❖ Clustering task
  - Undirected
  - Segmenting into similar clusters
  - Different from classification
  - Examples
    - Customers with similar buying profiles
    - Products with similar demand patterns
    - Classification and segmentation: yes/no

## Data Mining Algorithms



- ❖ Four algorithms commonly cited
  - Association Rule (used in over 90% of the cases!)
  - Nearest Neighbor
    - quick and easy but models get large
    - Cluster detection using geometric and statistical methods
  - Decision Tree
    - Used for prediction techniques
  - Neural Network
    - difficult to interpret and large time

## Market Basket Analysis: Association Rule Mining



- ❖ Form of clustering used for finding items that occur together (in a transaction or market basket)
- ❖ Likelihood of different products being purchased together as rules
- ❖ Planning store layouts, limiting specials to one of the products in a set,...

## Supermarket Transaction data



Customer	Products
1	Milk, Soda
2	Milk, Beer, diapers
3	Milk, cleaner
4	Beer, diapers, soda
5	Beer, soda

## Association Rules: Support and Confidence Parameters



- ❖ looking for a rule that says: If A then B
- ❖ **Support** is defined as the ratio of number of transactions that include both A and B to total number of transactions
  - How useful are A and B
- ❖ **Confidence** is defined by the ratio of the number of transactions that include both A and B to the number of transactions that include A.
  - *if A then B with probability p*
- ❖ How do you specify 'significant' support and confidence?

## Association Rule example



- ❖ Using the sample data create a co-occurrence table
  - What items occur together

## Co-occurrence Matrix



	Milk	Soda	Beer	Diapers	Cleaner
Milk	3	1	1	1	1
Soda	1	2	2	1	0
Beer	1	2	3	2	0
Diapers	1	1	2	2	0
Cleaner	1	0	0	0	1

## Association Rule example



- ❖ Using the sample data create a co-occurrence table
- ❖ Let relevant Support = 25% and Confidence= 50%:
  - Beer and Diapers appear in  $2/5 = 40\%$
  - If beer then diapers has confidence of  $2/3 = 67\%$
  - Thus, "If customer buys beer then customer buys diapers" satisfies 25% support & 50% confidence
- ❖ Conclusion drawn by mining system:
  - Customers who buy beer also buy diapers

## Applying the Results



- ❖ Is the relationship useful ?
  - Beer and Diapers may not be of use
  - Retailers use transaction mining to send specific apparel to specific stores -- Microstrategy software
- ❖ Who defines "usefulness"
  - only as good as rules specified by humans/ marketing workforce
  - NBA mining: designers of s/w did not include height mismatches at first...coaches made the correction

## Algorithm for Finding Association Rules



- ❖ Input is Min-Support and Min-Confidence
- ❖ Find all sets of items with Min-Support (*frequent itemsets*)
  - Can mean searching all subsets?
- ❖ **Frequent Itemsets Property: Every subset of a frequent itemset must also be a frequent itemset**
  - iterative algorithm: start with frequent itemsets with one item, and construct larger itemsets using only smaller frequent itemsets.

## Data Mining: Summary



- ❖ "Using the new media of the one-to-one future, you will be able to communicate directly with customers individually....." - Don Peppers & Martha Rogers (One-to-One Future)
- ❖ "What are you afraid of?....Even if you're not afraid of these things, the beauty is,with proper marketing, we can make you afraid"-- Michael Saylor, CEO Microstrategy.

## Course Announcements/Trivia...



- ❖ Project Demos:
  - Now scheduled for Wednesday
    - Lab section 9am
    - Lecture session 12:45
- ❖ Phase 3 teams will be assigned Thursday
  - Description posted

## Relational DBMS



- ❖ Logical level design
  - Get requirements from application
  - Design schema using SQL
  - Design queries using SQL and application language
- ❖ Physical Schema
  - Index structures
  - Locks and granularity of locks
  - Tuning
- ❖ Are relational DBMS the only form of information storage and retrieval systems ??

## Information Retrieval



- ❖ A research field traditionally separate from Databases
  - Goes back to IBM, Rand and Lockheed in the 50's
- ❖ Products traditionally separate
  - Originally, document management systems for libraries, government, law, etc.
  - Gained prominence in recent years due to web search
- ❖ Multimedia and spatial data adds new dimension

## Text "Indexes"



- ❖ When IR folks say "text index" ...
  - Usually mean more than what DB people mean
- ❖ In our terms, both "tables" and indexes
  - Really a logical schema (i.e., tables)
  - With a physical schema (i.e., indexes)
  - Usually not stored in a DBMS
    - Tables implemented as files in a file system

## A Simple Relational Text Index



- ❖ Create and populate a table  
**InvertedFile**(term string, docURL string)
- ❖ Build a B+-tree or Hash index on **InvertedFile.term**
  - Note: URL instead of RID, the web is your "heap file"!
    - Can also *cache* pages and use RIDs
- ❖ This is often called an "inverted file" or "inverted index"
  - Maps from **words** -> **docs**
- ❖ Can now do single-word text search queries!

## An Inverted File



- ❖ Search for
  - "databases"
  - "microsoft"

term	docURL
data	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
database	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
date	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
day	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
dbms	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
decision	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
demonstrate	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
description	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
design	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
desire	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
developer	<a href="http://www.microsoft.com">http://www.microsoft.com</a>
differ	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
disability	<a href="http://www.microsoft.com">http://www.microsoft.com</a>
discussion	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
division	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
do	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>
document	<a href="http://www-inst.eecs.berkeley.edu/~cs186">http://www-inst.eecs.berkeley.edu/~cs186</a>

## Handling Boolean Logic



- ❖ How to do "term1" OR "term2"?
  - Union of two DocURL sets!
- ❖ How to do "term1" AND "term2"?
  - Intersection of two DocURL sets!
    - Can be done by sorting both lists alphabetically and merging the lists
- ❖ How to do "term1" AND NOT "term2"?
  - Set subtraction, also done via sorting
- ❖ How to do "term1" OR NOT "term2"?
  - Union of "term1" and "NOT term2".
    - "Not term2" = all docs not containing term2. Large set!!
  - Usually not allowed!
- ❖ Refinement: What order to handle terms if you have many ANDs/NOTs?

## Boolean Search in SQL



"Windows" AND ("Glass" OR "Door")  
AND NOT "Microsoft"

- ❖ 

```
(SELECT docURL FROM InvertedFile
WHERE term = "windows"
INTERSECT
SELECT docURL FROM InvertedFile
WHERE term = "glass" OR term =
"door")
EXCEPT
SELECT docURL FROM InvertedFile
WHERE term ="Microsoft"
ORDER BY relevance()
```

## Boolean Search in SQL



- ❖ Really only one SQL query in Boolean Search IR:
  - Single-table selects, UNION, INTERSECT, EXCEPT
- ❖ **relevance ()** is the "secret sauce" in the search engines:
  - Combos of statistics, linguistics, and graph theory tricks!
  - Unfortunately, not easy to compute this efficiently using typical DBMS implementation.

## Relevance/Ranking?



- ❖ Lots of variation here
  - Often messy; details proprietary and fluctuating
- ❖ Relevance calculation involves how often search terms appear in doc, and how often they appear in collection:
  - More search terms found in doc → doc is more relevant
  - Greater importance attached to finding *rare* terms
  - Doing this efficiently in current SQL engines is not easy
- ❖ Combining subsets of:
  - **IR-style relevance**: Based on term frequencies, proximities, position (e.g., in title), font, etc.
  - **Popularity information**
  - **Link analysis information**

## Relevance: Going Beyond IR



- ❖ **Page "popularity"** (e.g., DirectHit)
  - Frequently visited pages (in general)
  - Frequently visited pages as a result of a query
- ❖ **Link "co-citation"** (e.g., Google)
  - Which sites are linked to by other sites?
  - Draws upon sociology research on bibliographic citations to identify "**authoritative sources**"

## Standard Web Search Engine Architecture

